

INSPIRE COMPUTING

International

Student Book

YEAR 8



INSPIRE **COMPUTING**

International

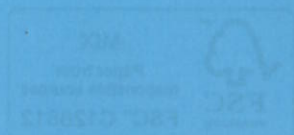
Student Book

YEAR 8

Paul Clowrey



Pearson



Overview contents

Detailed contents	iv
About the Student Book	vi
Unit 1: Binary and online graphics	2
Unit 2: Data on the Web	24
Unit 3: Processing Data	46
Unit 4: Analysing Data	74
Unit 5: Programming	94
Unit 6: Software and communication	116
Glossary	140

Detailed contents

Unit 1: Binary and online graphics

1 Vector graphics	4
2 Creating a vector graphic	6
3 Bitmaps	8
4 Comparing vectors and bitmaps.....	10
5 Binary and pixels	12
Mid-unit assessment	14
6 Colour depth and binary representation	16
7 and 8 Digital image properties	18
9 & 10 Binary addition	20
End-of-unit assessment	22

Unit 2: Data on the Web

1 Bits, bytes and nibbles	26
2 Storage capacity	28
3 Binary and Denary	30
4 Creating a 'power of' calculator	32
5 Internet Bandwidth	34
Mid-unit assessment	36

6 Creating a bandwidth calculator	38
7 & 8 Compression	40
9 & 10 Run Length Encoding	42
End-of-unit assessment	45

Unit 3: Processing Data

1 Computer hardware	48
2 Wearable technology	50
3 Logo design	52
4 Introduction to spreadsheets	54
5 Spreadsheet formulas and functions ...	56
Mid-unit assessment	58
6 Spreadsheet modelling	60
7 Environmental impact of technology	64
8 Presenting data in an informative way	66
9 Surveys and questionnaires	68
10 Advanced spreadsheet functions	70
End-of-unit assessment	72

Unit 4: Analysing Data

1 Social media	76
2 What makes a good spreadsheet	78
3 Knowledge, data and information	80
4 Sorting and filtering a spreadsheet	82
5 & 6 Design techniques for an online advertisement	84
Mid-unit assessment	86
7 and 8 What makes a good chart?	88
9 and 10 Designing an infographic	90
End-of-unit assessment	92

Unit 5: Programming

1 and 2 Algorithms and turtle programming	96
3 and 4 Sequencing and iteration algorithms	98
5 Programming errors	100
Mid-unit assessment	102
6 Using lists in a Python program	104
7 Use of selection in an algorithm	106
8 Creating a simple chatbot	108

9 Subprograms in an algorithm	110
10 Creating a missing word game	112
End-of-unit assessment	115

Unit 6: Software and communication

1 Surveillance technology	118
2 Types of software	120
3 Operating systems	122
4 Utility software	124
5 Internet hardware	126
Mid-unit assessment	128
6 Wireless communication technology	130
7 The internet and the World Wide Web	132
8 and 9 Surveillance ethics	134
10 Data breaches and cyber-crime	136
End-of-unit assessment	138

Glossary	140
----------------	-----

Welcome to Inspire Computing

Whether for school, fun, work or staying in touch with relatives around the world, digital technology is all around us.

Through coverage of ICT and Computer Science you will discover how this amazing technology works, how it connects the world together and it has revolutionised the classroom, workplace, and home.

Related topics

Other topics linked to the subject that can also be explored.

Real-world examples

How the learning applies to the world outside the classroom.

Further investigation

Take your learning from the lesson further!

Unit 5: Programming Part 1 Lesson 2

Sorting algorithms

Related topics

- Understanding and using variables
- Real-world systems

Key words

ascending	program
descending	sort
algorithm	list
alphabetical	string
	numerical

Learning objectives

- Understand the purpose of a sorting algorithm.
- Create, run and edit a simple sorting program.

A simple sort
A sorting algorithm reorders a list of values into the order required. Most sorts are numerical or alphabetical. Here are two examples of sorts.

Numerical sort
Original values: 2,6,4,9,1,3
Sorted into ascending order: 1,2,3,4,6,9
Sorted into descending order: 9,6,4,3,2,1

Alphabetical sort
Original values: E,D,Z,U,P,A
Sorted into ascending order: A,D,E,P,U,Z
Sorted into descending order: Z,U,P,E,D,A

Real-world examples
Sorting algorithms are used for:

- high score tables in a computer or smartphone game
- the songs in your playlist
- the contacts in your smartphone
- sales of items in a shop, sorted by price or product.

A sorting program
This program demonstrates how to create a simple alphabetical sort using Python. The `sort` command is applied to the characters stored in the `letters` list.

```
letters = ['E','D','Z','U','P','A']
letters.sort()
print(letters)
```

You can create a numerical sort in the same way.

```
numbers = [5,4,5,98,1,45,68]
numbers.sort()
print(numbers)
```

Key words

These are important words to know!

Learning objectives

This is what you will know or be able to do by the end of the lesson.

Unit 6: Internet protocol and binary Lesson 1

Real-world examples

Over 100 years ago, programs for the first digital computers were written using punched cards or paper tape. A hole represented 1 and no hole represented 0.

Binary place value tables

A place value table allows a decimal number to be converted to its binary equivalent. When converting computer data, decimal numbers are referred to as **denary** numbers. You will learn more about converting between denary and binary numbers later in this unit (see pages 88–91).

This is a 4-bit place value table, which means it can show numbers with four binary digits. The highest denary number a 4-bit binary sequence can represent is 15.

Denary number (Decimal)	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

When a computer represents 4-bit data, each number is created by turning on (1) or off (0) the binary switch that represents each of the values 8, 4, 2 and 1. Look at these two examples of equivalent binary and denary numbers.

Binary: 0011
Denary: $0 \times 8 + 0 \times 4 + 1 \times 2 + 1 \times 1 = 3$

Binary: 1011
Denary: $8 \times 1 + 0 \times 4 + 1 \times 2 + 1 \times 1 = 11$

Further investigation

- Experiment with creating your own image grids like the ones shown in this lesson.
- Investigate the meaning of the term 'denary'.

Success criteria

- I can explain how computers use binary numbers.
- I am able to create simple 1-bit binary graphics.
- I can use a 4-bit binary place value table.

Success criteria

What has been understood from the lesson.

We hope you will find this book useful in developing your knowledge of digital technology, its effective use of applications and in supporting future learning.

Each topic includes easy to understand theory, real-world examples, and ideas for further investigation. You can also test your knowledge of keywords and regular exam-quality questions with supported answers. A checkpoint at the end of each lesson is a quick and easy way to check your own understanding.

Assessment pages

Example of exam question and answer midway and end of unit to help with exam preparation.

Unit 6: Programming Part 2 Mid-unit assessment

Unit 6 Mid-unit assessment

Typical 4-mark exam question

You have been asked to help build a new school records system for storing data about all the students in the school. You are going to be working with a range of data types.

Explain what is meant by the term data type and state at least two examples of data types.



Specimen 4-mark answer

Data types are used to identify any information stored in a program. If the data type is not correctly identified, the data cannot be processed by the computer.

Two examples of data types are whole number integers and Boolean.

Unit 6: Programming Part 2 End-of-unit assessment

End-of-unit assessment

What good things can we see in this answer?

1. The answer includes the key terms: joined, string, processed and variable.
2. The process of concatenation is clearly described.
3. An example of concatenation has been provided: joining together a first and second name.

Which parts of the answer could be better?

1. The first sentence defines concatenation as joining two strings, but two or more strings can be joined.
2. The answer does not give a reason for joining the two names together.

How can we improve this answer?

1. Improve the first paragraph by stating that two or more strings can be joined together.
2. Give a reason for joining together the first and second names. For example, the full name could be displayed on screen or added to a printed report.

Analysis

Evaluation of example answer at mid-unit assessment and end-of-unit assessment to hone analytical skills and provide useful guidance.

End-of-unit checklist

Checklist at the end of every unit to quickly assess your understanding and progress.

Unit 5: Programming Part 1 End-of-unit assessment

End-of-unit checklist

What good things can we see in this answer?

1. The answer includes some key terms: rules, function and spelling.
2. There is a clear description of what a syntax error is, and an explanation that it prevents the program from running.
3. Two tips have been given, as required.

Which parts of the answer could be better?

1. It would be helpful to include examples of spelling or character mistakes in the first paragraph.
2. The second sentence is a little vague. It is not clear what 'a particular part' refers to.

How can we improve this answer?

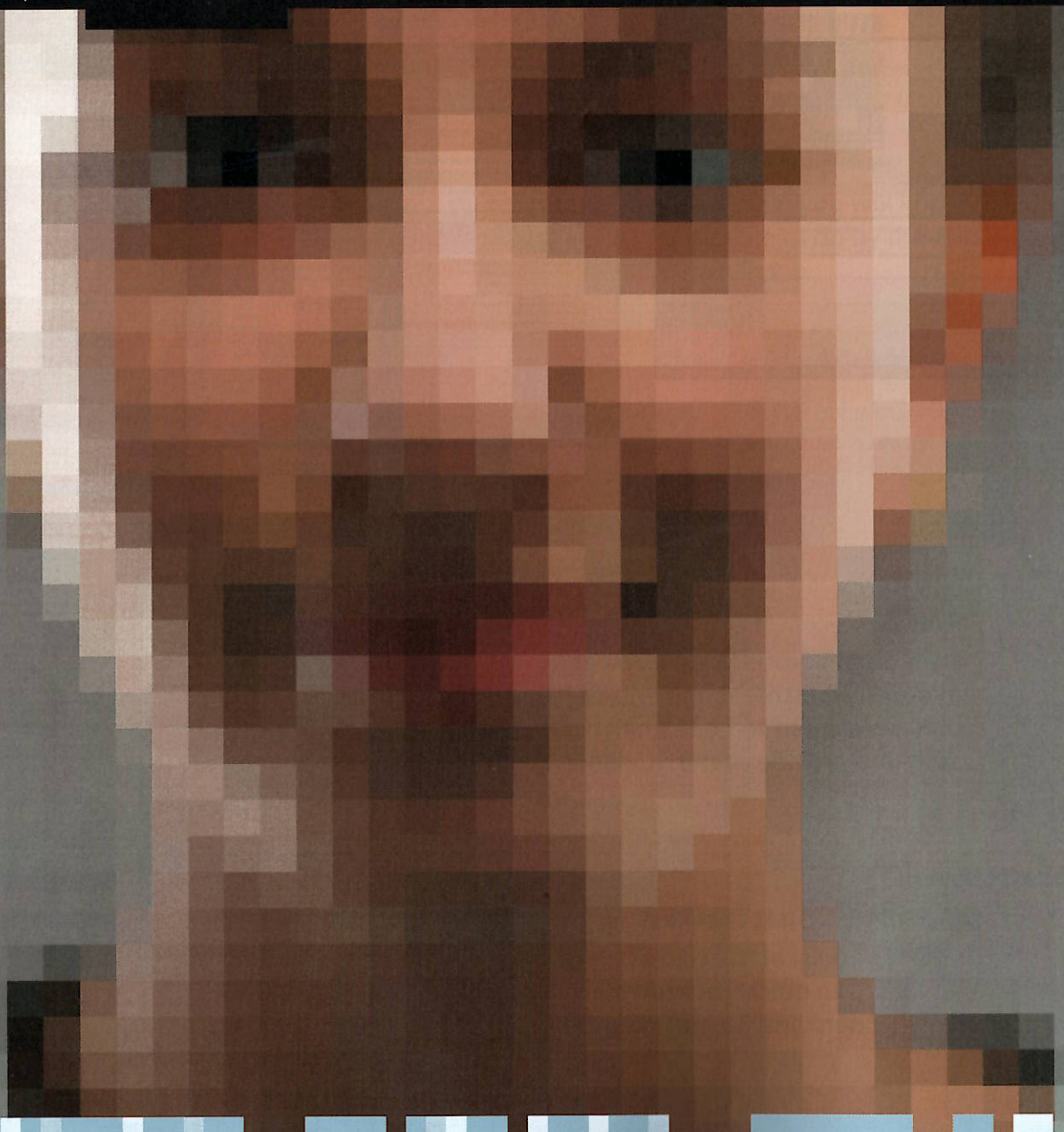
1. Improve the first paragraph by giving examples of syntax errors, such as misspelling the function print or missing out a bracket.
2. In the second sentence, explain that an incorrect function will halt the program at that point until the error is resolved.
3. Alternative tips for finding syntax errors could include checking the case and checking the layout of functions.

End-of-unit checklist

- ☐ I know what an algorithm and a computer program are.
- ☐ I know that there are different styles of programming and many different programming languages.
- ☐ I know the purpose of text-based and visual programming languages and why pseudocode is used in planning programs.
- ☐ I know what BIDMAS is and why it is used.
- ☐ I can create simple programs in the Python programming language.
- ☐ I can create simple sorts and use arithmetic and relational operators in a simple program.
- ☐ I can search a simple database using programming terms.
- ☐ I can describe the purpose of computer models and simulations and their advantages and disadvantages.
- ☐ I can use sequences, selection and iteration in a simple Python program.
- ☐ I know what a syntax error is and can give some examples.
- ☐ I know how to find and fix syntax errors.
- ☐ I know what a subprogram is and I can use pre-existing subprograms in a simple program.

Unit 1

Binary and online graphics



You are already familiar with the use of multimedia elements in your classwork, for example a presentation can contain text, images, sound, and video. But how do computers store them? You will learn about how these files are created, saved, and processed and how binary isn't just a short sequence of zeros and ones.

Key objectives:

1. Learn about vector graphics and how they are created.
2. Learn about bitmap images and how they are created.
3. Learn about the properties of vector and bitmap images and the differences between them.
4. Be able to create simple vector and bitmap graphics.
5. Learn about the link between binary and pixel-based images.
6. Learn about how colour is displayed in binary.

By the end of the unit you will:

- be able to create simple vector graphics
- be able to create simple bitmap images
- be able to recommend either vector or bitmap graphics for a given scenario
- convert a bitmap into binary and back again.

Vector graphics

Related topics

- Algorithms and programming
- Creating graphics for a target audience

Key words

coordinates
file size
geometric shapes

image quality

scale down

scale up

vector

Learning objectives

1. Understand the nature and purpose of a vector image.
2. Understand how a vector image can be created using coordinates.
3. Know how and why vectors are used for specific situations.

What are vectors?

A **vector** is a type of image, created on a computer and used in graphic design all around the world. Consider a simple dot-to-dot image where the image is generated by following a series of coordinates (x and y positions from a fixed point) from start to finish. A vector image follows the same principle and requires the following instructions. These instructions form a simple algorithm.

1. A starting point
2. An end point
3. The type of line
4. Any curves or angles
5. Line and fill colours.

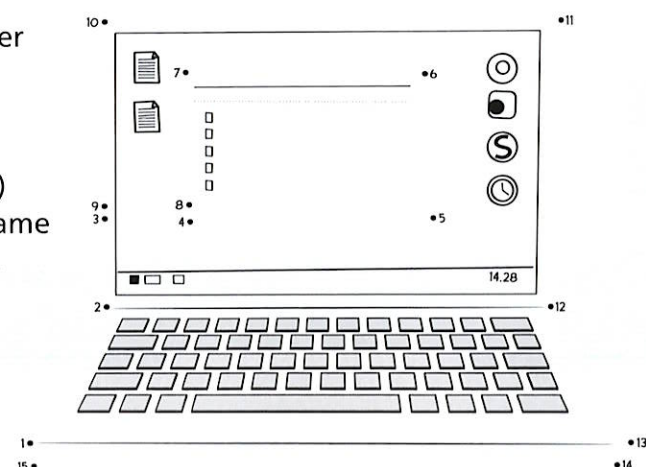
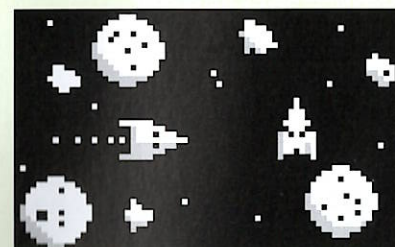
Vector graphics have some key benefits over other types of graphic. These include:

- Ability to be scaled without losing **image quality**
- Small file size
- Easily edited
- Ideally suited for **geometric shapes**.

Real-world examples

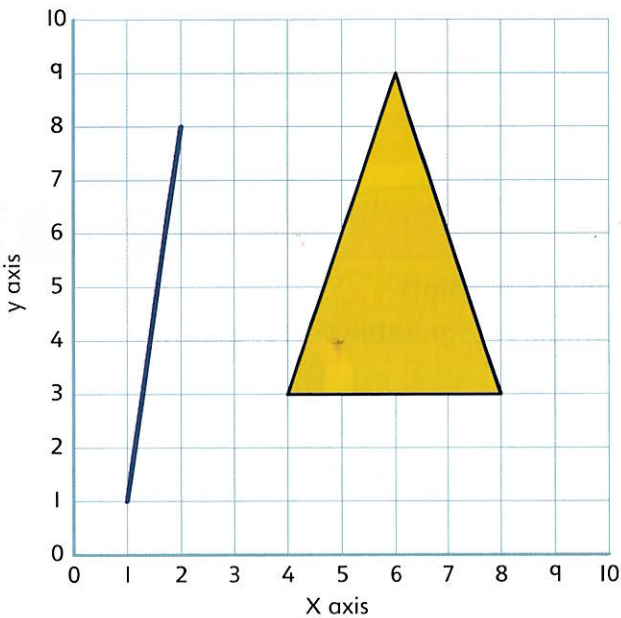
Vectors are used by graphic artists in many industries around the world. These include:

- 2D and 3D animation, from television shows to movies.
- Gaming – many of the early games were purely vector-based, however they are still used within modern console games.
- Advertising, from logos to cartoon characters.



Creating a vector graphic

Below is an example of how to create two simple vector graphics using a series of simple instructions. By plotting coordinates and then joining the dots, a shape is made.



Graphic 1 – Blue line

Start: X1,Y1

End: X2,Y8

Colour: Blue

Graphic 2 – Yellow Triangle

Start: X4,Y3

Mid: X6,Y9

End: X8,Y3

Fill Colour: Yellow

Scaling a vector graphic

One of the key benefits of a vector graphic is that it can be scaled to any size, big or small, without ever losing image quality.

Graphic 3 – Yellow rectangle

Start: X1,Y1

End: X2,Y3

Graphic 4 – Yellow rectangle x 2

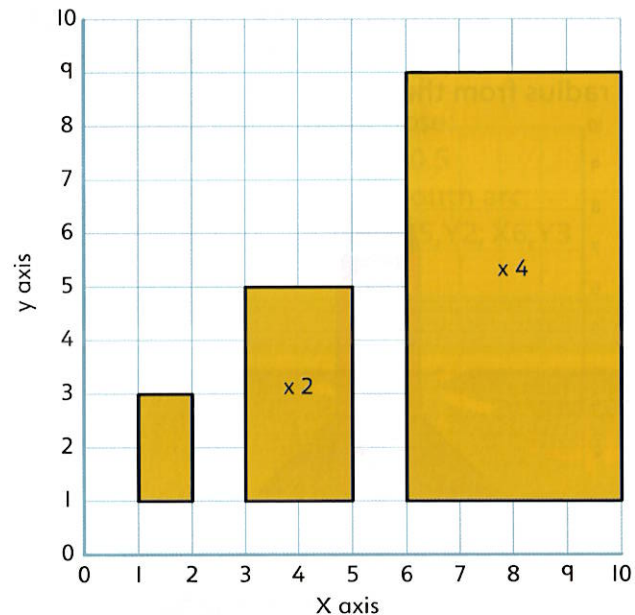
Start: X3,Y1

End: X5,Y5

Graphic 5 – Yellow rectangle x 4

Start: X6,Y1

End: X10,Y9



Further investigation

- Create some similar shapes of your own - **scale up** or **scale down**.
- Investigate further uses of vector graphics in industry.

★ Success criteria

- I know what a vector graphic is.
- I know some of the benefits of vector graphics.
- I know how a simple graphic can be created using coordinates.
- I can scale a very simple vector graphic.

Creating a vector graphic

Related topics

- Algorithms and programming
- Creating graphics for a target audience

Key words

algorithm
coordinates

image quality

scaling

vector

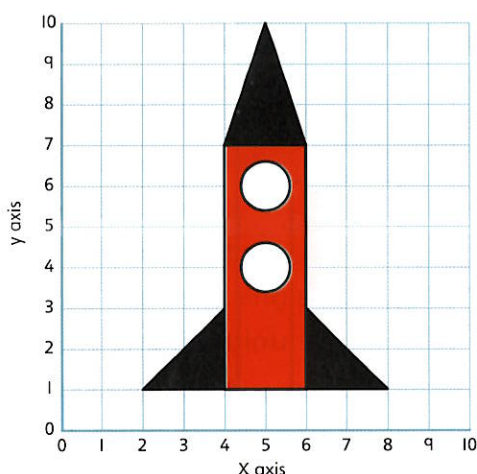
Learning objectives

- Understand how a graphic can be created using multiple points.
- Demonstrate understanding of how to create a simple recognisable vector graphic.

Creating graphics using an algorithm

An **algorithm** is simply a set of instructions so a **vector** graphic can be plotted using a series of **coordinates** in a similar style to a dot-to-dot image.

Each line of instructions in the graphics shown creates a different shape. In the previous examples, each shape had a start and finish point. In these examples, multiple points are plotted, to create a triangle, for example. Circles are also created using the letter R to represent a radius from the XY centre.



Graphic 1: Rocket

Black triangle left:

X2,Y1; X4,Y3; X4,Y1

Black triangle right:

X6,Y1; X6,Y3; X8,Y1

Red rectangle:

X4,Y1; X4,Y7; X6,Y7; X6,Y1

Black nose cone:

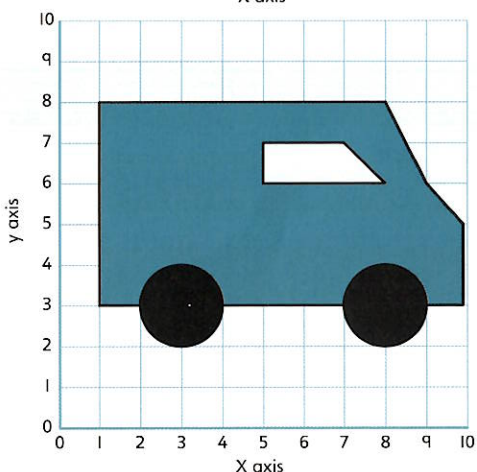
X4,Y7; X5,Y10; X6,Y7

White window 1:

X5,Y6,R0.5

White window 2:

X5,Y4,R0.5



Graphic 2: Car

Blue car body:

X1,Y3; X1,Y8; X8,Y8

X9,Y6; X10,Y5; X10,Y3

White window:

X5,Y6; X5,Y7; X7,Y7; X8,Y6

Black wheel 1:

X3,Y3,R1

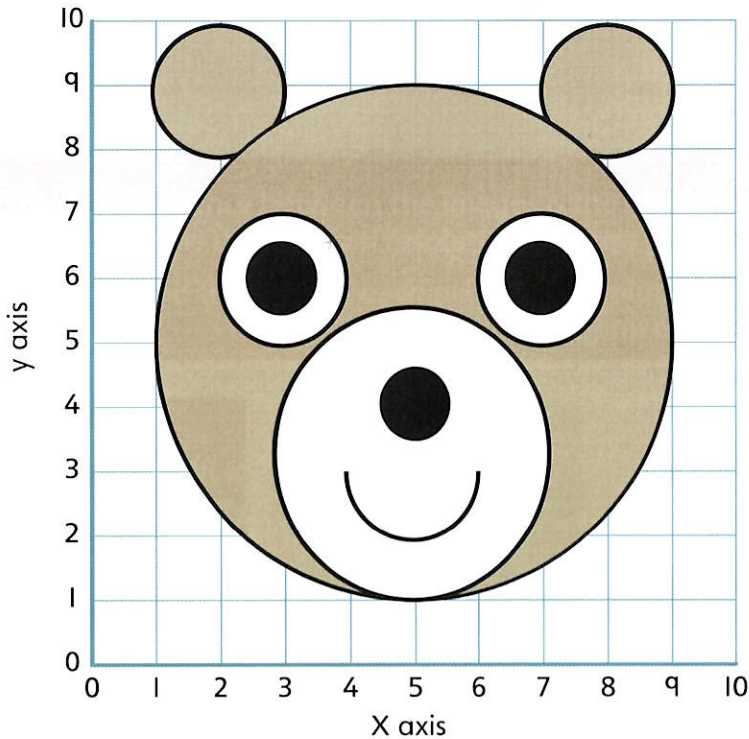
Black wheel 2:

X8,Y3,R1

Real-world examples

The Scratch visual coding app has a pen tool that can be used to create simple vector graphics using coordinates in a similar style to this page.

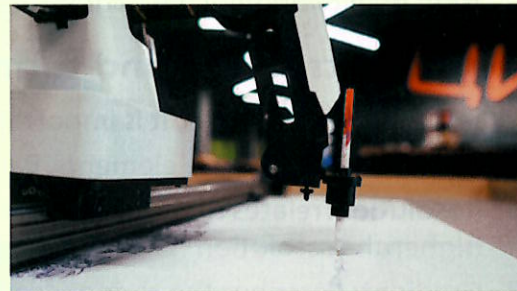
Graphic 3: Animal



Brown head:
X5,Y5,R4
Brown left ear:
X2,Y9,R1
Brown right ear:
X8,Y9,R1
White left eye:
X3,Y6,R1
White right eye:
X7,Y6,R1
Black left eye:
X3,Y6,R0.5
Black right eye:
X7,Y6,R0.5
White snout:
X5,Y2,R2
Black nose:
X5,Y4,R0.5
Black mouth arc:
X4,Y3; X5,Y2; X6,Y3

Real-world examples

A graphics plotter is a device that follows a series of instructions to create a graphic, for example architectural drawings. The pen is lowered onto the paper as required and lifted when it needs to start another shape.



Further investigation

- Investigate software packages designed to create vector graphics.
- Experiment with creating shapes of your own choice.

Success criteria

- I know how shapes can be created using a set of instructions.
- I can create a graphic of multiple parts using a set of instructions.

Bitmaps

Related topics

- Digital images
- Creating graphics for a target audience

Key words

bitmap
file size
gradient

image quality

pixel

pixels per inch (PPI)

resolution

Learning objectives

1. Understanding of what a bitmap graphic is.
2. Understand the basic structure of a bitmap.
3. Understand the link between resolution and image quality

What is a bitmap?

A **bitmap** is an image file made up of dots, or pixels. Each **pixel** can be a different colour. The greater the number of pixels, the more detailed the image.

These two images both show a pineapple. The one on the left is a vector; the one on the right is a bitmap.



Bitmaps can be used for:

1. Digital photographs, either taken with a digital camera or scanned using a scanner.
2. High quality original game graphics.

Bitmap structure and terminology

When exploring bitmaps, it is important to understand the following key terms:

1. **Pixel:** short for picture elements. Each pixel is a tiny coloured square that makes up an image.
2. **Resolution:** relates to the number of pixels within the image. The more detailed the image, the higher the resolution is required.
3. **Gradient:** describes the change from one colour to another, fading from black to white for example with the pixels moving from black to grey to white.

Real-world examples

Bitmaps are often referred to as raster images. Common bitmap filetypes include the following: BMP, JPG, GIF, TIFF, PNG and PICT.



Resolution and image quality

Consider the images shown. Each pair show the same image but one is clearly sharper than the other.



The key difference between the images is the resolution – the number of pixels used to create the image. The unit of pixel resolution is **pixels per inch (PPI)**.

The image on the right of each pair has a resolution of 300 PPI and the image on the left is 10 PPI.

The **file size** of the images is related to the resolution – the higher the resolution, the larger the file size.

This means that increasing the physical size of any bitmap will reduce the resolution, lowering the **image quality**.

📌 Real-world examples



Games consoles are often compared based on their technical specifications. At the time of writing, the most recent Xbox and PlayStation consoles can play games at an 8K resolution of 7680×4320 pixels. This means controlling over 33 millions pixels at once!

🔍 Further investigation

- Open some images using photo-editing software and experiment by changing the resolution.
- Investigate the number of pixels a smartphone or tablet can display.

★ Success criteria

- I know what a bitmap is.
- I know how a bitmap is made.
- I can describe the link between resolution and image quality.

Comparing vectors and bitmaps

Related topics

- Creating graphics for a target audience
- Appropriate ways of working

Key words

Bitmap
file size
image quality

pixel
resolution
scaling
vector

Learning objectives

1. Understand the key differences between vectors and bitmaps.
2. Consider the advantages and disadvantages of vectors and bitmaps in a given situation.

Similarities and differences

When comparing similar technology or working methods, it is important to have a list of criteria. For example:

- **file size:** how much storage space will the image require?
- **image quality:** what is the level of detail shown?
- **scaling:** can the image be increased or reduced in size so that the image is still recognisable?
- **use of colour:** how many colours can be displayed?



Vectors vs Bitmaps

Let's compare these two image types against some criteria.

Criteria	Vectors	Bitmaps
File size	Very small file sizes compared to a bitmap and it is not size- or resolution-dependent.	The file size increases as the resolution , size of image and number of colours increases.
Image quality	At any distance, size or magnification, a vector image will maintain the same level of quality but is not suitable for high resolution, realistic images.	Bitmaps are ideal for showing high-quality photos as every pixel can display a different colour.
Scaling	A vector can be scaled to any size, large or small, because it is based on mathematical data.	Images can be made smaller in size but if made bigger there will be a reduction in image quality as the pixels become further apart.
Use of colour	Lines, shapes and areas can be set to any colour but vector graphics cannot easily create gradients.	Bitmaps can display millions of colours and detailed gradients.

Real-world examples

Graphic designers will use both vectors and bitmaps in their work and there are popular software applications for both. The Adobe Creative Suite, for example, includes Photoshop for bitmap editing and Illustrator for creating vector graphics.



Real-life scenarios

Let's consider the following scenarios and for each make a choice about whether it would be better to use bitmaps or vectors as the main image format. We will use the comparison table for each scenario.

Scenario 1: A logo

A designer is creating a new logo to promote a fairtrade chocolate. To be used on a variety of different sized packages, the logo needs to use simple shapes, and be eye-catching.



Scenario 2: A fashion magazine

An editor must select a range of digital photographs every month to be added to the magazine. Each photograph must be crystal clear when printed so the fine detail of the clothing can be seen.



Scenario 3: An animated character

A popular online animated series requires a brand new character. The style of the series uses thick, strong lines and bright vibrant backgrounds. This character will need to be edited and animated regularly.



Scenario 4: A movie poster

A designer has been asked to create a movie poster for a new movie. It will feature photo-realistic images of the main characters and a fantasy background. It will be used in a range of online and printed locations.



Real-world advice

When an image is printed in a commercial publication, such as a magazine, it is recommended that the image has a resolution of at least **300 dots per inch (DPI)** or PPI. Any lower than this and the image can appear blurry.

Further investigation

- Investigate some popular graphic art that you like and decide whether it was created using vectors, bitmaps or a mixture of both.
- Investigate the vector and bitmap editing software that you have access to at school.

Success criteria

- I know some of the key differences between bitmaps and vectors.
- I can decide which graphic would be the most appropriate for a given situation.

Binary and pixels

Related topics

- Creating graphics for a target audience
- Computer systems

Key words

binary

binary word

bitmap

code

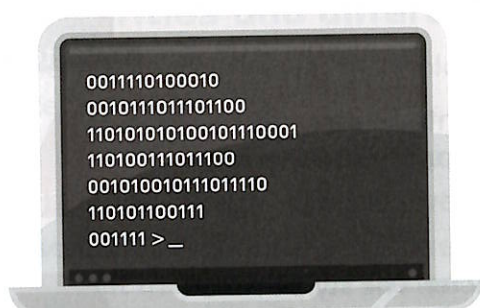
pixel

resolution

Learning objectives

1. Understand how binary is used to represent pixels.
2. Demonstrate how to create a simple graphic using a binary sequence.

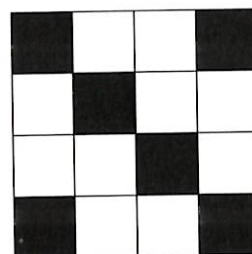
Binary transmission



As explored in Year 7, all computer data must be converted to **binary** in order to be understood. Binary is the language of computers and contains only two characters, 1 and 0.

Binary and pixels

Consider a simple black and white **bitmap** image with a low **resolution**. Each **pixel** can be represented in binary with either a 1 (black) or a 0 (white).

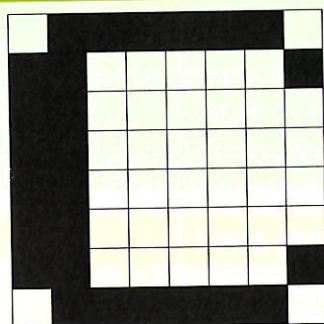


1	0	0	1
0	1	0	0
0	0	1	0
1	0	0	1

What is a binary word?

A collection of 1s and 0s groups together for a purpose is known as a **binary word**. For example, the simple bitmap image above would have the following binary word: 1001010000101001.

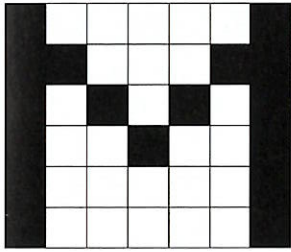
Real-world examples



Early computer typefaces were created as bitmaps, allowing for elaborate designs. However, they were only available in certain sizes and could not be resized without losing quality.

Simple binary graphics

Consider the following example of a simple character created in binary. It is shown as a bitmap image, a binary image and a binary sequence.

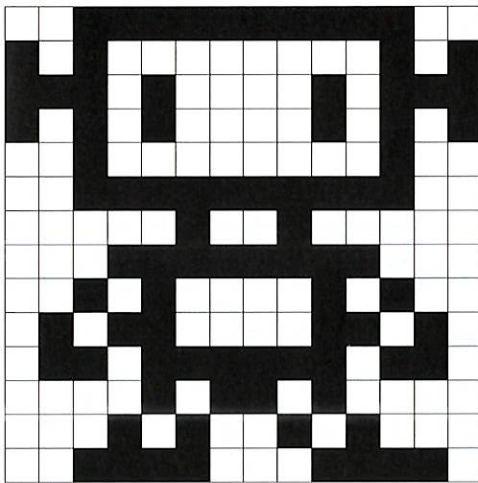


1	0	0	0	0	0	1
1	1	0	0	0	1	1
1	0	1	0	1	0	1
1	0	0	1	0	0	1
1	0	0	0	0	0	1
1	0	0	0	0	0	1
1	0	0	0	0	0	1

10000011100011101010101100100110000011000001

High-resolution binary graphics

Consider the following example of a simple character created in binary. It is shown as a bitmap image, a binary image and a binary sequence.



0	0	1	1	1	1	1	1	1	1	1	1	0	0
1	0	1	0	0	0	0	0	0	0	0	1	0	1
1	1	1	0	1	0	0	0	0	1	0	1	1	1
1	0	1	0	1	0	0	0	0	1	0	1	0	1
0	0	1	0	0	0	0	0	0	0	0	1	0	0
0	0	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	0	1	0	0	1	0	0	0	0	0
0	0	0	1	1	1	1	1	1	1	1	0	0	0
0	0	1	0	1	0	0	0	0	1	0	1	0	0
0	1	0	1	1	0	0	0	0	1	1	0	1	0
0	1	1	0	1	1	1	1	1	1	0	1	1	0
0	0	0	0	1	0	1	1	0	1	0	0	0	0
0	0	0	1	0	1	0	0	1	0	1	0	0	0
0	1	1	1	1	0	0	0	0	1	1	1	1	0

00111111111100101000000001011110100001011110101000010101001000000001000011
 111111111000000010010000000011111111000001010000101000101100001101001101111
 110110000010110100000001010010100001111000011110

Further investigation

- Try creating your own 1-bit art in a variety of sizes.
- Investigate the history of computer typefaces and their design.

★ Success criteria

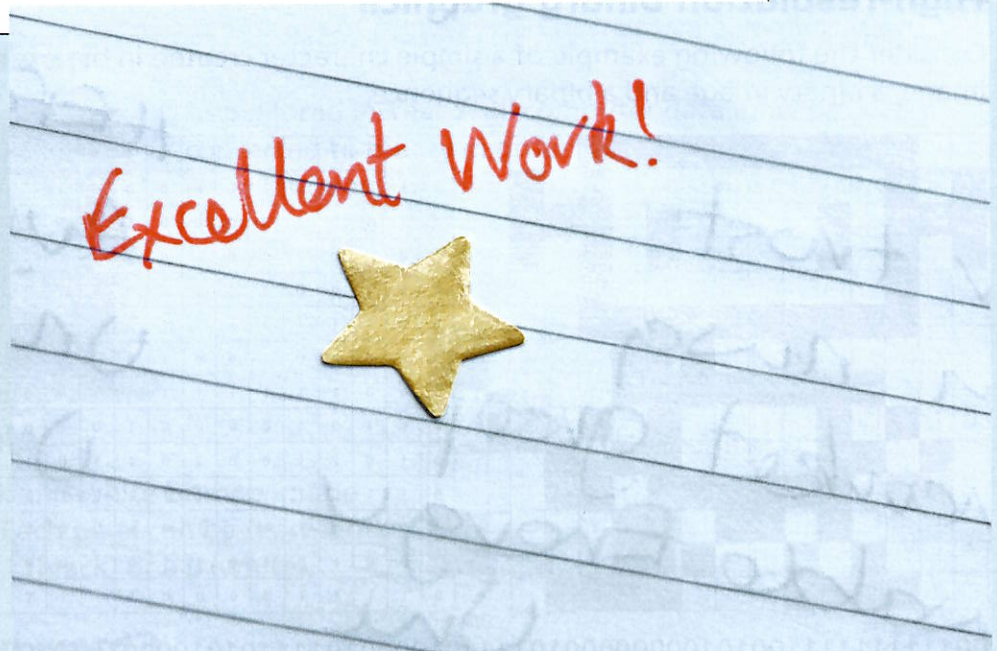
- I know how binary is used to represent pixels.
- I can create a simple pixel bitmap and show the binary sequence that created it.

Unit 1 Mid-unit assessment

Typical 4-mark exam question

You have been asked to create a range of award stickers for an online educational application. The graphics will be used within the application and on websites and possibly physical merchandise.

State whether you would use bitmap or vector-based software to create them and describe three reasons for your choice.



Typical 4-mark answer

The most appropriate choice would be vector graphics software. This is because it can be resized without losing image quality. The online file sizes would be small and multiple versions of the graphics could be easily created and edited as required.



What good things can we see in this answer?

1. Appropriate key terms used: image quality, file size, edited.
2. Correct choice of vector software.
3. Provides three reasons: resizing without losing image quality, small files, multiple edits.



Which parts of the answer could be better?

1. No reasons given for choices.
2. Examples of editing could be provided.



How can we improve this answer?

1. Explain that the reason it can be resized and maintain small file size is due to the mathematical structure of vectors.
2. Give examples of multiple versions, for example the text in the graphics can be easily edited or the shapes used to create the graphic can be easily adjusted.

Colour depth and binary representation

Related topics

- Creating graphics for a target audience
- Computer systems

Key words

binary

binary word

Bitmap

bit

colour depth

represent

Learning objectives

1. Understand the purpose of colour depth in digital images.
2. Understand how colour depth is represented in binary.
3. Demonstrate how to represent simple colour images in binary.

What is colour depth?

Colour depth is the number of colours an image can contain, represented by the number of **bits** per pixel. The greater the number of bits, the more colours can be shown in that individual pixel. Remember a 1-bit pixel can only be 1 or 0, therefore it can only be black and white.

Number of bits per pixel	Number of possible colours
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256



2 colours



8 colours



32 colours

Real-world examples

The original Nintendo Gameboy, released in 1989, had a 2-bit display. This means it could display four colours, from light to very dark olive green.



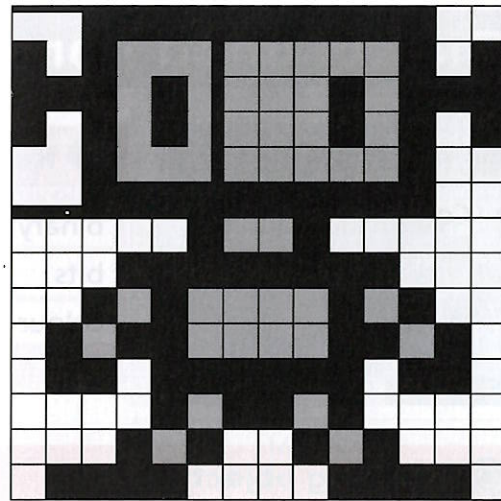
Representing 2-bit graphics

Let's look again at the graphic from the previous lesson and this time **represent** it as a 2-bit graphic. A 2-bit graphic can show four colours as each colour can be assigned a unique **binary word**. For example:

Binary		Colour
0	0	White
0	1	Light grey
1	0	Dark grey
1	1	Black

Here is the first 6x6 grid from the top left of the image in binary.

Now that it is a 2-bit graphic, the image looks very different. However, as every pixel can contain one of four colours, the binary word is twice as long!



01	01	11	11	11	11
11	01	11	10	10	10
11	11	11	10	10	10
11	01	11	10	10	10
01	01	11	10	10	10
01	01	11	11	11	11

Real-world examples

The majority of modern computers and smartphones have a 24-bit display. Known as True Colour, a 24-bit display can show 16,777,216 possible colours.



Further investigation

- Complete the binary sequence of the 2-bit graphic in this lesson.
- Investigate how many colours the human eye can actually see.

Success criteria

- I know what the term colour depth means.
- I know how colour depth can be shown in binary.
- I can create a simple 2-bit, 4-colour image using a binary reference.

Digital image properties

Related topics

- Computer systems

Key words

binary

bits

colour depth

dimension

file size

pixels

pixel size

resolution

Learning objectives

1. Understand the key properties of a digital image.
2. Understand the link between pixel dimensions and file size.
3. Be able to calculate the file size of a simple image.

Key image properties

Let's consider a simple image and list all the ways in which it can be defined:

- **Pixel size:** how many **pixels** in total make up the **image**?
- **Dimension:** how many pixels high and wide is the image?
- **Resolution:** how many pixels are there in one square inch of the image?
- **Colour depth:** how many **bits** are used to encode the colour?

How to calculate the file size in bits

The following information about the image is required:

- The number of pixels (height x width)
- The colour depth in bits

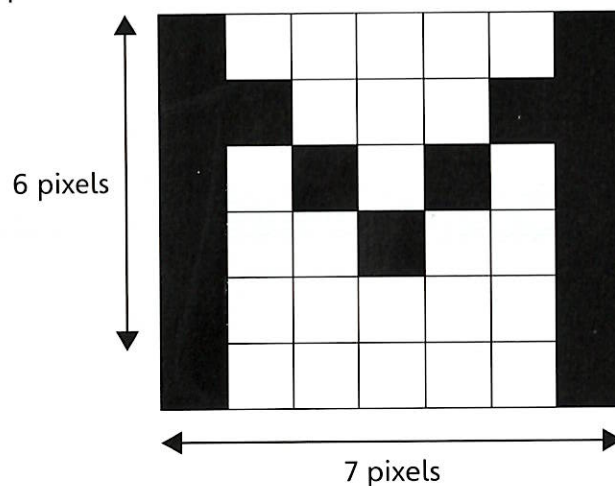
This simple 1-bit letter graphic would be:

File size = pixels × colour depth

File size = (7 × 6) × 1

File size = 42 × 1

File size = 42 bits



Examples of images and their properties

The following images each have their own properties and the file size in bits can be calculated for each.

First, let's look at the robot example from earlier in the book:

2-bit graphic = 4 colours

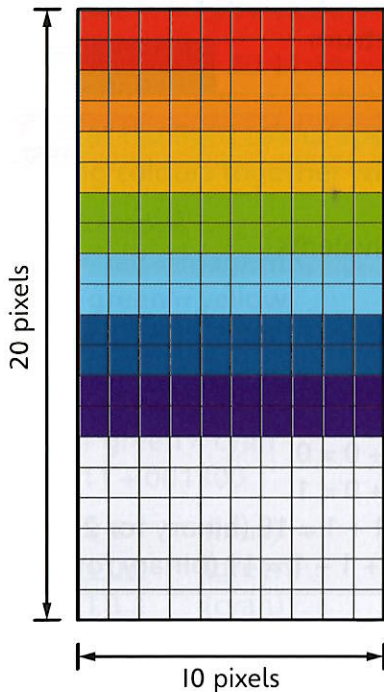
File size = pixels x colour depth

File size = $(14 \times 14) \times 2$

File size = 196×2

File size = 392 bits

What about a more colourful graphic?



3-bit graphic = 8 colours

File size = Pixels x Colour depth

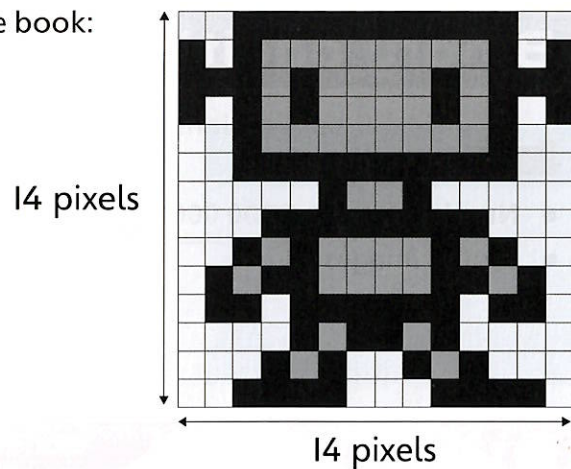
File size = $(20 \times 10) \times 3$

File size = 200×3

File size = 600 bits

Pixel dimensions and file size

As clearly shown in the examples, the larger the number of pixels, the larger the file size. This also applies to the number of colours used, as the number of pixels is multiplied by the colour depth in bits.



Real-world example

Image-sharing websites around the world often offer multiple download options for the same image. This allows the user to download either a low-resolution, standard or high-resolution version of an image, depending on the user's need. For example: a small social media post image will normally be viewed on a mobile device and needs to load quickly.

Further investigation

- Look back at pixel images you have created already and calculate their file size in bits.
- Experiment with creating your own graphics with a limited number of colours.

Success criteria

- I know how to describe an image using key properties.
- I know how the number of pixels increases the file size of an image.
- I can calculate the file size in bits of a simple graphic.

Binary addition

Related topics

- Number bases
- Digital images

Key words

binary

column addition

cyan

denary

hexadecimal (HEX)

magenta

overflow

spectrum

Learning objectives

1. Understand how to add two binary words together.
2. Demonstrate how to add colours together using their binary equivalents.

Binary addition

Adding two **binary** words together is a straightforward process if you use the following rules.

Let's add the two binary numbers:

10101 + 00011

Add the columns, working from the right and carrying the extra value to the left. This method is called **column addition**.

1	0	1	0	1	(Equal to 21)
0	0	0	1	1	(Equal to 3)
1	1	0	0	0	(Equal to 24)
	1	1	1		

Rule One: $0 + 0 = 0$

Rule Two: $1 + 0 = 1$

Rule Three: $1 + 1 = 10$ (binary for 2)

Rule Four: $1 + 1 + 1 = 11$ (binary for 3)

A longer example

11011011 + 00100100

1	1	0	1	1	0	1	1	(Equal to 219)
0	0	1	0	0	1	0	0	(Equal to 36)
1	1	1	1	1	1	1	1	(Equal to 255)

In the example above, the total comes to 255. As discussed in Unit 4 of the Year 7 student book, the maximum value an 8-bit binary sequence can represent is 256 (0–255, including 0). If an 8-bit binary addition is carried out that exceeds this value, the sum cannot be carried out. This is called an **overflow**, and in a real computer system, calculating a value outside of the defined range will result in the computer crashing!

Adding binary colours

The method for adding binary words can also be used for adding colours together. Let's consider the following colours and assign a 6-bit binary word to each.

Colour	6 bit binary
Red	11 00 00
Green	00 11 00
Blue	00 00 11
Magenta	11 00 11
Yellow	11 11 00
Cyan	00 11 11

Please note:

The actual binary word for red is stored in 24-bit binary but it has been simplified for the examples.

Red: 1111 1111 0000 0000 0000 0000

Look at the light colour **spectrum** on the right. It shows how adding colours together will result in a different colour.

For example:

blue + red = **magenta**

red + green = **yellow**

blue + green = **cyan**

This can be confirmed using binary addition:

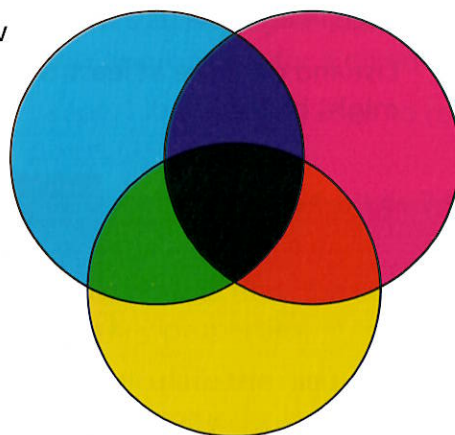
blue + green = cyan

000011 + 001100

0 0 0 0 1 1 (blue)

0 0 1 1 0 0 (green)

0 0 1 1 1 1 (cyan)



Real-world examples

Colours are normally stored on computers in a hexadecimal format.

Hexadecimal (HEX) is a base-16 system and includes letters and numbers.

The 24-bit binary word is converted to six 4-bit words that are each converted to a hexadecimal. For example, red is FF0000.



Further investigation

- Apply the same calculations to the other colours in the colour spectrum diagram.
- Read about the hexadecimal number base.
- Investigate the term 'overflow' in binary.

Success criteria

- I know how to add simple binary words together.
- I can add together colours using their associated binary value.

Typical 4-mark exam question

Each image lists its associated properties on the site.

An illustration of a photographer's workspace. A white laptop is open, displaying a photo of a woman with sunglasses. To the left, a tablet shows a beach scene. A hand is on the laptop trackpad, and another hand is on a drawing tablet with a stylus. Surrounding these are various photography accessories: a DSLR camera, a lens, a lens cap, a pencil, a stack of papers, a wallet, a camera strap, and a pair of sunglasses. The background is a solid orange color.

The site could include the following:

Pixel dimensions: the width and height in pixels of the image.

File size: the calculation in bits of the image.

Colour depth: the number of different colours that make up the image.

**What good things can we see in this answer?**

1. Some key terms used: pixels, width/height, colour depth, file size, bits.
2. More than two examples included.

**Which parts of the answer could be better?**

1. Reference to pixels could be made in colour depth.
2. An example of colour depth could be given.

**How can we improve this answer?**

1. Describe how each pixel can represent the number of colours allowed.
2. Include an example of 1-bit having 2 colours or 8-bit having 256 colours.
3. Include an additional answer, describing image resolution.

**End-of-unit checklist**

- ☐ I know what a vector graphic is.
- ☐ I know how to create a simple graphic using coordinates.
- ☐ I know what a bitmap is and how it is structured.
- ☐ I know some of the key differences between bitmaps and vectors.
- ☐ I know which graphic would be the most appropriate for a given situation.
- ☐ I know how binary is used to represent pixels.
- ☐ I can create a simple 1-bit or 2-bit image.
- ☐ I know how to describe an image using key properties.
- ☐ I can calculate the file size in bits of a simple graphic.
- ☐ I know how to add simple binary words together.

Unit 2

Data on the web

It makes sense, the bigger the file – the more zeros and ones are needed to represent it in binary. Understanding how data is stored, processed, and even compressed will help you make informed choices when programming computers and working with network communication.

Key objectives:

1. Learn about the structure of binary data.
2. Learn about storage capacity.
3. Understand the key differences between binary and denary.
4. Calculate file sizes in base 2 and base 10.
5. Learn about internet bandwidth and its impact on networks.
6. Learn about the purpose and different types of compression.

By the end of the unit you will:

- be able to convert binary to denary and back again
- create visual and text-based calculator programs
- calculate numbers to the power of another and file transfer speeds
- apply compression methods to different scenarios
- be able to apply run length encoding to a simple graphic.

Bits, bytes and nibbles

Related topics

- Computer systems
- Creating original graphics

Keywords

binary

bit

byte

nibble

pixel

storage

Learning objectives

1. Understand the purpose of a storage device.
2. Understand the terms bits, bytes, and nibbles in terms of storage capacity.
3. Demonstrate an ability to calculate the file size of simple graphics.

What is a storage device?

A storage device stores computer information. Examples include:

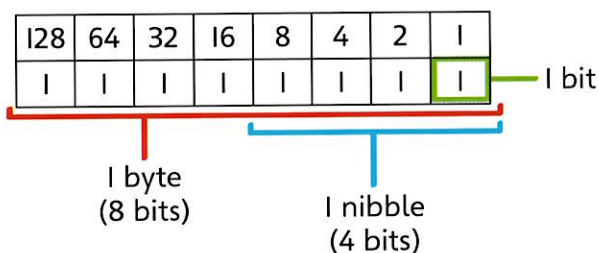
- Internal hard drive: the non-removable storage part of every computer.
- Flash drive (USB): used to transport information from one computer to another.
- Memory cards: used to add storage to devices such as smartphones and digital cameras.
- External hard drive: used to add additional large-scale storage to a computer system.



Bits, bytes, and nibbles

As covered in Year 7, a **bit** is a single binary digit, or one of eight digits in an 8-bit **binary** sequence. We need terms to describe larger groups of bits, two of which are **nibbles** and **bytes**.

Let's consider the value 255 in binary (11111111):



Real-world examples

Ever wonder where the 'Save' symbol originated from on your computer? It was inspired by a floppy disk! Created in the 1980s, it is one the oldest examples of **storage** devices, capable of storing 1.44 megabytes of data. Although the disks are no longer manufactured, you can still buy floppy disk readers for home computers.



Calculating text file sizes

Let's consider the word sizes in the following table. Based on the rules we have discussed, how much data storage would each one require in bytes?

Word size	Bytes	Nibbles	Bits
boat	4	8	32
Adventure	9	18	72
Environment	11	22	88
extraordinary	13	26	104

Calculating image file sizes

We can apply the same rules to simple pixel-based images we created in Unit 1.

2-bit, 4-colour robot

File size in bytes = (pixels × colour depth)/8

$$\begin{aligned}
 &= ((14 \times 14) \times 2)/8 \\
 &= (196 \times 2)/8 \\
 &= 392/8
 \end{aligned}$$

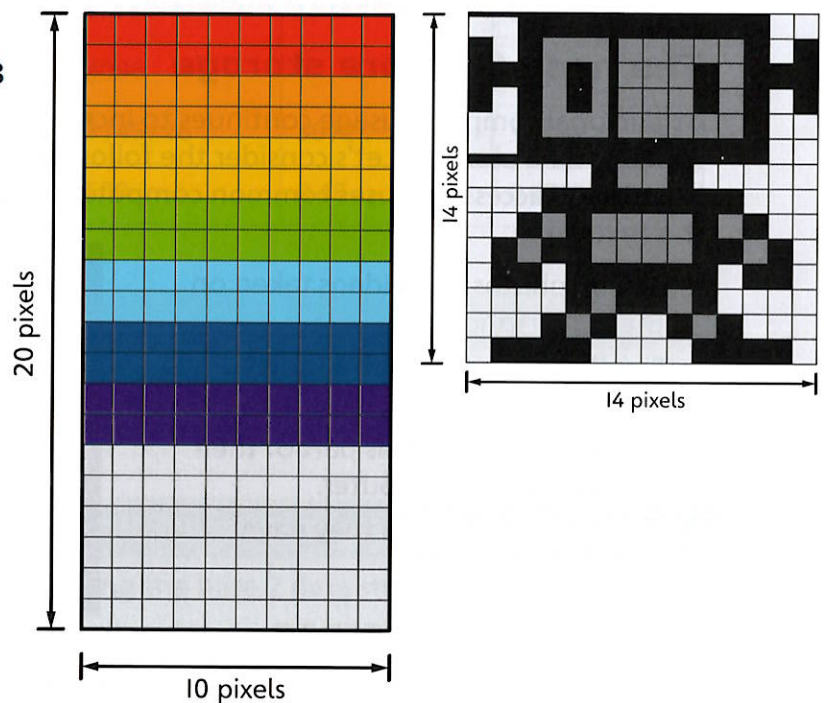
File size = 49 bytes

3-bit, 8-colour rainbow

File size in bytes = (pixels × colour depth)/8

$$\begin{aligned}
 &= ((20 \times 10) \times 3)/8 \\
 &= (200 \times 3)/8 \\
 &= 600/8
 \end{aligned}$$

File size = 75 bytes



Further investigation

- Choose a selection of words and images and calculate their file size in bytes.
- Investigate common storage devices and the amount of data they can hold.
- Talk to your teacher about the storage devices used in your school.

★ Success criteria

- I know what a storage device is.
- I understand the following terms and the relationships between them: bits, bytes and nibbles.
- I can calculate the file size of simple text and graphic files.

Storage capacity

Related topics

- Computer systems
- Document types

Keywords

base 2

base 10

binary

data storage

gibibyte (GiB)

gigabyte (GB)

kibibyte (KiB)

kilobyte (KB)

mebibyte (MiB)

megabyte (MB)

Learning objectives

1. Understand the need for larger file storage capacity.
2. Define larger storage in terms of kibi-, mebi- and gibibytes.
3. Demonstrate the ability to calculate larger file sizes.

The need for more storage

As personal computer usage continues to increase globally, so does the need for large capacity **data storage**. Let's consider the following scenario: a young adult lives in a town with broadband access and uses common computer devices daily. They may need to be able to store the following:

- Digital photos and videos taken on their smartphone.
- Family emails saved on a tablet computer.
- Documents created as part of their job, saved on a computer.
- Music and video files they have downloaded.



Units of data storage

To prevent having to describe a document as having a file size in millions of bytes, the following additional **base 2** units are used. They are in base 2 so that the number of **binary** bits can be calculated.

- **1 kibibyte (KiB)** = 1024 (2^{10}) bytes
- **1 mebibyte (MiB)** = 1048576 (2^{20}) bytes, or 1024 kibibytes
- **1 gibibyte (GiB)** = 1073741824 (2^{30}) bytes or 1024 mebibytes

Base 10 units are also used to describe the number of bytes required. These terms may be more recognisable but do not allow the number of bits to be calculated, just the number of bytes.

- **1 kilobyte (KB)** = 1000 (10^3) bytes
- **1 megabyte (MB)** = 1000000 (10^6) bytes or 1000 kilobytes
- **1 gigabyte (GB)** = 1000000000 (10^9) bytes or 1000 megabytes

Real-world examples

The binary units kibibyte, mebibyte and gibibyte were first introduced by the International Electrotechnical Commission (IEC) in December 1988. Although not as accurate, the terms kilobyte, megabyte and gigabyte are more commonly used.

Typical file sizes

Commonly used file types and related file sizes.

File type	Size in bits	Size in bytes	Size in kibibytes (KiB)	Size in mebibytes (MiB)	Size in gibibytes (GiB)
Word processed document	41943040	5242880	5120	5	0.005
Digital photo	25165824	3145728	3072	3	0.003
Audio file	33554432	4194304	4096	4	0.004
Video file	25769803776	3221225472	3145728	3072	3
Vector graphic	4096000	512000	500	0.48	0.0005

Real-world advice

The highest unit of base 2 storage defined by the IEC is the Yobibyte (YiB).

This is equal to 2^{80} or 1,208,925,819,614,629,174,706,176 bytes!

Further investigation

- Double check the calculations shown above using the base 2 data storage rules.
- Investigate the names of storage levels above gibibytes.
- Compare file sizes on your school computer to the examples in the table.

Success criteria

- I know why larger storage units are required.
- I can define file sizes in bits, bytes, kibibytes, mebibytes and gibibytes.
- I can recognise typical document file sizes.

Binary and Denary

Related topics

- Computer hardware

Keywords

base 2

base 10

binary

central processing unit (CPU)

denary

place value table

power of

Learning objectives

1. Understand the key differences between base 2 and base 10.
2. Demonstrate how to present denary and binary numbers.
3. Understand how to convert binary to denary and vice-versa.
4. Demonstrate the use of place tables and indices.

Number bases

Let's recap base 2 and base 10 number bases.

Base 2: the basis of **binary**, the language spoken by all computers. It uses only two digits, 0 and 1, representing the switching on and off of one of the millions of transistors inside a **central processing unit (CPU)**.

Base 10: the basis of our decimal system. It uses ten digits, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. With the base 10 number system, we can create any possible positive or negative number.

Denary numbers

Denary, or decimal, is a **base 10** system. It can be represented using an indices **place table** like the one below, where each column contains an increasing power of 10.

Power	10^6	10^5	10^4	10^3	10^2	10^1	10^0
Description	Millions	Hundreds of thousands	Tens of thousands	Thousands	Hundreds	Tens	Ones
Example 1	0	0	0	1	9	7	7
Example 2	6	1	8	5	3	0	1

$$\begin{aligned}
 \text{Example 1} &= (0 \times 10^6) + (0 \times 10^5) + (0 \times 10^4) + (1 \times 10^3) + (9 \times 10^2) + (7 \times 10^1) + (7 \times 10^0) \\
 &= 0 + 0 + 0 + 1000 + 900 + 70 + 7 \\
 &= 1977
 \end{aligned}$$

$$\begin{aligned}
 \text{Example 2} &= (6 \times 10^6) + (1 \times 10^5) + (8 \times 10^4) + (5 \times 10^3) + (3 \times 10^2) + (0 \times 10^1) + (1 \times 10^0) \\
 &= 6000000 + 100000 + 80000 + 5000 + 300 + 0 + 1 \\
 &= 6185301
 \end{aligned}$$

Real-world examples

Ancient history has many examples of number systems being developed around the world, all based around the number ten. It seems logical to assume these systems developed from counting the digits on our hands.



Binary numbers

Binary can also be represented by an indices place table, where each column contains an increasing power of 2. However, unlike the denary table, each column can only contain either a 1 or a 0. Think of each of the columns as a switch, switching between 0 and 1. Adding a 1 in a column will include that value in the total; a 0 will not.

Power	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Description	128	64	32	16	8	4	2	1
Example 1	1	1	0	0	1	1	0	0
Example 2	0	0	1	1	1	1	0	1

$$\begin{aligned}
 \text{Example 1} &= (1 \times 2^7) + (1 \times 2^6) + (0 \times 2^5) + (0 \times 2^4) + (1 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + 0 \times 2^0 \\
 &= 128 + 64 + 0 + 0 + 8 + 4 + 0 + 0 \\
 &= 204
 \end{aligned}$$

$$\begin{aligned}
 \text{Example 2} &= (0 \times 2^7) + (0 \times 2^6) + (1 \times 2^5) + (1 \times 2^4) + (1 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + 1 \times 2^0 \\
 &= 0 + 0 + 32 + 16 + 8 + 4 + 0 + 1 \\
 &= 61
 \end{aligned}$$

Converting between denary and binary

Using a base 2 place table like the ones shown previously is the simplest way to convert between denary and binary.

Remember: an 8-bit table can only convert numbers up to a value of 255.

- To convert a denary number to binary, simply place a 1 in each of the boxes, or switches, in the table that when added together will create the required value. For example, for 46, turn on the 32, 8, 4 and 2 switches, creating 00101110.
- To convert binary to denary, place the sequence in the table and add up the columns. For example, 10000001 turns on the 128 and 1 switches, adding up to 129.

Further investigation

- Use the place tables shown to create examples and conversions of your own.
- Challenge yourself to memorise the binary sequence for a short series of denary numbers.

Success criteria

- I know the difference between base 2 and base 10.
- I know how to create base 2 and base 10 place tables using indices.
- I can use place tables to convert denary and binary numbers.

Creating a 'power of' calculator

Related topics

- Denary numbers
- Arithmetic operators
- Algorithmic thinking

Keywords

denary
input

integer

output

power of

Scratch

sprite

variables

Learning objectives

1. Understand the requirements of a simple algorithmic calculator.
2. Demonstrate an ability to create a simple visual-coding calculator.
3. Demonstrate an ability to create a simple calculator using Python.

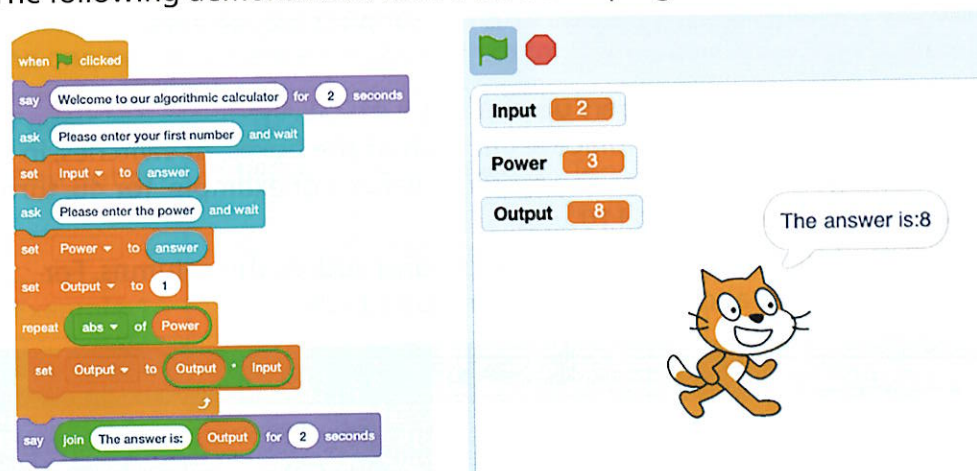
Program requirements

It is possible to create a base 2 'power of' program in **Scratch**. Scratch contains programmable functions and variables that can be combined to calculate a **denary** number of one number to the **power of** another. To do this, we will need the following **variables** and functionality:

- **Input:** The initial number, for example the 2 in 2^3
- **Power:** The power of value, for example the 3 in 2^3
- **Output:** The resulting denary number.

Scratch example

The following demonstrates how the Scratch program works:



1. A welcome message is displayed.
2. The user is asked for the first number (input).
3. The user is asked for the 'power of' (power).
4. The input value is multiplied by itself the same number of times as the power value.
5. The resulting denary number is saved as output and displayed on screen by the **sprite**.

Real-world advice

Scratch has been designed for sharing, downloading, and editing. There will always be an example similar to what you want to create, either in the help forums or uploaded by another Scratch user.

Python example

Using the same variables and functionality, a similar program can be created using Python. Note that the initial input is named 'userInput' to prevent any clashes with the input function. Also, the two inputs need to be converted to **integers** before any calculations can be done.

The following demonstrates how the Python program works:

1. The user is asked for the first number (userInput).
2. The user is asked for the 'power of' (power).
3. The ** operator (multiply the number on the left to the power of the number on the right) is used to create the variable output.
4. The resulting denary output is displayed on screen.

```
userInput = int(input("Please enter your first number:"))
power = int(input("Please enter the power:"))
output = userInput**power
print("The resulting denary number is:",output)
```

```
Please enter your first number: 2
Please enter the power: 3
The resulting denary number is: 8
```

Real-world advice

From the two examples, it is clear there is no specific 'power of' function in Scratch but there is in Python. This means that a calculator is much easier to create in Python.

Further investigation

- Experiment with both Scratch and Python programs. Can either of them calculate *any* number to the power of another?

Success criteria

- I can create a simple program in Scratch that can calculate the 'power of' one number to another.
- I can create a similar program in Python.

Internet Bandwidth

Related topics

- Network devices
- The internet

Keywords

bandwidth

bit

byte

data

file size

kibibyte (KiB)

mebibyte (MiB)

megabits (MB)

network

router

transfer

Learning objectives

1. Understand the concept of bandwidth.
2. Demonstrate an ability to calculate internet file transfer speeds.
3. Understand the term 'bandwidth bottleneck'.

What is bandwidth?

Consider a binary data stream of 1 second (s) and 0 seconds (s) travelling down a fibre optic cable. As fibre optic transmits light from one end to the other - light switched on is a 1 and light switched off is a 0. Repeatedly switching the light on or off creates a series of bits.



Bandwidth is the amount of **data** that can be sent in 1 second, in **bits** per second.

Using the fibre optic example:

- on, off, on, off, on, off, on, off, on, off, on, off in 1 second
- = 12 bits in 1 second (6 on + 6 off)
- = 12 bps (bits per second)

Megabits and Mebibits of bandwidth

When calculating bandwidth, bits per second is too small a unit to describe a modern internet connection. Modern bandwidths are measured in the base 10 unit of megabits (MB) per second (Mbps). Refer to pages 30–31 to look again at the difference between base 2 and base 10 units of data.

In these examples, we will be using a base 2 unit of **mebibytes (MiB)** per second. This means we need to remember the following:

1 **byte** = 8 bits

1 **kibibyte** = 1024 bytes

1 **mebibyte** = 1024 kibibytes

1 **kibibyte** = 8192 bits (1024×8)

1 **mebibyte** = 8388608 bits (8192×1024)

Real-world examples

The following are the average worldwide bandwidth speeds in megabits per second over a ten-year period:

2011: 2.0 Mbps

2016: 27.5 Mbps

2021: 53.0 Mbps

Calculating file transfer speeds

If you know the size of the file being downloaded and the current bandwidth, it is possible to calculate the file **transfer speed**, which is the time it takes to download a file.

For example:

File size = 120 KiB

Bandwidth = 40 KiB per second

File transfer speed = 3 seconds (120/40)

Typical file transfer speeds

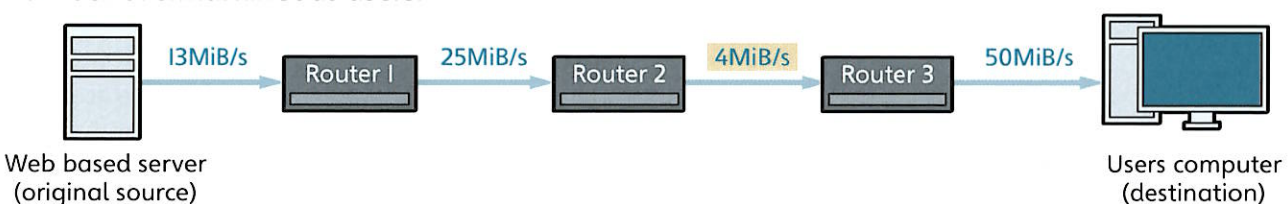
The following table contains three typical files and the time it will take to download each file using two different bandwidths.

	Bandwidth = 5MiB	Bandwidth = 40MiB
2048 KiB Email attachment	2048 KiB = 2 MiB $2 / 5 = 0.4$ Seconds	$2 / 40 = 0.05$ Seconds
15 MiB Audio file	$15 / 5 = 3$ Seconds	$15 / 40 = 0.375$ Seconds
50 MiB Video clip	$50 / 5 = 10$ Seconds	$50 / 40 = 1.25$ Seconds

Bandwidth bottlenecks

When you download a file from the internet, you do not have a direct connection to the original source of the file. The file will pass through many different networks on its journey, where each part, or **network**, along the way may contain its own bandwidth.

If too many users are accessing the internet at the same time, a bit like too much traffic on a road, the network will slow down and at this point any file can only be downloaded at the slowest bandwidth speed available. The following example shows the journey across multiple network **routers** and the bandwidth in MiB/s between them. The slowest bandwidth is 4 MiB/s – therefore, if a bottleneck occurs this will become the speed across the whole journey. To resolve this, it is important to check all equipment along the route and if possible, reduce the number of simultaneous users.



Further investigation

- How does the bandwidth in the area you live compare with the global average?
- Download a large file. Calculate the bandwidth and time how long it takes to arrive.

★ Success criteria

- I understand the concept of bandwidth.
- I can calculate internet file transfer speeds.
- I can describe the term bottleneck.

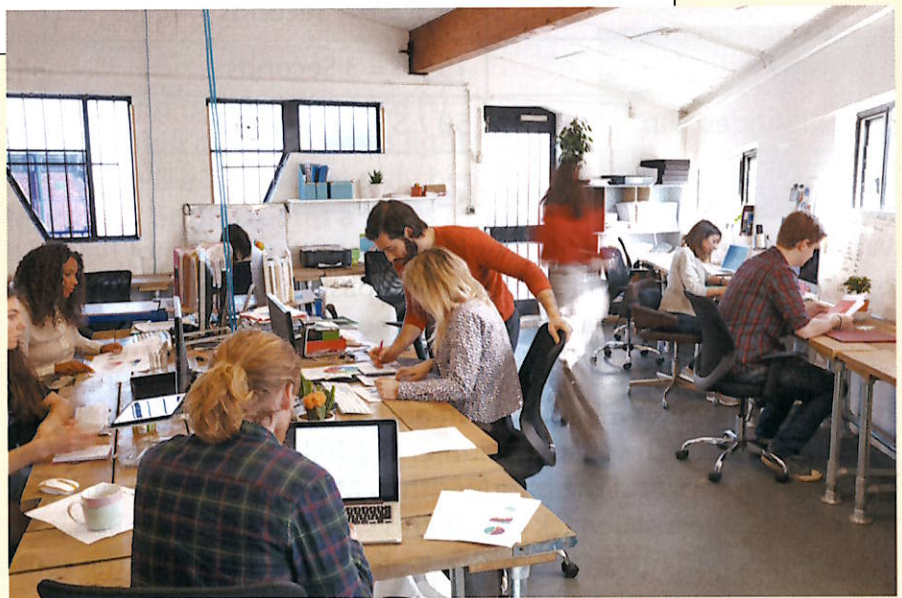
Unit 2 Mid-unit assessment

Typical 4-mark exam question

You are working as a network assistant in a large organisation. The building you work in is very large with lots of network equipment.

Employees are complaining about poor network speeds and you have been asked to investigate whether this is caused by a bandwidth bottleneck.

Describe the term bandwidth bottleneck and suggest two ways it might be resolved.



Typical 4-mark exam question

Bandwidth bottleneck is when you have too many users accessing the same network at the same time. This means the connection slows down to the slowest speed on the network.

To prevent this the number of users should be reduced and the equipment in the network should be checked for problems.



What good things can we see in this answer?

1. Good use of key words: users, connection, slowest.
2. Description of bottleneck includes too many users, equipment, and slowest section.
3. Two basic solutions suggested.



Which parts of the answer could be better?

1. No mention of technical equipment, such as routers or cables.
2. The 'slowest speed' could be explained.
3. Solution of reducing users is quite vague.



How can we improve this answer?

1. Describe how different bandwidth speeds occur between routers and that poor equipment can have an impact.
2. Describe how the slowest speed relates to the slowest section of the network.
3. Improve the answer to describe how it might be possible to reduce the number of users working simultaneously.

Creating a bandwidth calculator

Related topics

- Units of storage
- Arithmetic operators
- Algorithmic thinking

Keywords

bandwidth
concatenate
data

input
integer
output
program
Python

Scratch
test table
transfer
variables

Learning objectives

1. Understand the requirements of a simple file transfer calculator.
2. Demonstrate an ability to create a simple visual-coding calculator.
3. Demonstrate an ability to create a simple calculator using Python.

Program requirements

It is possible to create a file transfer **program** in **Scratch** that can calculate the time it takes for a file to download. To do this, we will need the following **variables** and functionality:

fileSize: The size of the file in bytes

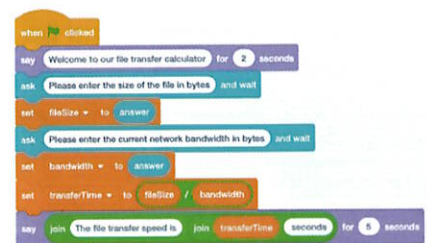
bandwidth: The current network bandwidth in bytes

transferTime: The time in seconds it will take to download

Scratch example

The following demonstrates how the Scratch program works here:

1. A welcome message is displayed.
2. The user is asked for the file size in bytes (**fileSize**).
3. The user is asked for the current bandwidth in bytes (**bandwidth**).
4. The **fileSize** value is divided by the **bandwidth** value.
5. The resulting denary number is saved as **transferTime** and displayed on screen by the sprite.



fileSize 120

bandwidth 40

transferTime 3

The file transfer speed is 3 seconds



Real-world advice

Notice how the 'say' command in the last line of the Scratch example can be used to **concatenate** (link together) messages and variables into a sentence.

Python example

Using the same variables and functionality, a similar program can be created using Python. Note the two **inputs** need to be converted to **integers** before any calculations can be done.

```
fileSize = int(input("Please enter the size of the file in bytes:"))
bandwidth = int(input("Please enter the current network bandwidth in
bytes:"))
transferTime = fileSize/bandwidth
print("The resulting file transfer speed is:",transferTime, "seconds")
Please enter the size of the file in bytes: 120
Please enter the current network bandwidth in bytes: 40
The resulting file transfer speed is: 3.0 seconds
```

The following demonstrates how the Python program works here:

1. The user is asked for the file size in bytes (fileSize).
2. The user is asked for the current bandwidth in bytes (bandwidth).
3. The fileSize value is divided by the bandwidth value using the divide operator.
4. The resulting denary transferTime is displayed on screen in seconds.

Test tables

It is important to carry out testing on any program and a **test table** is used to check the functionality of the program. The example below contains four tests (file sizes) and the expected answers.

fileSize (input)	Bandwidth (input)	transferTime (expected)	transferTime (actual result)	Changes required	Retest result
120	30	4			
1000	10	100			
480	60	8			
2400	60	40			

If any of the four tests produce an answer different to the expected result, the program will need to be checked and edited. Any changes can then be logged in the remaining three columns of the test table.

Further investigation

- Edit the Python program to still run correctly if the file size is entered in mebibytes, rather than bytes. It will need an additional conversion.

Success criteria

- I can create a simple program in Scratch that can calculate the file transfer speed of a download.
- I can create a similar program in Python.

Compression

Related topics

- Data storage
- Digital image properties

Keywords

compression
decompress

lossless

lossy

Learning objectives

1. Understand the purpose of compression.
2. Understand the difference between lossy and lossless compression.
3. Demonstrate an understanding of how you would use the most appropriate compression choice for a given scenario.

What is compression?

Compression is simply the reduction in size of any computer file. In order to compress a file size, the data structure needs to be altered so that it still resembles the original file, but requires less storage space. In this unit we will consider two methods of compression: lossy and lossless.

Some reasons why compression is useful:

- Attaching documents to emails (there is often a file size limit).
- Storing large files on a computer (taking up limited space).
- Transferring files on a storage device (e.g. a USB) that has limited space.
- Speeding up the download time of large files.

There are two ways to compress the size of a file: delete part of it or change the way it is structured.

Lossy compression and lossless compression

Lossy compression deletes part of the file, reducing the file size, but when the file is decompressed, the deleted parts cannot be restored. Lossy compression is never used for text files as characters will be lost, but it is often used for image and sound files. For example:

Image files (JPEG): pixels with very similar colours are replaced with the same colour that can be duplicated.

Sound files (MP3): low frequency sounds are removed, for example sounds that the human ear would normally miss.

Real-world examples

Two commonly used compression files are:

JPEG: An image file type that allows the user to select how much compression is used and compare this to the quality of the compressed image.

MP3: An audio file that also allows the user to control the compression settings.

Lossless compression prevents any data or quality being lost when the file is compressed and decompressed. This is ideal for files where it is essential that neither data nor quality is lost. However, as no information is deleted, this doesn't reduce the file size as much as a lossy compression.

Compression scenarios

Let's consider a range of scenarios, decide on the most appropriate choice of file compression and give reasons:

Scenario	Lossy or Lossless	Reason for choice
A medical database of patients and their treatment.	Lossless	It is essential that no data is lost and that no mistakes are created.
A full colour image for an advertising billboard.	Lossless	The image needs to be clear and maintain the quality of the original design.
Copies of holiday photos to email to members of the family.	Lossy	Prevent email file size limits being broken and as they are copies it is ok that the quality may be lower.
An audio recording of friends to post on a social network page.	Lossy	Smaller files are easier to upload, and friends won't worry about the quality.
A student's university application.	Lossless	It is important that all words and personal details are exactly as they were written.

Image compression settings

When saving an image using a file type with compression settings, it will normally adjust the following to reduce the amount of data required:

- **Colour depth:** the number of colours that can be created in one pixel, the lower the number of colours, the less data is required.
- **Resolution:** The number of pixels in one square inch, reducing this will reduce the clarity of the image.
- **Image size:** The number of horizontal and vertical pixels that make up the image, directly connected to file size.

Further investigation

- Investigate the history of JPEG and MP3 files and their impact on the internet today.
- Research file compression software packages.
- Explain why file compression might be used at home or at school.

★ Success criteria

- I know why file compression is important.
- I know the difference between lossy and lossless compression.
- I can choose the most appropriate compression option for different scenarios

Run Length Encoding

Related topics

- Creating original images
- File storage

Keywords

bit
compress
decompress

encoding
pixels
run length encoding (RLE)

Learning objectives

1. Understand the purpose of run length encoding.
2. Understand the link between run length encoding and binary pixels.
3. Demonstrate an ability to compress a simple graphic using run length encoding.

What is run length encoding?

Run length encoding (RLE) is a lossless compression method. It works by analysing and reducing any repeated characters in a string by grouping them together. The string is re-written in a series of pairs of bytes. For example:

YYXXXXXXXXYYYYYYZZZZZZZZ

Could be rewritten as: 2Y7X7Y8Z (2*Y,7*X,7*Y,8*Z)

If each character is 1 byte (assuming ASCII is being used), the file size can be calculated:

The original string is 24 bytes

The RLE version is 8 bytes

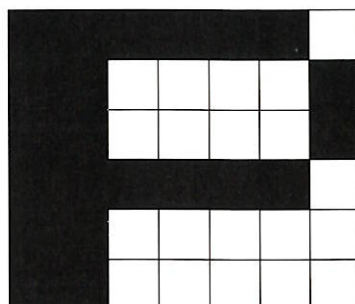
Image data

Before compressing an image using RLE, it is important to analyse the existing data to create a string sequence and calculate the original file size. The following is required:

- Image height and width in pixels
- The colour depth in bits. This will be 1 as we are using 1-bit images.

Compressing a simple image

Let's consider a simple 1-bit image similar to one in Unit 1 and write out the string sequence using b (black) and w (white).

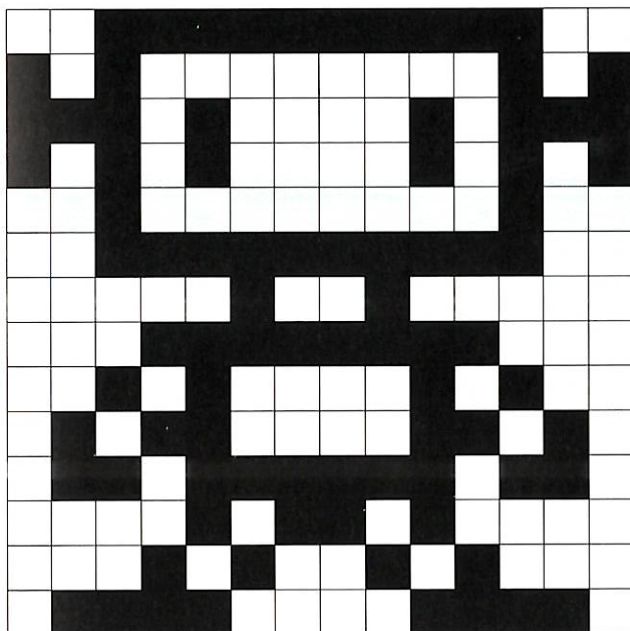


b	b	b	b	b	b	w
b	b	w	w	w	w	b
b	b	w	w	w	w	b
b	b	b	b	b	b	w
b	b	w	w	w	w	w
b	b	w	w	w	w	w

Original string	Size in bytes	RLE version	New size in bytes
bbbbbbw	7	6b1w	4
bbwwwwb	7	2b4w1b	6
bbwwwwb	7	2b4w1b	6
bbbbbbw	7	6b1w	4
bbwwwww	7	2b5w	4
bbwwwww	7	2b5w	4
Total file size	42		28

Compressing a detailed image

Let's apply the same theory to a larger image.



w	w	b	b	b	b	b	b	b	b	b	b	w	w
b	w	b	w	w	w	w	w	w	w	w	w	b	w
b	b	b	w	b	w	w	w	w	b	w	b	b	b
b	w	b	w	b	w	w	w	w	b	w	b	w	b
w	w	b	w	w	w	w	w	w	w	w	b	w	w
w	w	b	b	b	b	b	b	b	b	b	b	w	w
w	w	w	w	w	b	w	w	b	w	w	w	w	w
w	w	w	b	b	b	b	b	b	b	b	w	w	w
w	w	b	w	b	w	w	w	w	b	w	b	w	w
w	b	w	b	b	w	w	w	w	b	b	w	b	w
w	b	b	w	b	b	b	b	b	b	w	b	b	w
w	w	w	w	b	w	b	b	w	b	w	w	w	w
w	w	w	b	w	b	w	w	b	w	b	w	w	w
w	b	b	b	b	w	w	w	w	b	b	b	b	w

Original string	Size in bytes	RLE version	New size in bytes
wwbbbbbbbbbbww	14	2w10b2w	7
bwbwwwwwwwwbwb	14	1b1w1b8w1b1w1b	14
bbwbwwwwbwbwb	14	3b1w1b4w1b1w1b1w1b	18
bwbwbwwwwbwbwb	14	3b1w1b4w1b1w1b1w1b	18
wwbwwwwwwwwbww	14	2w1b8w1b2w	10
wwbbbbbbbbbbww	14	2w10b2w	7

wwwwwwbwwwbwwwwww	14	5w1b2w1b5w	10
wwwbbbbbbwww	14	3w8w3w	6
wwbwbwwwbwbwww	14	2w1b1w1b4w1b1w1b2w	18
wbwbbwwwbwbw	14	1w1b1w2b4w1b1w1b1w	18
wbbwbwwwbwbw	14	1w2b1w6b1w2b1w	14
wwwwbwbwbwww	14	4w1b1w2b1w1b4w	14
wwwbwbwbwbwww	14	3w1b1w1b2w1b1w1b3w	18
wbbbbwwwbwww	14	1w4b4w4b1w	10
Total file size	196		182

Real-world advice

Not every image compressed in this way will result in a smaller file size. For example, creating the letter M in a similar style increases the file size from 42 to 48 bytes!

Further investigation

- Look back at binary graphics you have already created in this book and apply RLE to see if they can be compressed. Select a small 2-bit graphic, due to the length of the code, and apply RLE.
- Investigate the history and modern equivalents of RLE.

Success criteria

- I know what RLE is.
- I can describe how RLE can be applied to a simple binary graphic.
- I can compress a simple 1-bit graphic using RLE.

Unit 2 End-of-unit assessment

Typical 4-mark exam question

As a reporter for an online news site, you are regularly posting articles and images to your social network feed. You are expected to apply compression to any images before uploading them.

Describe what compression is and why it is required in this role. Also explain if you would use lossy or lossless encoding.

Typical 4-mark answer

Compression is the process of reducing the size of a file. When posting to a social network, it is important that file sizes are small, so they don't take long to upload.

You should use lossy encoding because it creates smaller files.



What good things can we see in this answer?

1. Some key terms used: reducing, smaller.
2. Correct description of compression.
3. Correct choice of lossy encoding.



Which parts of the answer could be better?

1. Very brief description of compression.
2. Doesn't mention impact of compression for readers of post.
3. Only one reason given for choosing lossy.



How can we improve this answer?

1. Extend description of compression to include that it either removes data or restructures the file.
2. Describe that images will be quicker to upload but also quicker to download for reader.
3. State that loss of quality is not a problem for small social network images.



End-of-unit checklist

- ☐ I know what a storage device is.
- ☐ I know the difference between bits, bytes and nibbles.
- ☐ I can calculate the file size of simple text and graphic files.
- ☐ I can define file sizes in bits, bytes, kibibytes, mebibytes and gibibytes.
- ☐ I know the difference between base 2 and base 10 and can create place tables using indices.
- ☐ I can use place tables to convert denary and binary numbers.
- ☐ I can create simple calculators in Scratch and Python.
- ☐ I understand the concept of bandwidth and I can calculate internet file transfer speeds.
- ☐ I know the difference between lossy and lossless compression.
- ☐ I can choose the most appropriate compression option for different scenarios.
- ☐ I can apply RLE to a simple binary graphic.

Processing Data



Open it up and look inside a computer – what will you find? What impact is computer technology having on our planet? How many types of computers can one person carry at one time? These are all questions discussed in this unit, along with the skills to create professional digital products.

Key objectives:

1. Learn about computer hardware and components.
2. Learn about wearable technology and how it may develop.
3. Be able to create a simple logo.
4. Be able to create a simple spreadsheet.
5. Learn about the impact of technology on our environment.
6. Learn about how surveys and questionnaires are carried out.
7. Learn about how data can be presented.

By the end of the unit you will:

- be able to create a simple logo
- be able to create simple spreadsheets
- be able to create a survey or questionnaire
- be able to analyse and present data in a visual way.

Computer hardware

Related topics

- Computer systems

Keywords

central processing unit (CPU)
hard drive
memory

motherboard
power supply unit (PSU)
sound card
storage
video card

Learning objectives

1. Understand the difference between hardware and software.
2. Understand the main components of a computer.
3. Understand the common elements of different devices.

Hardware and software

Every computer system contains hardware and software – both rely on each other to function.

- Hardware refers to the physical components that we can hold/pick up, such as a **hard drive**, mouse, monitor or keyboard
- Software refers to the programs, commands and instructions that use the hardware to carry out a specific job, for example, an operating system, word processor or video game.

Types of computer

Many of the products we use daily contain some sort of computer system. The most commonly used computer devices include:

- Desktop computer
- Laptop computer
- Tablet
- Smartphone
- Other smart devices, such as watches, speakers, and televisions.

Real-world examples

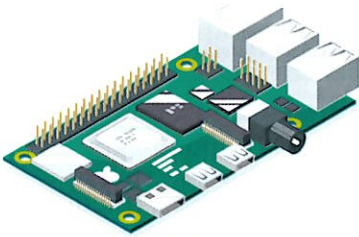
The first recognisable desktop computer for home use was the Apple II, which was launched in 1977. It could be plugged into a colour television and included word processing, graphical applications, and simple games.

Computer components

Although physically different, the key components across modern computer devices are the same and have the same function.

Component	Description
Central Processing Unit (CPU)	The 'brain' of the computer. It carries out all processing and the link between input and output data.
Motherboard	The main circuit board that all other components connect to.
Memory	The short-term memory, it stores program data and the results from calculations that allow programs to run. Also known as RAM.
Hard drive	The main storage device, it stores document files, programs, photos, games, and anything else held on your computer.
Power supply unit (PSU)	Provides the correct power supply to all components.
Graphics (video) and sound cards	Used to process video, images, and sound. A good quality graphics, or video card is essential for gaming.

Real-world advice



In recent years, a number of tiny computers have been created to help promote the use of computers around the world at a very low cost. One popular example is the Raspberry Pi. It is available for less than \$50 and it functions as a fully-working computer.

Further investigation

- Investigate the concept of processor speed and how it has increased in recent years. Take a look at a range of different computers and try to identify their components lists.

Success criteria

- I know the difference between hardware and software.
- I know there are different types of computer.
- I can describe some of the main components of a computer.

Wearable technology

Related topics

- Computer hardware
- Technology and society

Keywords

augmented reality (AR)
sensors

virtual reality (VR)

wearable technology

Learning objectives

1. Understand the concept of wearable technology.
2. Be able to describe some of the components used in wearable technology.
3. Be able to describe some of the advantages and disadvantages of wearable technology.

What is wearable technology?

Wearable technology is still a computer. It requires the same components as described on the previous page, but just extremely small versions of them. Wearable technology describes any computer device that is connected to the internet, and placed on your person, i.e. without the need to hold it in your hand. New examples are constantly being developed, but common examples include:

- Smart watches: providing smartphone notifications, heart rate and fitness tracking, contactless payment and audio playing capabilities.
- Smart jewellery: necklaces, rings and bracelets that can act as **sensors**, provide notifications and enable contactless payment.
- Smart glasses: providing either **augmented reality (AR)** as an overlay or **virtual reality (VR)** to fully immerse the viewer.
- Smart clothing: technology woven into our clothing that can either monitor our movement or provide information.

What can wearable technology do?

The purpose and functionality of each device is often linked to how it is worn. Functionality being incorporated into wearable technology includes:

- Augmented reality (AR) and virtual reality (VR): either replacing everything we see via a VR headset or adding additional information to our field of view through a pair of glasses.
- Fitness tracking: monitoring location, movement, and the health of the wearer through their heart-rate and number of calories burned.
- Payments: making contactless payments without the need for cash or a credit card.
- Communication: messaging and social network notifications.

Real-world examples

The battery life of a smart device is important to users and constantly having to charge puts many people off using them. For example, a modern smartwatch can often require daily charging if many of the features are used.

Advantages and disadvantages of wearable technology

A lot of wearable technology has been designed to become a part of our everyday lives. However, this increased use of technology has raised many questions regarding the potential advantages and disadvantages to the user. For example:

Advantages	Disadvantages
Always with us, less likely to forget it when we need it.	Limited battery life, regular charging often required.
Able to monitor our health and help set fitness goals.	Increasing our reliance on technology, giving us a false sense of security.
Able to provide immersive video games, making the player feel like they are there.	New technology is often very expensive.
Hands-free access to technology.	Privacy concerns.

The future of wearable technology

As smart clothing, better batteries and smaller components are further developed, many believe the future of wearable technology could include:

- Smart contact lenses displaying augmented reality.
- Smaller health monitors built into jewellery, buttons or even makeup.
- Smart footwear that could charge our devices through movement.
- Sensors built into our clothing, monitoring our temperature and stress levels.
- Implantable devices placed beneath our skin to monitor our health, provide notifications or even contactless payments!

Real-world advice

Augmented reality apps are becoming more and more popular in smartphone games. A great example is the Pokémon GO app, where popular game characters are brought into the real world using the device's camera!

Further investigation

- Investigate how AR and VR are increasingly used in video games.
- Discuss the increased use of smart devices. How do you think this varies around the world.
- Research some of the examples of future wearable technology. Will they work?

Success criteria

- I can describe the term 'wearable technology'.
- I know that wearable technology requires the same components as any computer system.
- I know there are many examples of wearable technology.
- I can describe some of the advantages and disadvantages of wearable technology.
- I am aware of how these devices might develop further in the future.

Logo design

Related topics

- Producing digital documents

Keywords

bitmap

brand

logo

marketing

transferability

vector

Learning objectives

1. Understand the purpose of a logo.
2. Understand the key components of a logo.
3. Understand some of the rules of good logo design.

What is a logo?

A **logo** is an original graphic that represents a product, **brand**, or service. It is created using any combination of text, photos and graphical images and is designed to be instantly recognisable by the viewer. The purpose of the logo might be any of the following:

- Marketing
- Advertising
- Product awareness
- To appeal to a specific target audience

A logo might be created for any of the following:

- A physical or virtual product
- A new game, film, or television programme
- An organisation, business, charity, or political party

Key logo components

A modern logo will contain the following components.

- The name of the product, brand, or service
- A colour scheme that might be transferred to additional marketing material
- A choice of font that reflects the style of the product, brand, or service
- Suitable imagery - either an original graphic or a photo/part of a photo.

The following fictitious logo highlights these key components.

Simple modern fonts

Simple graphics



Name of brand

Simple colour scheme that can be changed to suit the background

Real-world examples

Logos or symbols that represent a group of people have been used for thousands of years. One of the oldest commercially registered logos is for the English tea company, Twinings. Its logo has changed very little since its creation in 1787.

Good logo design rules

The list of rules below are designed to help create an interesting logo that will appeal to the target audience.

Rules	Description
Know the target audience	Your idea needs to appeal to them. What are their interests?
Construction	Will the logo be made using bitmaps or vectors ? Each has its advantages and disadvantages as discussed in Year 7.
Theme	What is the theme of your product or service? Is it meant to be fun, informative, or linked directly to a style?
Transferability	Can the logo be easily transferred to other documents or products of different sizes and shapes? This also includes electronic versions: websites, apps, social media pages.
Content	What must be included in the design and does the client have an existing/preferred style?
Text or graphical based	Does the full name need to be included or its initials? Are there images already associated with the brand that need to be included?

Real-world advice

Take a look at some really famous logos that you see around today. Many of them don't even include the name of the brand, just a symbol or shape is enough to be recognised (e.g. Nike). Often designs try to include too much content in a logo, such as the full name and a complex image of multiple parts. If in doubt, keep it simple.

Further investigation

- Choose three of your favourite brands and examine their logos. How are they constructed?
- Choose three topics, such as technology, sport, music or fashion. Create a simple logo like the one in this lesson, following the rules in the table.

Success criteria

- I know what the purpose of a logo is.
- I know what the key components of a modern logo are.
- I can describe rules that should be followed when designing a logo.

Introduction to spreadsheets

Related topics

- Data types
- Software applications

Keywords

alignment

border

cell

columns

data

formatting

formula

graph

merging

rows

spreadsheet

value

worksheet

Learning objectives

1. Understand the purpose of a spreadsheet.
2. Understand the key features of spreadsheet software.
3. Demonstrate knowledge of common spreadsheet formatting functions and tools.

What is a spreadsheet?

A **spreadsheet** is a document that allows the entering of numerical **data** into **rows** and **columns** that can be analysed, sorted, or used to complete calculations. Examples of common spreadsheet uses include:



- Finance: from a simple budget calculator to the salary records of an organisation.
- School data analysis: recording assessment scores and attendance data.
- Presenting data: creating visual representations of important data for information or marketing purposes using charts or graphics.
- Recording results: from the performance of a sports team to counting the variety of birds in a park for a wildlife survey.

Spreadsheet features

Modern spreadsheets normally include the following functions:

- Cell-based layout: each **value** can be placed in its own area, or **cell**. This allows data to be correctly referenced.
- Tables: arranging data in rows and columns.
- Multiple **worksheets**: allows multiple tables on separate pages to be saved in one file
- **Formulas** and functions: allow calculations to be carried out on data stored in the spreadsheet.
- Charts or **graphs**: presenting data in a visual way to help the viewer to understand.

Formatting tools

Formatting tools are used to improve the appearance of data in a spreadsheet, making it easier for the user to analyse.

- **Text:** changing the size, colour, font, and style of text.
- **Cell:** changing the colour and size of cells.
- **Border:** changing the lines around the edge of each cell. This allows you to create tables and highlight key data.
- **Alignment:** text can be aligned to the centre or to a specific corner of a cell.
- **Cell merging:** if it helps the layout of the spreadsheet, cells can be merged together into one. For example, the title of a table.

6 Month Sales Figures 2022	
Month	Profit
Jan	234
Feb	225.6
March	354.76
Apr	543.1
May	743.09
Jun	801

6 Month Sales Figures 2022	
Month	Profit
Jan	\$ 234.00
Feb	\$ 225.60
March	\$ 354.76
Apr	\$ 543.10
May	\$ 743.09
Jun	\$ 801.00

Data types

In the same way as processing data in a program, data within a spreadsheet can be formatted to suit any analysis that needs to be carried out.

Data type	Formatting	Example
Number	Number and decimal points	1.25
Currency	Symbol and decimal points	\$9.99
Percentage	Decimal points and percentage symbol	75%
Date	Day, month, and year, depending on local preference	04/05/1977
Time	Hours, minutes, and seconds as required	07:30:00

Real-world examples

The first version of Microsoft Excel was released on the Apple Macintosh in 1985, but it didn't appear on Windows computers until 1987.

Further investigation

- Investigate which spreadsheet software you have access to at school or at home.
- Open a blank spreadsheet, enter some text and numbers and experiment with the formatting tools in a similar way to the example above.

Success criteria

- I know what a spreadsheet is and how it is used.
- I can describe the key features of common spreadsheet applications.
- I know how information can be formatted in a spreadsheet.
- I know the different data types a spreadsheet can process.

Spreadsheet formulas and functions

Related topics

- Arithmetic operators
- Data types

Keywords

AVERAGE

cell

cell referencing

equals sign

formula

function

MAX

MIN

SUM

worksheet

Learning objectives

- Understand the purpose of formulas and functions in spreadsheet software.
- Understand the importance of cell referencing in spreadsheets.
- Demonstrate understanding of a range of common formulas and functions.

The need for formulas and functions

A simple spreadsheet is ideal for recording and formatting data in a table. Introducing formulas and functions allows for calculations to be carried out, the modelling of potential scenarios and asking *what if* questions about the data.

Cell referencing

Cell referencing allows any piece of data, across multiple **worksheets**, to be referenced in a **formula** or **function**.

The diagram shows examples of cell references. The reference is the column (letter) followed by the row (number).

	A	B	C	D	E
1				D1	
2		B2			E2
3					
4	A4				
5					E5
6			C6		

Real-world examples

Referencing positions using letters and numbers is how the popular board game Battleships is played. Each player calls out a letter and number reference to see if the other player has a ship at those coordinates.

Spreadsheet formulas

Operator	Description
+	Add
-	Minus
/	Divide
*	Multiply
=	Equal to

Formulas are used in spreadsheets to carry out arithmetic calculations. The five basic operators are in the following table. It is important to note that the **equals sign (=)** is used at the start of a spreadsheet formula, not the end.

For example:

$2 + 2 = \text{Answer}$

now becomes:

$\text{Answer} = 2 + 2$

Examples of formulas can be found later in this lesson.

Spreadsheet functions

There are loads of spreadsheet functions available. Four of the most common are shown in the table below.

Function	Description
SUM	Displays the total of a range of selected cells
AVERAGE	Displays the average of a range of selected cells
MIN	Displays the lowest value in a range of cells
MAX	Displays the highest value in a range of cells

Examples of formulas and functions

The following spreadsheet example shows a range of formulas and functions being used. In real life, the cells would only show the answer. This sample shows the formula or function and the answer in red.

	A	B	C	D	E
1	4	2	=A1+B1 6		5
2	3	3	=A2-B2 0		10
3	10	5	=A3/B3 2		6
4	6	2	=A4*B4 12		3
5					=SUM(E1:E4) 24
6					=AVERAGE(E1:E4) 6
7					=MIN(E1:E4) 3
8					=MAX(E1:E4) 10

Further investigation

- Recreate some of the examples on this page in your spreadsheet software and experiment with different values.
- Investigate three additional functions and try them out in your spreadsheet software.

★ Success criteria

- I know why formulas and functions can help when creating a spreadsheet.
- I know how to use the five basic operators in a spreadsheet
- I know how to use four basic functions in a spreadsheet

Unit 3 Mid-unit assessment

Typical 4-mark exam question

You are working for a computer manufacturer and have been invited to work on a new project to create a smart watch.

Describe two features that might be included in the design.
Describe two examples of computer components that should be included in this sort of device.



Typical 4-mark exam question

A smart watch could include the following two features: recording your heart rate and displaying your notifications. To be able to function it would require a CPU and some sort of hard drive to store the information.

**What good things can we see in this answer?**

1. Some key words used: heart rate, notifications, CPU, hard drive
2. Two correct features given.
3. Two correct components described.

**Which parts of the answer could be better?**

1. It does not say where the notifications might be from.
2. The letters CPU are not expanded.
3. A reason for hard drive is given but lacking any reason for CPU.

**How can we improve this answer?**

1. Explain that the notifications may have come from a linked smartphone or a social media site.
2. Expand the letters CPU to central processing unit.
3. Expand on mention of hard drive to include storing heart rate data or smart watch programming.

Spreadsheet modelling

Related topics

- Using variables
- Real-world scenarios
- Arithmetic operators

Keywords

AVERAGE
equals sign (=)
formula

function
goal seek
MAX
MIN
models

profit
SUM
what if scenarios

Learning objectives

1. Understand how spreadsheet modelling can be used to model real-life situations.
2. Demonstrate understanding through the creation of a simple spreadsheet model.
3. Demonstrate an understanding of the terms '*what if*' scenario and 'goal seek'.

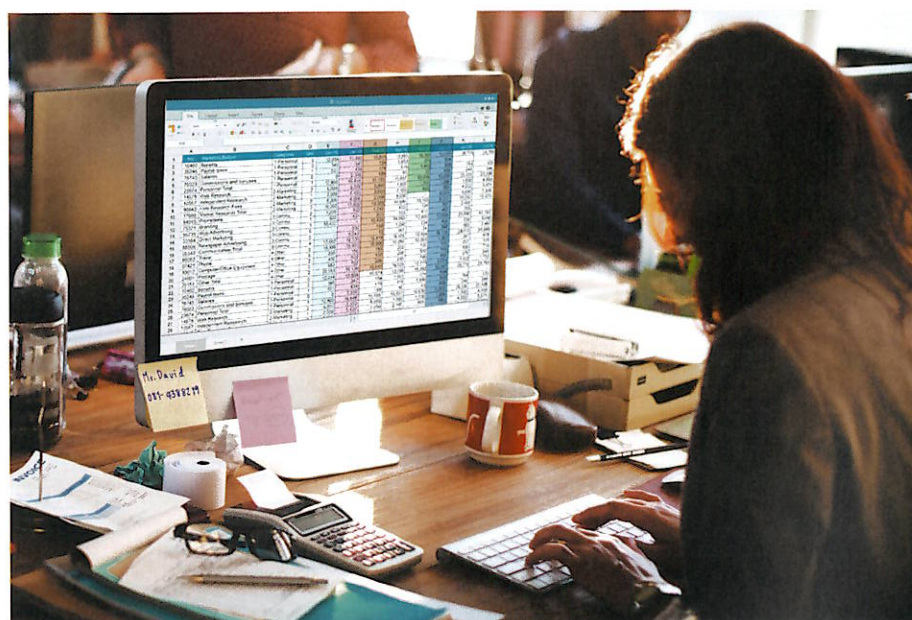
What is a spreadsheet model?

It allows the user to change variables in a spreadsheet in order to recreate a real situation or predict the result of a situation or scenario. Examples of simple **models** that include arithmetic **formulas** or the use of spreadsheet **functions** might include:

- The potential profit or loss that could be made by a new shop selling milkshakes.
- The potential cost of decorating a home by pricing materials and equipment.
- The potential grade outcomes of students based on previous assessments.

Real-world examples

Families sometimes create a simple spreadsheet model to record their weekly spending. They can use this to plan ahead and look for ways to save money.



A simple numerical model

The simple model below shows it is possible to gradually increase the distance to meet the annual target of 250km. The answers are displayed in red.

	A	B	C	D	E	F
1	Month	Distance		Annual total (km)		=SUM(B2:B13) 210
2	Jan	5		Annual target (km)		250
3	Feb	5		Distance remaining (km)		=F2-F1 40
4	Mar	10		Average (km)		=AVERAGE(B2:B13) 17.5
5	Apr	10		Lowest (km)		=MIN(B2:B13) 5
6	May	15		Highest (km)		=MAX(B2:B13) 30
7	Jun	15				
8	Jul	20				
9	Aug	20				
10	Sep	25				
11	Oct	25				
12	Nov	30				
13	Dec	30				

A simple financial model

Using some of the components from Lesson 5, this model calculates the cost of building a computer for a small computer shop and the potential **profit**.

	A	B	C	D
1	Component	Price (\$)	Staff build cost	\$150.00
2	Central Processing Unit	\$75.00	Total cost	=B11+D1 \$859.00
3	Motherboard	\$90.00	Retail price	\$799.00
4	Memory	\$50.00	Profit	=D3-D2 -\$60
5	Hard drive	\$100.00		
6	Power supply unit	\$75.00		
7	Video card	\$65.00		
8	Case	\$79.00		
9	Monitor	\$150.00		
10	Keyboard & mouse	\$25.00		
11	Build total	=SUM(B1:B10) \$709.00		

The cost of building the computer is higher than the retail price, so they would make a loss.

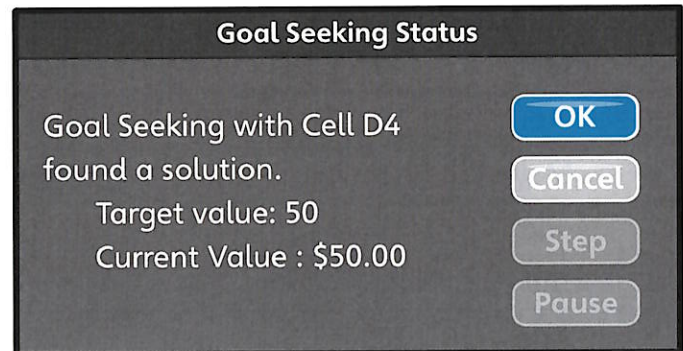
What if scenarios

One of the key benefits of a model is being able to see what might happen before having to actually do it. This means the computer shop owner above could consider several **what if scenarios**. For example:

- What if the components can be bought cheaper?
- What if the staff build cost could be reduced?
- What if the retail price is increased?

The goal seek function

Spreadsheet software includes a **goal seek** function that can adjust a variable until a desired outcome is created. In the example above, it could be used to find out the best cost of having a member of staff build the computer to make a profit of \$100.



Q Further investigation

- Create a similar version of the example spreadsheets shown in this lesson and experiment with your own values instead.
- Research real-life component prices to add to the computer model and recommend a new retail price if needed.
- Investigate what other costs might need to be included in a shop based model.

★ Success criteria

- I know how to create a simple numerical spreadsheet model.
- I know how to create a simple financial spreadsheet model.
- I know what a *what if* scenario is and what the goal seek tool is used for.
- I know what the difference is between a retail and wholesale price.

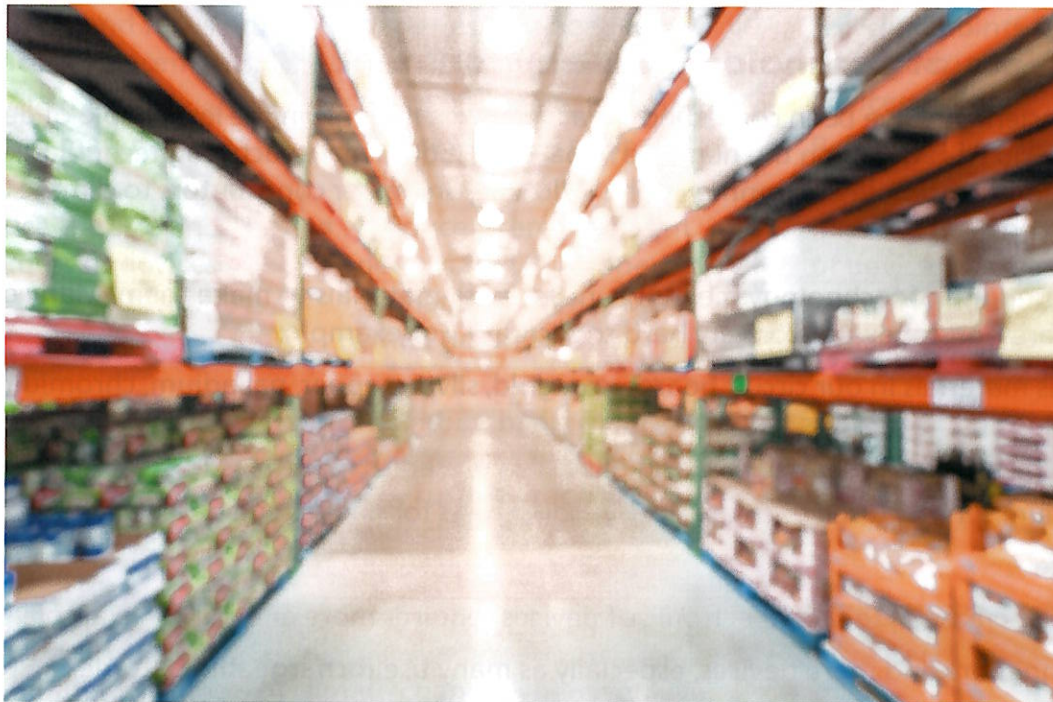
Retail and wholesale prices

In the example on page 61, the profit from a computer is calculated by taking away the total build cost from the **retail** price. This is correct if we are buying products directly from the people that make it. However supermarkets don't make all the food they sell and a website selling laptops may not have built them all themselves.

Companies that buy products from other companies, so they can sell them to us, buy their products at a **wholesale** price.

The differences between retail and wholesale prices

- Retailers sell products directly to us, the consumers, at a retail price. This is more expensive than wholesale because it includes the price they paid to buy the product and their costs to be able to sell it, such as a shop, staff or website.
- Wholesalers sell products only to retail companies at a wholesale price. The wholesale price is cheaper because retail companies will buy lots of stock, fifty computers instead of one for example.



Environmental impact of technology

Related topics

- Computer hardware
- Impact on society

Keywords

electronic waste (e-waste)
hazardous
hibernate

landfill
rechargeable
recycled
toxic

Learning objectives

1. Understand the effects of increased use of technology on our environment.
2. Understand what impact material and energy use has on our environment.
3. Understand the effect of electronic waste disposal on our environment.
4. Understand some ways we can reduce environmental damage caused by technology.

The effects of technology our environment

Although scientists have been concerned for a long time, it is only in the last couple of decades that the world has agreed on the negative impact of modern manufacturing technologies and power generation on the **environment**. The following are major contributors to climate change.

Material choices

Our increased use of technology means we need more materials to make them. Many of these materials are non-renewable or very hard to recreate. For example:

- Precious materials in our circuit boards, e.g. gold, silver, copper, palladium, chromium, along with **toxic** and **hazardous** materials such as lead, cadmium, and mercury.
- Sand, which is used to make glass and oil, which is used to make plastic.
- Aluminium, which is used to create sleek product cases for phones and other electronic devices.

Energy use

The global increase in the use electronic of devices requires more energy. For example:

- Charging more and more devices, especially as many use **rechargeable** batteries.
- Powering homes, satellites, mobile phone communication towers and the infrastructure behind the internet.

Electronic waste (e-waste)

E-waste describes devices that are no longer used and, instead of being **recycled**, have just been dumped in **landfill** sites. As new models are launched so frequently, those who want to always have the latest technology are not using original devices to their natural end. Problems caused by e-waste include:

- Perfectly usable devices are not recycled.



- Rare and toxic materials can leak into the ground if devices are not disposed of properly.
- Discarded electronics in landfill sites can be a health risk to workers.

Real-world examples

In the year 2000, there were around 400 million internet users. Just 20 years later in 2020, there were over 4.6 billion users!

By the end of 2019, the world had produced over 50 million metric tons of electronic waste. Only 17.4% of this was officially recycled that year.

Preventing environment damage

Fortunately, there are solutions to these problems and ways to reduce the damage, as shown in the following table.

Material choices	Energy use	Electronic waste
■ Reduce the need for rare elements and focus on artificial replacements. ■ Recycle the materials used to manufacture devices. ■ Reduce the size and thickness of devices.	■ Create more efficient batteries that need less charging and last longer. ■ Focus on renewable energy sources such as solar and wind. ■ Produce energy-efficient devices that will 'hibernate', so that they save power when not in use.	■ Create collection and recycling points for damaged/unwanted devices. ■ Rehome unwanted devices. ■ Improve the upgradability of devices, such as adding memory, rather than replacing them.

Real-world advice

Many technology companies are now running recycling schemes where unwanted devices are collected as a trade-in for a new product.

Further investigation

- Investigate the effects of climate change where you live.
- Discuss in a group how the number of devices you use has increased in your lifetime.
- Choose a technology company you are familiar with and investigate their efforts to reduce their environmental impact.

Success criteria

- I know that technology can have a negative impact on our environment.
- I know that it is important to consider how devices are built, powered, and disposed.
- I know ways that any negative impact can be reduced.

Presenting data in an informative way

Related topics

- Surveys and questionnaires

Keywords

chart
data

graph
infographic
labels

Learning objectives

1. Understand how data can be presented in a visual way.
2. Understand the importance of presenting data in an informative and interesting way.
3. Understand the purpose of an infographic.

The purpose of charts and graphs

A table of **data**, either collected or calculated, is not always easy to read and it can be difficult to draw conclusions from it. To make it clearer, a **chart** or **graph** is used to take data and convert it into a visual format. Data can represent:

- Percentages of a whole
- Columns of different sizes, linking size to their value
- Changing values over time.

Types of chart and graph

There are many types of chart, each is designed for a particular purpose. Four of the most common charts are:

- Pie chart: used to show individual parts of a whole, ideal for percentages.
- Line chart: shows the changing value over time.
- Bar/column chart: displays the value of a range of similar items against the same quantity, ideal for comparing prices.
- Scatter chart: displays two sets of data to see if there is a connection between them.

Real-world examples

Check out the United Nations website, a real-world source of data, which is ideal for experimenting with charts: <https://data.un.org/>

What is an infographic?

An **infographic** is document conveying information in a visually interesting and straightforward way. It might contain:

- Text in short blocks, bullet points, headlines or as **labels**.
- Charts/graphs such as any of the examples described in this section.
- Diagrams that explain a topic or item using simple graphics.
- Timelines like a year-by-year timeline that can be linked to relevant events.
- House style, where the text, graphics and charts follow a theme in respect to colour and layout.

Why do we use infographics?

Popular uses of infographics include:

- Marketing to promote a product or service.
- Historical events – visually telling the story through a timeline.
- Government leaflets explaining important or essential information in a fun, straightforward way.
- Educational material explaining a process or event.

Real-world advice

Born in Scotland in 1759, the engineer and political economist William Playfair was credited with creating the infographic. Not only had he already invented many charts and graphs we use today, but he combined them with information and graphics.



Further investigation

- Explore the charts and graphs you can create in your chosen spreadsheet software,
- Search for and download some popular infographics. What similarities do they have to one another?
- Investigate the history of infographics.

Success criteria

- I know how data can be presented in a visual way.
- I know about different types of charts and graphs.
- I know what an infographic is.

Surveys and questionnaires

Related topics

- Data analysis
- Computers in society

Keywords

bias
leading question

qualitative
quantitative
questionnaire
reliability

sample size
survey
unreliable

Learning objectives

1. Understand the purpose of collecting user data.
2. Understand how surveys are used.
3. Understand the importance of good questioning techniques.
4. Understand the difference between qualitative and quantitative data.

Collecting user data

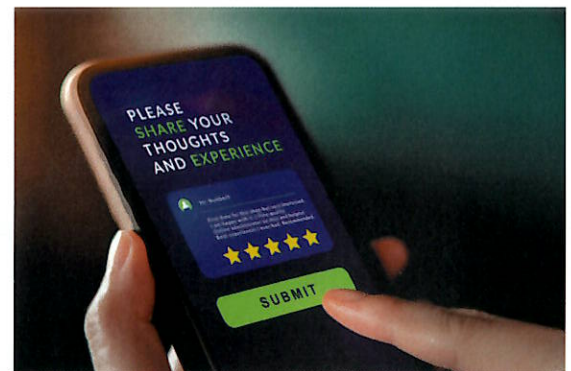
The collection of data, opinion and information is an important part of many aspects of our lives, including:

- New product launches, such as a smartphone.
- New government rules or changes to the way systems run, like changes to the way people vote.
- Evaluating existing products in order to improve them.

Surveys

A **survey** or **questionnaire** is a collection of questions, normally based around collecting data on a specific theme. Surveys can be carried out in a number of ways, for example:

- In person
- Virtually in person, via video or audio link
- Online via a website
- Via text message or app
- As a written document to complete.



Real-world examples

In recent years, the traditional paper-based data collection survey has almost entirely been replaced by electronic versions. This is quicker and more convenient for most people, but it can leave out some groups, such as the elderly, so alternative methods need to be provided.

Good questioning techniques

Asking the right questions is essential when collecting data. The following should be considered for every method of data collection.

- Audience: are you talking to the right people and will they be interested in the topic?
- Confusion: over-complicated questions will confuse the user and result in poor responses.
- Language: are questions written in a clear, simple way using words suitable for all?
- Bias: make sure leading questions are not being used. (A leading question makes someone give the answer you want and you can't be sure of the **reliability** of your results.)
- Sample size: are the right number of people being consulted?

Qualitative and quantitative data

When creating a survey or questionnaire, it is important to consider the type of data you will collect from it. You can then choose to collect **qualitative** data, **quantitative** data or a mixture of both.

Qualitative data: surveys are questionnaires with open-ended questions, so they cannot be answered with a simple 'yes' or 'no'. These will normally be answered in written sentences or voice recordings. These might include:

- Interviews
- Group discussions
- Observations.

Quantitative data: these are close-ended questions with specific answer choices, designed to count and analyse afterwards. These are much easier to conduct online and might include:

- Yes/no questions
- Multiple choice
- Choosing a point on a scale, e.g. 'Choose a number from 1 to 5'.

Real-world advice

Collecting online reviews is now a major part of online shopping sites. Before making a purchase of a product or service, people can read the opinions and feedback of previous customers. It is important to carefully consider these reviews, as some may be unreliable or untrue.

Further investigation

- Investigate how your school conducts surveys and questionnaires.
- Create a class survey of your own, try to include questions that will generate qualitative and quantitative data.

Success criteria

- I know why data is collected.
- I know the different ways data can be collected.
- I can describe examples of good questioning techniques.
- I know the difference between qualitative and quantitative data.

Advanced spreadsheet functions

Related topics

- Surveys and questionnaires
- Presenting data

Keywords

axis

COUNT

COUNTIF

criteria

formatting

formula

function

Learning objectives

1. Demonstrate understanding of the COUNT and COUNTIF function.
2. Understand the importance of appropriately formatted charts.

	A	B	C	D	E
1	Person	Devices		Total People	=COUNT(A2:6) 5
2	01	3		Total devices	=COUNT(B2:6) 14
3	02	2			
4	03	1			
5	04	5			
6	05	3			

The **COUNT** function counts the number of values in a range of cells. It is important to note that this function can only count numbers, not letters.

The example shown above counts the numbers of values in a simple survey, asking people how many computer devices are in their home.

	A	B	C	D
1	Person	OS Choice		Number of Apple OS
2	01	Windows		=COUNTIF(BA2:6,"Apple") 2
3	02	Apple		
4	03	Apple		
5	04	Linux		
6	05	Linux		

The **COUNTIF** function is used to count the number of values in a range of cells that meet specific **criteria**.

The example shown above counts the numbers of time the term 'Apple' is used when asked about their current operating system.

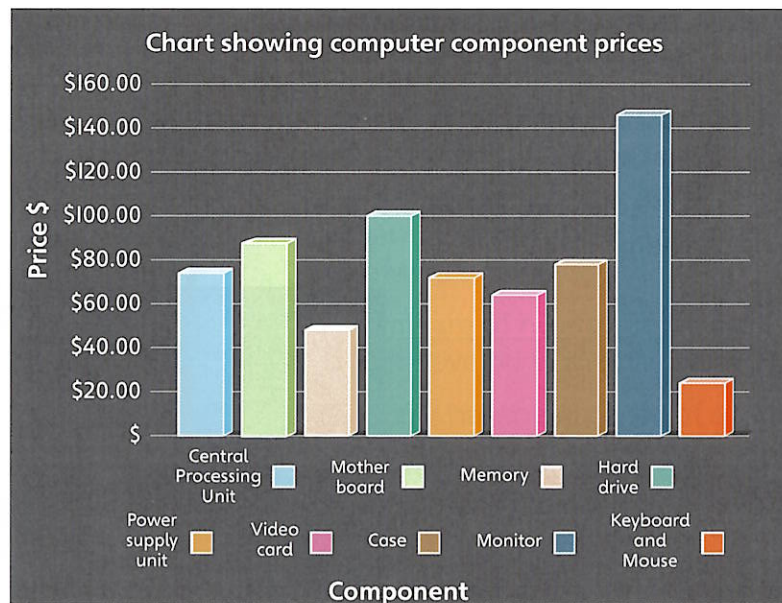
Real-world examples

- The COUNT and COUNTIF function can be found in MS Excel, Google Sheets and Apple Numbers.

Formatting charts

When presenting a chart, it is important to consider the audience and how they will look at the data being presented. There are a number of considerations when choosing your **formatting**:

- The colours used
- The style of text
- The language used
- The type of chart used – discussed earlier in this section
- The graphical presentation, either 2D or 3D



This example shows a typical column chart and the key areas that can be edited to make the chart as presentable as possible.

Real-world examples

Charts are often used by the government to present either financial or medical information. Occasionally, they have been accused of omitting (leaving out) key information, such as an axis, scale or key that makes the chart appear biased in one particular direction.

Further investigation

- Experiment with the COUNT and COUNTIF functions, creating similar examples to those shown.
- Experiment with charts you have already created and use the formatting tools to edit the style, layout, and colour.
- Discuss in a group any examples of biased charts being used in public.

Success criteria

- I know how to use the COUNT and COUNTIF functions.
- I know how important it is to present data clearly.
- I know which elements of a chart can be edited.

Unit 3 End-of-unit assessment

Typical 4-mark exam question

You are working for an environmental charity and are producing leaflets to highlight the impact of technology on our environment. Describe two current concerns and the two potential solutions to them.



Typical 4-mark exam question

Two major concerns are the increased use of energy and the pollution caused by e-waste. We can try and reduce these by choosing renewable and low-energy systems. To reduce e-waste, we should try and use our electronic devices for longer.

**What good things can we see in this answer?**

1. Some key terms used: pollution, e-waste, renewable, low-energy.
2. Two clear examples of concerns.
3. Two examples of potential solutions given.

**Which parts of the answer could be better?**

1. The reason for increased energy use isn't explained.
2. The term e-waste isn't explained.
3. Why using our devices for longer helps the situation isn't really described.

**How can we improve this answer?**

1. Explain that the many new devices we use all require power, increasing the demands on power stations.
2. E-waste is the disposal of unwanted or malfunctioning electronic equipment into landfill.
3. Explain that using our devices for longer will reduce the need to manufacture new ones.

**End-of-unit checklist**

- ☐ I know the difference between hardware and software.
- ☐ I can describe some of the main components of a computer.
- ☐ I can describe examples of wearable technology.
- ☐ I can describe some of the advantages and disadvantages of wearable technology.
- ☐ I am aware of how these devices might develop in the future.
- ☐ I know what the purpose of a logo is.
- ☐ I can describe rules that should be followed when designing a logo.
- ☐ I know what a spreadsheet is and how it is used.
- ☐ I know the different data types a spreadsheet can process.
- ☐ I know why formulas and functions can help when creating a spreadsheet.
- ☐ I know how to create a simple spreadsheet model.
- ☐ I know that technology can have a negative impact on our environment.
- ☐ I know about different types of chart and graph.
- ☐ I know what an infographic is.
- ☐ I know why data is collected and the different ways it can be collected.
- ☐ I can describe examples of good questioning techniques.
- ☐ I know the difference between qualitative and quantitative data.
- ☐ I know how to use the COUNT and COUNTIF functions.

Unit 4

Analysing Data



You are working as a researcher. What is the difference between primary and secondary sources of information and in what ways can this data be processed and analysed? In this unit you will learn about the classification of data, how it can be effectively processed and how social networks are used to both spread and collect information.

Key objectives:

1. Learn about social networks and what they offer.
2. Learn about how surveys are used.
3. Learn how spreadsheets can be used.
4. Be able to create simple vector and bitmap graphics.
5. Learn about what is included in an infographic.

By the end of the unit you will:

- create simple surveys
- create spreadsheets based on surveys
- be able to use sort and filter functions in a spreadsheet
- be able to create appropriate charts and graphs
- be able to create an infographic using data I have collected.

Social media

Related topics

- The internet and the world wide web (WWW)
- Privacy issues

Keywords

advertisements
blogging
cyberbullying

gaming
internet relay chat (IRC)
media sharing
messaging

networking
platform
social marketing
social media
target audience

Learning objectives

1. Understand the term social media.
2. Understand the different types of social media and the features of each type.
3. Understand some of the advantages and disadvantages of social networking.

What is social media?

We socialise with friends and family in a variety of ways, for example:

- Chatting
- Meeting at events
- Talking to new people with similar interests
- Playing games or watching films
- Sharing photos and videos.

All of these activities can now be done online, and this is the basis of a social media network – taking part in social activities via the internet and across any distance.

Types of social media

These are many types of **social media**. Rather than offering every type of social activity, they normally focus on a particular activity or group of activities. Common examples include:

- **Instant messaging:** real-time chat via keyboard.
- **Blogging:** posting time and date-stamped articles on a topic, maybe a hobby or profession. A blog is different from a static website as the content is constantly being updated and pages are stored in date order.
- **Media sharing:** sharing original or internet-sourced photos, videos, music, or animation with others.
- **Networking:** connecting with others with similar interests anywhere in the world.
- **Gaming:** playing or watching online games with international players.

What is cyberbullying?

Cyberbullying is the use of any internet-linked service to bully, harass or be anti-social to other users. It can happen on all types of social network on any compatible device.

Real-world examples

Internet relay chat (IRC), introduced in 1988, allowed people to send instant text messages to each other around the world. Many current social networks developed from this system.

Common social media features

Across the range of social networks there are many common features. This helps users swap from one **platform** to another. These include:

- Creating posts: these are date-stamped text messages or picture messages.
- Creating a profile: personal information about the user, their hobbies, interests and so on.
- Friend requests: these might be friends or work colleagues. A link is made between them and their activities are shared on the site.
- Followers: you may follow a person, organisation or club and their activity will be shared with you.
- Advertising: **social marketing** is part of every free social media site. Advertisers will target users (their **target audience**) with specific **advertisements** based on their profile and interests.

Advantages and disadvantages of social media

Advantages	Disadvantages
Ability to chat with family and friends around the world	Friend requests from strangers
Many social networks are free to use	Risk of sharing personal information online
Ability to look for and apply for jobs	Users often targeted with scams
Social interaction if living in a remote area	Access to inappropriate content
Up-to-date news and world events	Cyberbullying
Chance to make new friends with similar interests	Fake news and biased opinion
Ability to link with professionals and celebrities around the world	Personal information being shared with other sites and advertisers and lack of privacy

Further investigation

- Discuss in a group the social networks you all use.
- Investigate the most popular social network in your country.
- Can you think of any other advantages and disadvantages of social networks?

Success criteria

- I know what a social network is.
- I can describe different types of social network.
- I know some of the advantages and disadvantages of social networks.

What makes a good spreadsheet

Related topics

- Designing digital documents

Keywords

data
formatting

spreadsheet

survey

user-friendly

worksheet

Learning objectives

1. Understand what makes a good spreadsheet.
2. Understand the layout and design tools that make a spreadsheet user-friendly.
3. Be able to add data from a simple survey to a spreadsheet.

What makes a good spreadsheet?

A good **spreadsheet** should be straightforward to use by anyone who wants to edit or add data to it. Making it **user-friendly** is essential and this can be done by:

- Making it easy to follow for the reader – is it clear which **data** has been processed and why?
- Using clear headings for tables.
- Using colours and **formatting** to highlight any key areas.
- Avoid overcrowding a **worksheet** – information can be spread across multiple sheets.

Layout and design tools

Most spreadsheet software will have versions of the tools shown in the following example. These are ideal for improving the visual appearance of a spreadsheet and some of these have already been discussed in Unit 3.

Use of text formatting

Table showing users favourite website type

Use of text wrapping within cells

Use of borders of different thicknesses

Background and cell colours

Website	Percentage of total
Shopping	20
Gaming	20
Blogging	15
Messaging	5
News	5
Fashion	5
Sport	5
Social	25
Total	100
Highest	25
Lowest	5

Real-world advice

Imagine a budget planner for an upcoming music festival. The planners may want to share the finance spreadsheet with others running the project but wouldn't like them to edit the figures. This can be achieved by using the protect or lock cells tool. This allows a password to be added to prevent anyone without the password making changes

A simple survey

The following is an example of a simple **survey** designed to collect information on social media. It has been designed using the tips from Unit 3.

On average, how many hours do you spend on a device each day?	
What is your favourite type of device to work on? 1: Laptop 2: Tablet 3: Smartphone	
How much do you spend each year on a device?	

Adding survey data to a spreadsheet

Let's take the results of this simple survey and record them in a spreadsheet model.

	A	B	C	D	E
1	Computer use survey				
2		Person	Time spent per day (Hours)	Favourite device (1,2,3)	Amount per year (£)
3		1	2	3	£150.00
4		2	3	3	£20.00
5		3	6	1	£3,000.00
6		4	0.5	2	£400.00
7		5	1	2	£350.00
8		6	2	1	£500.00
9		7	4	2	£175.00
10		8	6	1	£1,500.00
11		9	7	1	£2,000.00
12		10	2	3	£500.00
13		Total	33.5		£8,595.00
14		Average	3.35		£859.50
15					

Real-world examples

When creating a new spreadsheet, most software applications will have existing templates that you can look at to get ideas for how to lay out sources of data for analysis.

Further investigation

- Experiment with the formatting tools in your choice of spreadsheet software.
- Investigate the lock and protect options in spreadsheets.

Success criteria

- I can describe what features make a good spreadsheet.
- I know how to use layout and design tools to make a spreadsheet more user-friendly.
- I can transfer data from a simple survey to a spreadsheet.

Knowledge, data and information

Related topics

- Analysing data

Keywords

data

information

knowledge

primary data

qualitative data

quantitative data

secondary data

survey

Learning objectives

1. Understand the terms 'knowledge', 'data', and 'information'.
2. Be able to describe examples of knowledge, data, and information.
3. Understand the terms 'primary' and 'secondary' data.
4. Be able to describe examples of primary and secondary data sources.

Knowledge, data and information

These three terms are often used interchangeably in the same sentence; however, it is important to be able to describe them and be aware of the differences between them.

- **Data** is a value that on its own has no meaning - the number 3, for example.
- **Information** is data that has meaning, for example, 3 out of 10 computers use the Apple operating system.
- **Knowledge** is analysing data and information and using them to create something new. For example, the design of a new program based on researching existing products and user opinions.

Examples of knowledge, data and information

The following examples show how values can be interpreted as data, information and knowledge. It is important to consider how to collect this data when designing a **survey**.

Data	Information	Knowledge
209	A website has had 209 views today.	That is more than yesterday, the new design is helping.
3%	When asked, only 3% of users liked the new purple smartphone.	Next time we will do more research before launching a new product variant.
\$1500	On average people spend around \$1500 on computer devices each year.	This has increased year on year. It is important to think about improved recycling.

Real-world advice

Remember you can't collect knowledge; it is gained from researching data and relevant information.

Primary and secondary data

We already explored the difference between **quantitative** and **qualitative data** in Unit 3. It is also important to consider the source of any survey data we wish to collect. These are described as either primary or secondary sources.

- **Primary data** is information you have collected yourself
- **Secondary data** has already been collected by someone else.

Examples of primary and secondary data

Primary data	Secondary data
A survey carried out by you.	Results from surveys already carried out by someone else.
Interviewing someone in person.	Information from existing interviews conducted by someone else.
Taking your own photographs or videos or recording audio.	Using existing images, sound or video, including television, film or radio created or recorded by someone else.
Articles, books, posts you have written	Articles, books, websites and magazines written by someone else.

Real-world advice

Neither one of primary or secondary data sources is better than the other. Good research will be based on both as it is impossible to travel back in time to carry out research in the past to compare with modern results. Secondary data can also be used to increase your sample size or the geographic span of your research.

Further investigation

- Investigate two primary and two secondary sources of information you might use for your own research.
- Try to add one more example of knowledge, data and information to the table in this lesson.

Success criteria

- I know what the terms knowledge, data and information mean.
- I can describe examples of knowledge, data and information.
- I know the difference between primary and secondary data sources.
- I can list examples of primary and secondary data sources.

Sorting and filtering a spreadsheet

Related topics

- Surveys and questionnaires

Keywords

alphabetically
ascending
criteria

data
descending
filter
numerically

pattern
sort
trends

Learning objectives

- Understand the purpose of the sort function in a spreadsheet.
- Understand the purpose of the filter function in a spreadsheet.
- Demonstrate an ability to use the sort and filter functions to identify trends in a simple model.

The sort function

Analysing **data** isn't just the use of formulas and functions, it's the organisation of data into a format that allows trends and patterns to be spotted. The **sort** function allows data to be sorted either **alphabetically** or **numerically** in an **ascending** or **descending** order.

Let's consider the table below, which shows the results from the survey on page 79. It is being sorted by the amount of money spent.

	A	B	C	D	E	F
1	Computer use survey					
2		Person	Time spent per day (Hours)	Favourite device (1,2,3)	Amount per year (£)	
3		1	2	3	£150.00	
4		2	3	3	£20.00	
5		3	6	1	£3,000.00	
6		4	0.5	2	£400.00	
7		5	1	2	£350.00	
8		6	2	1	£500.00	
9		7	4	2	£175.00	
10		8	6	1	£1,500.00	
11		9	7	1	£2,000.00	
12		10				
13		Total				
14		Average				
15						
16						
17						

The table is to be sorted by the amount spent column.

The **sort** tool includes options for headings and whether to sort in ascending or descending order.

Sort

Add levels to sort by: ☒ My list has headers

Column Start on Order

Sort by Amount per year (£) Values Smallest to largest

Real-world advice

When creating a **sort** or **filter** using spreadsheet software, be careful when selecting the cells that you need. The software might assume the first row is a heading and ignore it.

The filter function

The filter tool allows only data to be displayed that meet certain **criteria**. This filter is applied to the whole table and only the rows containing those criteria are displayed.

In the following example, the filter function has been used to only display those people that said their favourite device was option 1 (a laptop).

It shows four people chose this option out of a possible ten.

	A	B	C	D	E
I	Computer use survey				
2		Person	Time spent per day (Hours)	Favourite device (1,2,3)	Amount per year (£)
6	6		2	1	£500.00
10	8		6	1	£1,500.00
11	9		7	1	£2,000.00
5	3		6	1	£3,000.00

Once the filter function is used, each column can be used to add a filter. In this example, the option 1 has been selected

Favourite Device (1, 2, 3)

Sort

↑ Acending
↓ Decending

By colour: None

Filter
By colour: None

Equals
1

☒ And
 ☐ Or

Choose one

☒ Select All
 ☒ 1
 ☐ 2
 ☐ 3
 ☐ (Blanks)

Clear Filter

Looking for trends in a simple model

If we use the sort and filter tools on this spreadsheet of survey results, it would be possible to find out the following to see if there are any **trends** or **patterns**.

- How many selected each device option.
- The highest and lowest amounts spent.
- Does the amount spent increase with the number of hours spent?
- Do users of certain devices spend more or less than other users?

Further investigation

- In your spreadsheet software, look for the sort and filter functions. Do they function in the same way as the example shown?
- Experiment with the sort and filter tool on a table of your own.

Success criteria

- I know what the sort and filter tools can do.
- I can use the sort and filter tools in a simple spreadsheet.
- I can use the sort and filter tools to look for trends in a simple model.

Design techniques for an online advertisement

Related topics

- The internet and the world wide web (WWW)
- Privacy issues
- Designing digital documents

Keywords

ad-blocker
advertising
attract
complementary colours

design
layout
persuade
social media platforms
target audience

Learning objectives

1. Understand the purpose of an online advertisement.
2. Be able to describe common sources of online advertisements.
3. Understand common layout and colour techniques for designing an online advertisement.

Online advertising

Unlike paper-based **advertising** in a magazine, newspaper or leaflet, online advertising is often targeted at specific users, a **target audience** with an interest in the product. Its purpose is to:

- Grab your attention by standing out.
- Provide enough information to persuade the user to click the advert to find out more about:
 - the name of the product
 - a special offer related to it
 - images of the product or people using it.



Sources of online advertising

The location of online advertising is almost as important as its content. It is very common to find advertising in locations with a theme linked to the advert, such as:

- Website pages
- Social media pages
- Adverts within applications or games (in-app).

Real-world examples

An advert can be targeted at a specific user in the following ways:

- Looking at your search habits/history and frequently visited web pages.
- The type of **social media platforms** you visit.
- The interests of friends and organisations you follow online.



Unit 4 Mid-unit assessment

Typical 4-mark exam question

You are working for an online sports magazine and have been asked to talk to new employees on their first day about information sources.

Describe the differences between knowledge, data and information and an example of each.



Typical 4-mark answer

Data doesn't have any meaning on its own, it's just the value. Information adds more detail to it, so it has a purpose. Knowledge is learning something from it.

An example of data is 20, information might be how many people like playing computer games and knowledge might be creating more appealing games.



What good things can we see in this answer?

1. Some key terms used: value, purpose.
2. Correct description of data, information and knowledge.
3. Correct examples of each given.



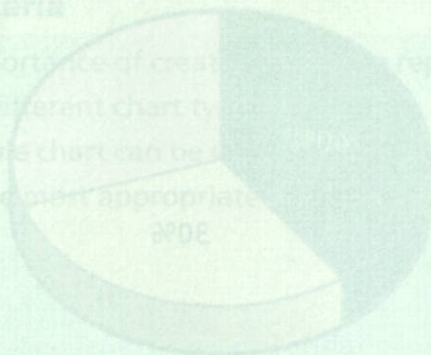
Which parts of the answer could be better?

1. The example of 20 people for information could be clearer, 20 out of how many?
2. The example of knowledge could be expanded.



How can we improve this answer?

1. Expand the information about 20 people to say 20 people out of 100, for example, giving an idea of sample size.
2. Expand the description of knowledge to talk about increasing the number of 20 out of 100 to a larger number.



What makes a good chart?

Related topics

- Surveys and questionnaires
- Primary data

Keywords

data

bar chart

chart

graph

line graph

pie chart

scatter graph

survey data

Learning objectives

1. Understand the purpose of a chart.
2. Describe examples of different chart types.
3. Understand how a pie chart can be used to display survey data.
4. Demonstrate how to select the most appropriate chart for a specific purpose.

The purpose of a chart or graph

In Unit 3 we looked at how the purpose of a **graph** or **chart** is to present **data** in a visually interesting way. This allows the reader to see patterns or make judgements on what they can see. A chart may present any of the following:

- Percentages of a whole
- Columns of different sizes, linking size to their value
- Changing values over time.

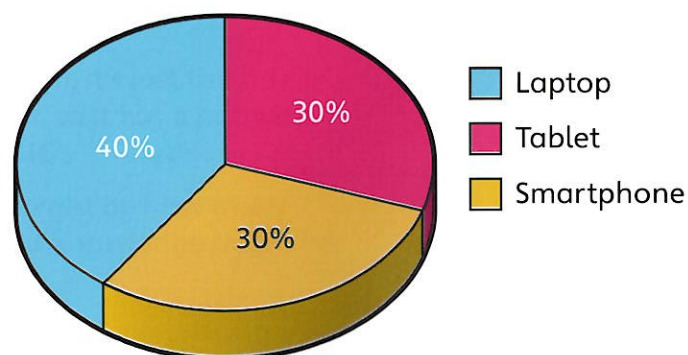
There are many types of graph, including:

- **Pie chart:** used to show individual parts of a whole – ideal for percentages.
- **Line graph:** shows changes in value(s) over time.
- **Bar/Column chart:** displays the value of a range of similar items against the same quantity – ideal for comparing prices.
- **Scatter graph:** displays two sets of data to see if there is a connection between them.

Using a pie chart to display survey data

Let's consider the survey in Lesson 2. It asked about the most popular type of device.

Pie chart showing popular device choices



Real-world examples

Most software applications have some sort of chart tool that will either recommend or automatically create a chart based on the data you select. The auto-tool is sometimes called the wizard.

Choose the most appropriate chart type

Consider the following survey questions and choose the most appropriate chart or graph to display the **survey data**.

Survey questions	Pie Chart	Line Chart	Bar/Column	Scatter chart
Money spent on technology per year.			✓ Comparing data of the same type.	
Average time spent on a device over the last ten years.		✓ Showing data changes over time.		
The average number of hours spent on a device for users of different ages.				✓ Comparing two data sets, such as age and hours, and looking for patterns.
The most popular computer device used by a range of people.	✓ Looking at percentages of a whole.			

Further investigation

- Try to create each of the charts listed and use the formatting tools.
- Research some of the other types of chart and graph not described in this lesson.

Success criteria

- I know the importance of creating charts to represent data.
- I know about different chart types.
- I know how a pie chart can be used to show percentages of a whole.
- I can choose the most appropriate chart for a given scenario.

Designing an infographic

Related topics

- Knowledge, data and information
- Surveys and questionnaires
- Spreadsheet models

Keywords

charts
graphs

infographic
pictogram
timeline

Learning objectives

1. Understand the purpose of an infographic.
2. Understand the key elements that make up an infographic.
3. Demonstrate how to choose suitable infographic content for a given scenario.

What is an infographic?

Briefly discussed in Unit 3, an **infographic** is designed to convey text, data and information in a visually interesting and straightforward way that grabs the attention of the reader. It might contain any of the following:

- Headlines, short blocks of text and bullet points.
- **Charts** or **graphs** formatted to suit the style of the document.
- Relevant diagrams, graphics or **pictograms** linked to the topic.
- **Timelines** that show either events or data progression over time.
- A clear, colourful house style that is suitable for the theme discussed or the style of the organisation that created the document.

Planning an infographic based on a survey

During this unit, a simple survey of ten people is carried out. It is about their use of technology, which has allowed us to collect data, information, and knowledge about the following:

- The popularity of different devices
- Average time spent on devices
- Amount spent annually on technology.



Real-world advice

When it comes to creating an infographic, you have a number of choices:

- Use word-processing or presentation software you are familiar with.
- Create charts and graphs in spreadsheet software and copy them into another document.
- Use an online infographic creator, which has ready-made templates to which you can simply add text and charts.

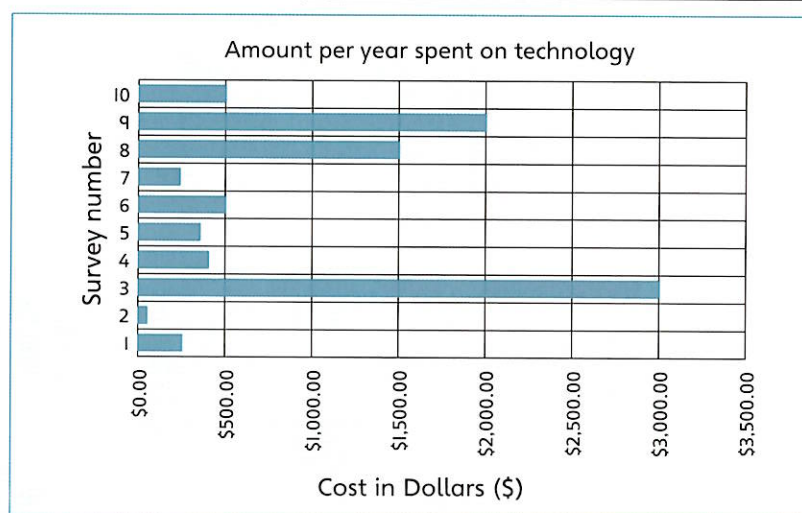
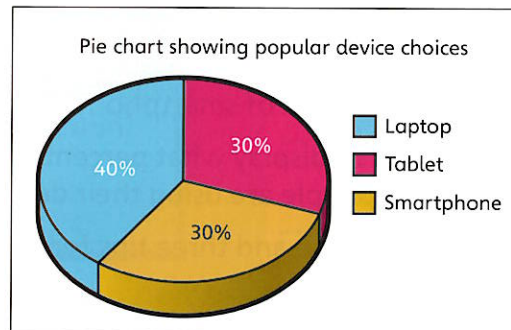


An example of an Infographic

The following example shows how survey results can be displayed using an infographic.

On average, \$1750 a year was spent on technology by laptop users. Those preferring smartphones spent only \$223 a year.

On average, people spend 3hrs and 20 minutes on their devices.



Real-world advice

There are no real rules when it comes to designing an infographic. Basically, it should be interesting to look at and informative. You have the freedom to add whatever you think is appropriate.

Further investigation

- Take a look at some online infographic creators and the tools they offer.
- Try creating your own infographic in either word processor or presentation software. Which do you think is easiest to use?

Success criteria

- I know the purpose of an infographic.
- I know the key elements that go into creating an infographic.
- I know how to create an infographic from data I have collected and analysed.

Unit 4 End-of-unit assessment

Typical 4-mark exam question

You are part of a team researching and presenting data to the public about the increased use of smartphones with young people.

You are creating a chart to display what percentage of certain times of the day young people are using their devices.

Describe a suitable chart type and three tips for ensuring the chart is clear and informative.



Typical 4-mark answer

The most suitable type of chart is a pie chart and it is really good for showing percentages.

Three tips would be:

1. Always add a clear title.
2. Make sure the percentages on the chart are large enough to see.
3. Make sure the names of each section can be clearly seen.

**What good things can we see in this answer?**

1. Some key terms used: pie chart, title, section.
2. Correct choice of pie chart.
3. Three correct tips provided.

**Which parts of the answer could be better?**

1. Tips could be a little clearer to include why they are important.
2. Some idea of how to carry out tips could be included.

**How can we improve this answer?**

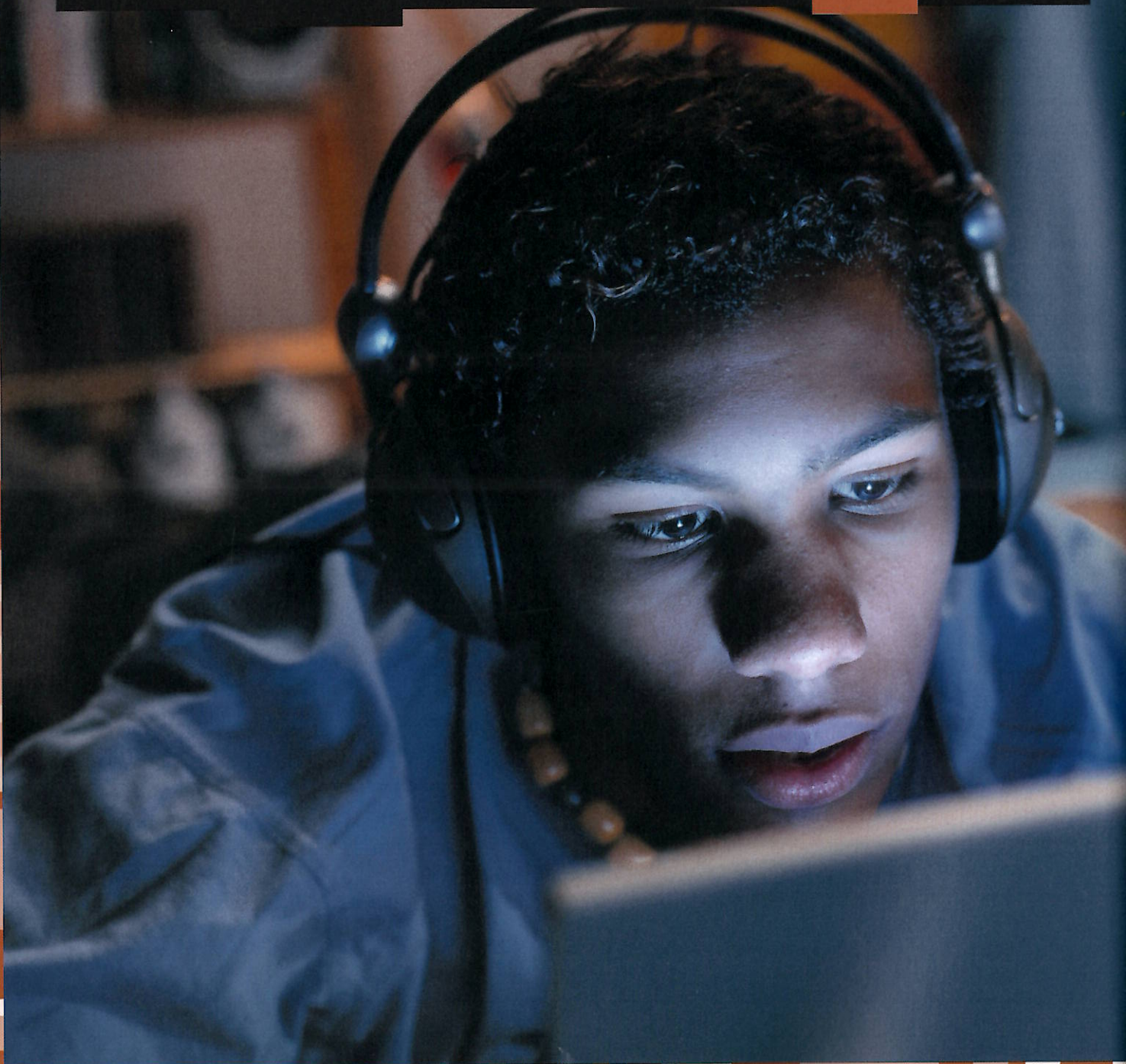
1. Describe why a title is important - that it provides context to the reader and explains its purpose.
2. In order to add percentages and names, describe that there are normally chart formatting tools that allow this.

**End-of-unit checklist**

- ☐ I know what a social network is.
- ☐ I know some of the advantages and disadvantages of social networks.
- ☐ I can create a good spreadsheet.
- ☐ I can transfer data from a simple survey to a spreadsheet.
- ☐ I know what the terms 'knowledge', 'data' and 'information' mean.
- ☐ I know the difference between primary and secondary data sources.
- ☐ I can use the sort and filter tools in a simple spreadsheet.
- ☐ I can describe the purpose and design of online advertisements.
- ☐ I know the importance of creating charts to represent data.
- ☐ I can choose the most appropriate chart for a given scenario.
- ☐ I know the purpose of an infographic.
- ☐ I know how to create an infographic from data I have collected.

Unit 5

Programming



Computers need instructions and programmers write these instructions. You will build on last year and improve your programming skills in this unit. From creating 2D shapes to interactive text programs, you will learn how to place one program inside another and work with data lists.

Key objectives:

1. Learn about how iteration can be used to create complete 2D shapes.
2. Learn about logic and syntax errors.
3. Be able to create more efficient programs.
4. Learn about the use of lists in programming.
5. Learn about how selection algorithms are used.
6. Learn about how automated chatbot software is used today.

By the end of the unit you will:

- create complex 2D shapes using Python
- create more efficient programs using iteration and selection
- be able to spot and resolve errors in your programs
- create a simple chatbot program
- create a missing word game, combining multiple programming techniques.

Algorithms and turtle programming

Related topics

- Creating original graphics

Keywords

algorithm
clear

commands
instructions
Python

sequence
turtle

Learning objectives

1. Understand how an algorithm can be used to create 2D graphics.
2. Demonstrate how to create simple 2D shapes using the Python turtle program.

A reminder on algorithms

An **algorithm**, as discussed in earlier units, is a step-by-step series of instructions. This can apply to any situation from making a cup of tea to plotting an assault course, to landing a rocket on the moon. Every algorithm requires a **clear**:

- starting point
- detailed **instructions** that can be followed without additional explanation.
- end point.

Creating 2D shapes in Python

In Year 7, you looked at creating simple graphics using the **turtle** module in **Python** – a built-in set of **commands** that allows the movement of a virtual pen around the screen. A basic 2D shape can be created using the following commands:

Command	Description	Example
name = turtle.Turtle()	Used to name the drawing turtle, for example marker	marker = turtle.Turtle()
Import turtle	Instructs Python to open the turtle drawing module.	Import turtle
fd(distance in pixels)	Forward by a distance in pixels	fd(100)
bk(distance in pixels)	backward by a distance in pixels	bk(100)
rt(turn angle)	Turn right by defined degrees	rt(90)
lt(turn angle)	Turn left by defined degrees	lt(90)
penup()	Lift up the pen, so no line is drawn when moved.	penup()
pendown()	Lower the pen, so lines can be drawn.	pendown()
pensize(size in pixels)	Sets the size of the drawing turtle.	pensize(100)
pencolor(size in pixels)	Sets the colour of the drawing turtle.	pencolor(red)

Please note: If you choose to name the turtle, each of the commands above should start with the name of the turtle followed by a full stop then the command, as shown in the examples on page 97.

Real-world advice

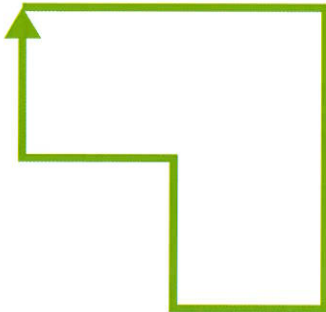
At the start of any turtle program, the turtle is at the centre of the screen, facing to the right, at the point 0,0.

Creating simple 2D shapes

The following shapes are all based on a **sequence** of step-by-step instructions, using the commands below, where each line follows the next and the program continues until it reaches the end.

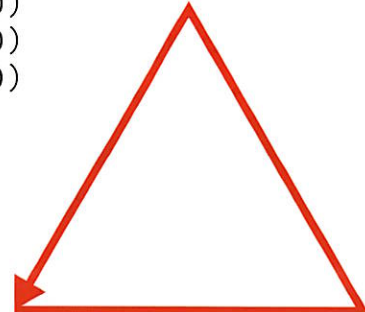
Example 1

```
import turtle
marker = turtle.Turtle()
marker.pencolor("green")
marker.fd(200)
marker.rt(90)
marker.fd(200)
marker.rt(90)
marker.fd(100)
marker.rt(90)
marker.fd(100)
marker.lt(90)
marker.fd(100)
marker.rt(90)
marker.fd(100)
```



Example 2

```
import turtle
marker = turtle.Turtle()
marker.pencolor("red")
marker.fd(200)
marker.lt(120)
marker.fd(200)
marker.lt(120)
marker.fd(200)
```



Real-world advice

In Year 7, the turtle module was imported slightly differently because there was no need to name the turtle pen when starting out. However, it is always good practice to name it as graphics become more complex.

Further investigation

- Create examples similar to the ones shown but also experiment with the pen size tool.
- Try creating two separate turtles, with their own names in the same program.
- Experiment with the pen up and down tool.

Success criteria

- I know what an algorithm is.
- I know how to name a turtle in Python.
- I know how to create simple 2D shapes, angles and distances.

Sequencing and iteration algorithms

Related topics

- Creating original graphics

Keywords

algorithm
code
efficiency

iteration

loop

random access memory (RAM)

sequencing

Learning objectives

1. Understand the terms 'sequencing' and 'iteration'.
2. Understand how sequencing and iteration can be used in Python.
3. Demonstrate the use of sequencing and iteration in a Python turtle program.

Sequencing and iteration

In Year 7, these terms were covered in the introduction to **algorithms**:

- **Sequencing**: a logical series of instructions, one followed by another from start to finish.
- **Iteration**: the use of **loops** or repeated actions until a specified point is reached.

A simple sequence

```
print("This is step one of three")
print("This is step two of three")
print("This is step three of three")
```

```
This is step one of three
This is step two of three
This is step three of three
```

An example of iteration

```
for letter in("Computing"):
    print(letter)
```

C
o
m
p
u
t
i
n
g

Real-world examples

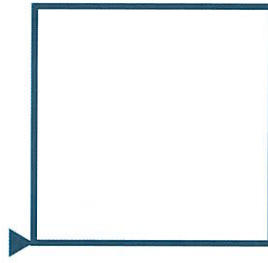
Using iteration can remove the lines of **code** required in any program, making it more **efficient** – reducing file sizes and reducing the impact on **random access memory (RAM)**.

Using iteration in Python

The examples of 2D shapes on page 97 are all based around sequences, where the instructions are carried out one line at a time, in order.

Iteration in Python is ideal for **repeated** patterns or shapes created by repeating a command, for example the creation of a square:

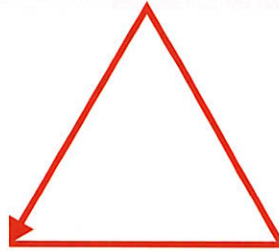

```
#Iteration square
import turtle
marker = turtle.Turtle()
marker.pencolor("blue")
for line in range(4):
    marker.fd(100)
    marker.lt(90)
```



Using iteration to create a more efficient turtle program

Consider the following 2D triangle– this could be made more efficient by using iteration to repeat lines. This example now has six lines of code whilst the original version had eight.

```
import turtle
marker = turtle.Turtle()
marker.pencolor("red")
for line in range(3):
    marker.fd(200)
    marker.lt(120)
```



📌 Real-world advice

Remember to note the position and direction of the turtle before writing additional code as it may need rotating first.

🔍 Further investigation

- Experiment creating shapes with repeating elements and different angles.
- Carry out research on the effect of file sizes on the RAM.
- Try creating the same shape using left turns followed by right turns.

★ Success criteria

- I know what the terms sequencing and iteration mean.
- I can use iteration to create a simple 2D graphic in Python.
- I know how to use iteration to make a program more efficient.

Programming errors

Related topics

- Logical thinking

Keywords

bug
computer program
debugging

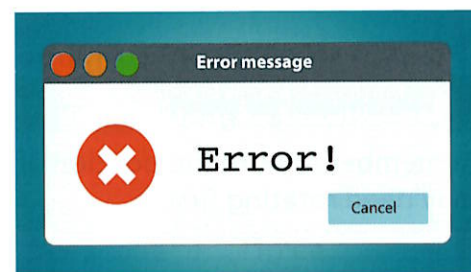
error
logic error
sentence case
syntax error

Learning objectives

1. Understand the importance of error checking in a program.
2. Understand the terms 'logic error' and 'syntax error'.
3. Demonstrate an ability to identify and resolve logic and syntax errors in a simple Python program.

Why is error checking important?

As humans, we are able to stop and think about any problem we come across. We can try to come up with a solution or speak to someone for advice/help. A computer just stops if it comes across a problem, or **error!** It doesn't have the ability to solve its own errors. This is why error checking any **computer program** is so important.



What are logic and syntax errors?

There are two types of common errors in programming:

- **Logic error:** a fault with the design of the program itself. This means the program may run but produce an incorrect output. These errors are obviously more difficult to spot and solve.
- **Syntax error:** a mistake within a program that causes the program to stop running. Each programming language has its own rules and a syntax error breaks one of these rules. Syntax errors are normally mistakes made by the programmer.

Error checking advice

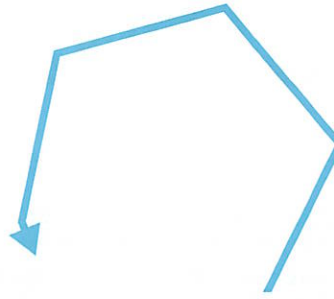
Error checking was discussed in Year 7. Here is a summary of the advice given:

- Check functions are spelled correctly.
- Check if the text is in **sentence case**, especially in functions and variables.
- Check characters and symbols.
- Check indentation, especially after an IF statement.
- Proofread a printed copy by hand.

Example of a logic error in Python

The following turtle program runs but does not produce the desired outcome. The programmer wanted to create a hexagon using a thick blue line but there are two logic errors.


```
#Hexagon
import turtle
marker = turtle.Turtle()
marker.pencolor("blue")
marker.pensize(10)
for line in range(5):
    marker.fd(100)
    marker.lt(65)
```

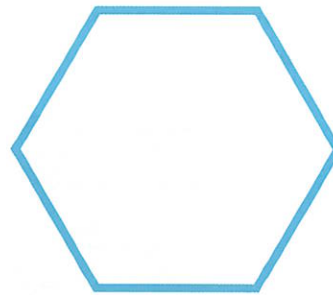


There are two logic errors in the program that cause the incomplete hexagon:

- The angle is set at 65, rather than 60 degrees.
- There are 5 loops used instead of 6.

The correct code should be:

```
#Hexagon
import turtle
marker = turtle.Turtle()
marker.pencolor("blue")
marker.pensize(10)
for line in range(6):
    marker.fd(100)
    marker.lt(60)
```



Examples of syntax errors in Python

Syntax errors are covered in detail in Year 7. Some typical errors in the program shown above might be:

- Incorrectly spelling 'turtle'.
- Forgetting to add the quotation marks around the colour "blue".
- Forgetting to add the indents after the 'for' command.

Real-world advice

Another term for solving errors in code is **debugging**, where the error is also referred to as a **bug**.

Further investigation

- Purposely add three errors to a code you have written and ask a partner to try and spot them.
- Investigate the term debugging.

Success criteria

- I know how important error checking is.
- I know what the terms logic and syntax error mean.
- I know how to spot and resolve a simple logic and syntax error.

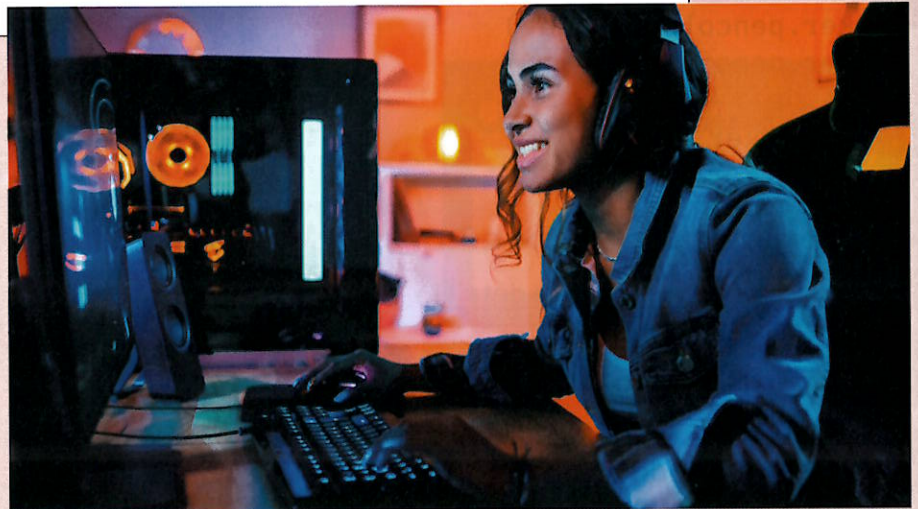
Unit 5 Mid-unit assessment

Typical 4-mark exam question

You are working as a program developer and carrying out testing on a simple 2D game.

Each time the program runs, the scoring variable is incorrectly multiplying the user's points when collected, instead of adding them.

State whether this is a logic error or a syntax error and describe the difference between them.



Typical 4-mark answer

The program is running, so it is a logic error. This is because the problem is a multiplication symbol has been used instead of an add symbol.

A logic error is a problem in the design of a program, made by the programmer. It might not stop the program from running. A syntax error is a mistake that is causing it to stop and display an error message because incorrect code has been used.



What good things can we see in this answer?

1. Good key terms used: design, programmer, stop, incorrect.
2. Correct choice of logic error.
3. Correct description of differences between logic and syntax.



Which parts of the answer could be better?

1. The term 'design of a program' could be clearer.
2. An example of a syntax error could be provided.



How can we improve this answer?

1. Expand the design of the program to say an incorrect output is produced but the program is working as instructed.
2. Include an example of a syntax error, such as an incorrect function spelling or a bracket has been used the wrong way round.

Using lists in a Python program

Related topics

- User interactivity

Keywords

index
interactivity

list

pseudorandom
random

response

user-generated

Learning objectives

1. Understand the purpose of using lists in a Python program.
2. Understand how to create and edit lists.
3. Demonstrate the use of lists in a simple Python program.

What are lists?

A **list** within Python is simply a group of objects, separated by a comma and held within square brackets. They have the following properties:

- Objects can be of any data type.
- Objects can be accessed by **index**, as used in Year 7. The first object is at position 0, the next at position 1.
- The list is mutable – this means it can be edited and re-ordered as required.

Creating and editing lists

Here is an example of three different user-generated lists:

```
#List with strings
names = ["Tariq", "Sasha", "Richard", "Vishna"]
#List with integers
heights = [46,13,15,47]
#List with floats
spaces = [1.23,5,43,2,6,135]
```

Using the middle list, here is how to delete one of the objects, the number 47.

```
#Delete the object at position 3
del heights[3]
print(heights)
```

[46, 13, 15]

Another object can also be added using the += command.

```
#Add and additional the object at the end
heights = [46,13,15,47]
heights += [50]
print(heights)
```

[46, 13, 15, 47, 50]

Making a random choice from a list

As already shown, a selection from a list can be made by printing the object at the required index position.

The following example shows how the **random** function can also be incorporated to allow the random selection of an object from a list, in this case suggesting a meal for that day.

The program shown asks the user to type in the food they had eaten on the previous day. This isn't used - it just adds **interactivity**.

A **response** is given and then a short delay is used to imply the computer is thinking. The final line chooses and prints a random item from the food list.

In order for the random and time programming to be used, they are imported at the start.

```
#Random food generator
import random
import time

food = ["Pizza", "Pasta", "Salad", "Fish", "A doughnut", "Soup"]
print("Let me choose a food for you")
meal = input("What did you eat yesterday?")
print("Interesting, I think today you should have...")
time.sleep(2)
print(random.choice(food))
```

Real-world examples

A truly random choice, such as rolling a dice or tossing a coin, is difficult for a computer to reproduce. The Python random function is described as **pseudorandom** because it is a computer-generated random number. It is suitable for programs like the one on this page but is not truly random.



Further investigation

- Create lists like the ones shown and experiment with editing them.
- Investigate the options for the random tool within Python.

Success criteria

- I know what a list is.
- I can create and edit a simple list.
- I can use a list within a simple program and print selections from it.

Use of selection in an algorithm

Related topics

- Arithmetic operators
- Relational operators

Keywords

algorithm

input

selection

Learning objectives

1. Understand the purpose of selection in an algorithm.
2. Understand how selection can be used in a visual or text-based language.
3. Demonstrate an ability to apply selection to a simple brief.

What is selection?

Selection is used within an **algorithm** when a question is asked, and when a response is required. The algorithm then makes a decision based on that response. The response could be any of the following:

- Yes or no
- True or false
- A calculation using arithmetic or relational operators
- A text-based response
- A value from a sensor, for example a thermometer.

A visual selection algorithm

This flowchart is an example of a simple selection algorithm.

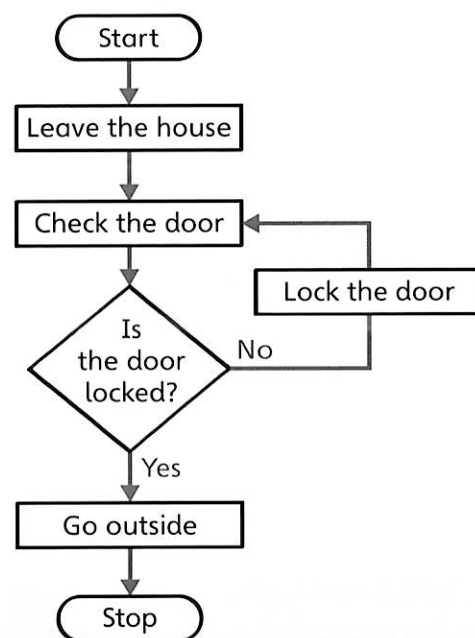
It requires a Yes or No response to a question we all ask when leaving the house.

The diamond shape in the centre is the question.

Question: Is the door locked?

No: Check the door

Yes: Go outside.



Creating a simple selection algorithm

Consider the following simple brief:

The new school heating system needs to be programmed. If the temperature in school drops below 18 degrees, the heating system should come on and stay on until the temperature reaches 18 degrees.

This could be broken down into pseudocode:

- IF temperature < 18, THEN turn heating on
- ELSE turn heating off



A simple selection algorithm in Python

After creating the simple algorithm, this can then be written as a program. In this example, the temperature can be entered by the user at any time.

```
#Selection algorithm
temp = int(input("What temperature is it in school?"))
if temp < 18:
    print("Thank you, heating is being turned on")
else:
    print("Thank you, heating not required")
```

The **input** command asks for the temperature, and this is stored as an integer as the variable temp.

The value of temp is compared to 18, if it is lower, the first statement is display.

Otherwise, the second statement is displayed.

Real-world advice

The term **THEN** is often used in pseudocode as part of a selection algorithm, but it is not required in Python programming.

Further investigation

- Investigate the full range of flowchart symbols and examples of their use.
- Create a simple flowchart for entering a computer password.
- Create an alternate version Python program, based around an age-checking system.

Success criteria

- I know what the selection algorithm is used for.
- I know how to create a visual selection algorithm.
- I know how to turn a simple brief into a Python program.

Creating a simple chatbot

Related topics

- Designing for a target audience
- User interactivity
- Use of computers in society

Keywords

artificial intelligence (AI)
chatbot
function

program
question
response
Turing Test

Learning objectives

1. Understand the concept of a chatbot.
2. Understand the key components of a simple chatbot program.
3. Demonstrate an ability to create a simple chatbot using Python.

What is a chatbot?

A **chatbot** is a computer program designed to simulate a two-way conversation between a real person and itself. This might be via:

- Text-based messages
- Text-to-voice based systems

A common form of chatbot are help systems on commercial websites that try to answer your help request, such as a lost package. If it cannot, it will transfer you to a real person.

The key components of a chatbot

A simple chatbot, written in Python **code**, should be able to do the following:

- Display a friendly welcome message
- Ask **questions** of the user and **respond** to the answers
- Include acknowledgement of user responses in their own responses.

Real-world examples

The **Turing test**, devised by computer pioneer Alan Turing in 1950, was created to challenge the ability of a computer to simulate a real person in questioning. He was interested in **artificial intelligence (AI)** and whether a computer's response could be seen as thinking.

A simple Python chatbot

The following example program is based on meeting these **functions**:

1. Chatbot asks the user their name.
2. Chatbot responds using the user's name.
3. Chatbot asks about the user's favourite place to be.
4. Chatbot responds with a random type of place from a **list**.
5. Chatbot asks the user if they have had a good day.

6. Chatbot responds according to the user response.

7. Chatbot closes the conversation.

```
#Chatbot
import random
import time
chatPlaces = ["the beach", "anywhere with snow", "any big city", "walking
in the hills"]
print("Welcome to Chatbot")
name = input("What is your name? ")
print("Hi there", name, "it's great to meet you!")
place = input("Where is your favourite place to be? ")
time.sleep(2)
print("That sounds great, I really love", random.choice(chatPlaces))
day = input("Have you had a good day? ")
day = day.lower()
if day == "yes":
    print("Good to hear that, me too")
else:
    print("Well I hope tomorrow is better")
print("Really enjoyed the chat, speak to you again soon. ")

Welcome to Chatbot
What is your name? Paul
Hi there Paul it's great to meet you!
Where is your favourite place to be? Out in the hills
That sounds great, I really love any big city
Have you had a good day? Yes
Good to hear that, me too
Really enjoyed the chat, speak to you again soon
>>>
```

Real-world advice

In the example above, at the end of an input question, you will see a space has been added. This is so when the program runs, there is a space between the question mark and the user response.

Further investigation

- Investigate the Turing test and the term 'artificial intelligence'.
- Discuss in a group some examples of chatbots you have used.
- Create your own version of the chatbot program, experimenting with different questions.

Success criteria

- I know what a chatbot is.
- I know the key features of a chatbot.
- I know how to create a simple chatbot.

Subprograms in an algorithm

Related topics

- Sequence, selection and iteration
- Computers in our daily lives
- Using relational operators

Keywords

algorithm
code

function
procedure
program
return

subprogram
subroutine

Learning objectives

1. Understand the purpose of a subprogram.
2. Understand how a subprogram can be used in real-life situations.
3. Demonstrate use of a subprogram in a range of scenarios.

What is a subprogram?

The purpose of a **subprogram** is to remove repetitive tasks from a larger **algorithm**, saving time and lines of **code**. A subprogram will be defined in the following way:

- It will be given a unique name.
- It is a self-contained **program** or module.
- It can be used in one or more programs.
- It can be called upon as and when required in the same program.
- Once the subprogram has been run, the main program will continue.

Where are subprograms used?

There are examples of subprograms in all the devices we use. Each function of a device has its own subprogram. These might include:

- The camera on your smartphone.
- The mathematical functions on your scientific calculator.
- The cooking programs in a microwave.
- The Wi-Fi settings on a games console.

Procedures and functions

There are two types of subprogram:

- **Procedure:** carries out a specific task, like reset a variable, before the main program continues.
- **Function:** once the subprogram has run, it will **return** a value that is used in some way in the main program, for example a unit conversion.

Real-world advice

- You will find subprograms are sometimes referred to as **subroutines**. In Python, subprograms are called functions.

Subprograms in our daily lives

Think about the actions you repeat in the same way on a daily basis. If your life was a long computer program, where might subprograms be used? Specific repetitive tasks might include:

- Brushing your teeth
- Making a meal
- Getting dressed
- Walking to school

Creating a subprogram in Python

The following example is a simple subprogram created in Python. It uses part of the random food generator created earlier in this unit. The program works in the following way:

- It asks the user if they are hungry.
- If the user says 'yes', the subroutine called 'hungry' is run and a random choice from the food list is printed.
- If the user gives any other answer, the program ends.

```
#sub program
import random

#def is used to define the sub program and give it a name
def hungry():
    food = ["Pizza", "Pasta", "Salad", "Fish", "A doughnut", "Soup"]
    return(random.choice(food))

#This is the main program
question = input("Are you hungry at the moment?")
if question == "yes":
    print("How about....", hungry())
else:
    print("OK, have a good day")
```

Real-world advice

In the example above, the term 'return' is used in the 'hungry' subprogram. This is used to return the value from the subprogram so it can be printed later.

Further investigation

- Investigate how subprograms are used in a computer device you use on a daily basis.
- Experiment with the example subprogram given in this lesson. Change the question and list to discuss different drinks.

Success criteria

- I know what a subprogram is.
- I can describe examples of subprograms.
- I can create a simple subprogram in Python.

Creating a missing word game

Related topics

- Sequence, selection and iteration

Keywords

algorithm
coding

input
string
user

Learning objectives

1. Understand how multiple programming techniques can be combined to create a complex program.
2. Demonstrate an ability to create a simple missing word game in Python.

Creating larger programs

In Year 7 and throughout this unit, you have learned about the following programming techniques:

- Using sequence, selection and iteration **algorithms**
- Creating and processing variables
- Using lists to store multiple objects
- Using subprograms to remove repetitive lines of code
- Using arithmetic and relational operators to compare variables
- Resolving syntax and logic errors.

In this following section you will combine multiple elements in a larger program – a simple word game.

The Missing Word Game brief

The following brief describes the overall idea behind the game:

*The game presents the **user** with three sentences and for each there is a missing word. Three possible missing words are provided and the user selects one. They are then told if they have got the question right or wrong.*



This can be broken down into the following key elements:

- Three missing word questions
- Each question requires an answer or an **input**
- The answers can be stored in a list
- The user selects an answer from the list
- The user is told if they have guessed the correct word.

🔗 Real-world advice

Notice how, when **coding**, #comments (e.g. **#This is question 2**) are used in the example to break up the structure of the program and highlight key areas.

The Python missing word game

```
#The Missing Word Game
import random
import time

choices = ["sun", "bike", "cat"]
print("Welcome to the missing word game\n")

#Sub program for question
def question():
    answer = input("\nWhich is the missing word?\nsun\nbike\ncat\n")
    return answer

#This is question 1
questionOne = "The beach is great when the _____ is out"
print(questionOne)
answer = question()
if answer == choices[0]:
    print("Brilliant, well done!")
else:
    print("Sorry, let's try another.")
time.sleep(2)

#This is question 2
questionTwo = "\nLast week I repaired the wheel on my _____"
print(questionTwo)
answer = question()
if answer == choices[1]:
    print("Brilliant, well done!")
else:
    print("Sorry, let's try another.")
time.sleep(2)

#This is question 3
questionThree = "\nA _____ always manages to land on its feet"
print(questionThree)
answer = question()
if answer == choices[2]:
    print("Brilliant, well done!")
else:
    print("Sorry, better luck next time.")

Welcome to the missing word game
The beach is great when the _____ is out
Which is the missing word?
sun
```



```
cat
bike
sun
Brilliant, well done!
Last week I repaired the wheel on my _____
Which is the missing word?
sun
cat
bike
bike
Brilliant, well done!
A _____ always manages to land on its feet
Which is the missing word?
sun
cat
bike
cat
Brilliant, well done!
```

Real-world advice

This program uses elements from all of the programs we have created so far, including **string** methods, indexing and subprograms.

Further investigation

- Print out the program and highlight the techniques mentioned in this lesson.
- Experiment with the program, changing the questions and answers.
- What other techniques could be used to improve the program?

Success criteria

- I know that a larger program can contain lots of programming techniques.
- I can create a simple missing word game in Python.

Unit 5 End-of-unit assessment

Typical 4-mark exam question

You are working as part of a team of programmers, writing applications for a smart speaker including alarms and notifications.

You have been asked to use subprograms where appropriate in code.

Describe the purpose of a subprogram and two benefits of using them when programming.

Typical 4-mark answer

A subprogram is used to carry out any repeated code in a longer program. It can be called on as required and only needs to be written once.

One benefit is it can reduce the size of the file and another is the same subprogram can then be copied to another program - both make the program more efficient.



What good things can we see in this answer?

1. Good key terms used: repeated code, file size, copied, efficient.
2. Description of subprogram is correct.
3. Two potential benefits have been provided.



Which parts of the answer could be better?

1. First sentence doesn't quite make the link to repeated code.
2. No reason as to why the file size is reduced.



How can we improve this answer?

1. Improve the first sentence to include that a subprogram is used to replace a block of code that may be repeated in the same program.
2. Explain that reducing the number of lines means using fewer characters and therefore less space is required to store it.



End-of-unit checklist

- ☐ I know what an algorithm is.
- ☐ I know how to create simple 2D graphics in Python.
- ☐ I know how to use iteration to make a program more efficient.
- ☐ I know how to spot and resolve simple logic and syntax errors.
- ☐ I can create and edit a simple list.
- ☐ I know what the selection algorithm is used for.
- ☐ I know how to turn a simple brief into a Python program.
- ☐ I know how to create a simple chatbot.
- ☐ I know what a subprogram is and how to create a simple subprogram in Python.
- ☐ I can create a missing word game in Python, combining lots of techniques.

Unit 6

Software and communication



In a discussion about the internet, it is impossible not to think about security and the dangers of allowing the internet into almost every aspect of our lives. By understanding the purpose of different types of computer software, network technology and how different countries use the internet in different ways, you can make your own informed opinion.

Key objectives:

1. Learn about computer surveillance.
2. Learn about application, operating system and utility software.
3. Learn about the devices we need to connect to the internet
4. Learn about NFC, Wi-Fi and Bluetooth
5. Learn about data breaches and cyberattacks.

By the end of the unit you will:

- make appropriate hardware and software choices when connecting to the internet.
- suggest suitable communication technology for a given scenario.
- be able to make software suggestions for a given scenario.
- be able to discuss reasons for and against increased computer surveillance.

Surveillance technology

Related topics

- Impact of technology on society
- The internet and the WWW

Keywords

facial recognition
hack
hardware

intercept
security
spyware
surveillance

Learning objectives

1. Understand surveillance technology in the context of computing.
2. Understand how technology is being used to create surveillance and security tools.
3. Understand a range of scenarios where surveillance equipment might be used.

What is computer surveillance?

The term **surveillance** describes the observation of someone without their knowledge. Data is collected and analysed with a purpose in mind. This generally refers to **security** services observing a potential criminal with the intention of observing a potential crime being committed.

Computer surveillance is the use of technology to monitor the activities, movements and communications of someone without their knowledge.

Surveillance and security technology

Here are some examples of how technology is used for computer surveillance.

- **Use of cameras and facial recognition:** cameras see us every day, from our streets, shops and road traffic cameras. Many of these are connected to the internet and, combined with **facial recognition** technology, our movements can be logged and tracked.
- **Smartphone and smartwatch data:** not only is our position available from mobile network towers when we use them but most also log their global positioning system (GPS) position for apps and services. Assessing this information can reveal the owner's exact location.
- **Spyware:** by installing special software, the keystrokes, documents and internet history can be recorded and examined. It is also possible to access the user's webcam without permission.
- **Intercepting** messages: if the messaging services of an individual are known, it is possible to hack and see messages being sent and received.
- **Social media sites:** by adding an individual as a friend, their online activities, other friends and their location can be analysed.
- **Hardware:** if it can be easily replaced, an external drive, USB stick, printer or any other device plugged into a computer can be adapted to spy on a computer.
- **Internet service providers:** if required, most law enforcement agencies can access the internet history of a user via their internet provider.

Real-world examples

With many more people working remotely and from their own home, large companies are experimenting with ways to observe their employees. Examples of employer surveillance include:

- Monitoring the number of emails being sent.
- Observing the number of times a user logs on and off.
- Analysing cloud-based documents to see when they have been worked on by an employee.
- Observing group chats and messaging.
- Watching video call meetings.

Surveillance scenarios

The following scenarios consider how technology might be used:

Scenario	Potential surveillance
Reading the messages sent between two people	■ Intercepting emails and messaging apps ■ Installing spyware on one person's device
Tracking a person across a city	■ Face recognition using CCTV ■ Smartphone tracking via mobile phone network tower or GPS from apps
Looking at files on an individual's computer	■ Installing spyware or hacking online document folders

Real-world advice

Be wary of free smartphone applications that seem too good to be true. For example, free games, movies or functionality. There have been examples of apps that also act as spyware, which can steal personal data from your device.

Further investigation

- Investigate the history of computer surveillance.
- How many devices do you have access to that could potentially give away your personal information or location?

Success criteria

- I know what the term 'computer surveillance' means.
- I can describe examples of technology used to observe someone without their knowledge.
- I can describe examples of how this technology might be used.

Types of software

Related topics

- Computer systems

Keywords

application software

computer-aided design

desktop publishing

graphics software

image editing

office applications

open source

operating system

software

sound editing

utility software

Learning objectives

1. Understanding of the terms 'application', 'operating' and 'utility software'.
2. Describe examples of each type of software.
3. Demonstrate an ability to choose the most appropriate software type for a given scenario.
4. Understand how the same software types apply to all computer-based devices.

Types of software

The term '**software**' can refer to three different categories:

- **Application software:** these are programs we use for day-to-day tasks, including word processing, **desktop publishing** or playing games.
- **Operating system:** the link between the hardware, the computer itself, and the software programs we use every day. The operating systems allow devices to communicate and application or utility software cannot be installed without one.
- **Utility software:** software that supports the operating system and performs specific tasks. This might be checking for and removing viruses or making sure the hard disk is running efficiently.

Software examples

The table below describes examples of each type of software:

Application	Operating system	Utility
Web design	Windows based systems	Anti-virus/malware
Graphics software	Apple based systems	Disk-defragmentation
Image editing	Linux based systems	Back-up
Video games		Compression
Video and sound editing		Encryption
Computer aided design (CAD)		Formatting tools
Office applications: word processing, spreadsheet, presentation or databases		

Real-world examples

The first versions of operating systems we would recognise were Apple Macintosh OS version 1.0 in 1984 and Windows 1.0 in 1985.

Choosing the right software

The table below describes the most appropriate software type for a given scenario.

Scenario	Software type	Reason
Concerned that a computer may have a virus.	Utility	Anti-malware software, designed to scan a computer and remove any evidence of a virus.
Having just built a new computer from bought components.	Operating system	Required to allow all components to communicate.
Wanting to type up a school project.	Application	Any software with a user-based purpose, for example to type or draw something, is an application.

Software in a range of devices

It isn't just computers and laptops that have software installed. Below are some examples of devices that still require software to run:

- Smart speaker: will contain an operating system and many have the ability to install applications with a specific purpose, such as linking to a heating system.
- Modern car: an operating system is installed for engine management, in-car entertainment and climate control.
- Smart television: applications for streaming services and games can be installed.

Real-world advice

There are many examples of software that are classified as **open source**. This means they have been written to be freely shared, distributed and edited by anyone at no cost. It is possible to install and run a modern computer completely using free software.

Further investigation

- Consider the software you use on a daily basis; can it be classified?
- Investigate the increasing number of smart devices with software requirements.
- Investigate the term 'open source software'.

Success criteria

- I know the difference between an application, an operating system and utility software.
- I can describe examples of each type of software.
- I can suggest the most appropriate software for a given scenario.

Operating systems

Related topics

- Computer software
- Impact of technology on society

Keywords

command line interface (CLI)
dialogue
gesture/touch
graphical user interface (GUI)
memory management

multi-tasking
operating system
peripherals
security
user interface

Learning objectives

1. Understand the purpose of an operating system.
2. Understand the key roles of an operating system as part of a whole system.
3. Understand the different types of operating system.

Why do we need an operating system?

An **operating system** is the link between all aspects of the computer and the user. Without it we couldn't do any of the following:

- Install applications or games from other creators.
- Install a printer, microphone, webcam or any other peripheral device we wish to add.
- Update our security settings to stay safe online.
- Check that internal components are running effectively.



Key roles of any operating system

At the centre of a computer system, the operating system has a number of roles, which include:

- **User interface:** essential for the user, it is the on-screen display that allows the user to interact with the computer, move the mouse and see applications.
- **Memory management:** allocating memory space to programs and data. Without management, the system would quickly crash and become unusable.
- **Peripherals:** allow applications to interact with devices connected to the computer, from mouse control to printing or playing music through speakers.
- **Multi-tasking:** allows two or more applications to run at the same time. For example, the user might be writing a report whilst listening to internet radio.
- **Security:** Monitoring and controlling access to programs and data and preventing any unauthorised access to the system.

Different types of operating system

The three most common operating systems for computers used in the home, school or workplace are Windows, Apple Mac and Linux. Normally, when we think of an operating system we think of a desktop or a laptop computer, but an operating system is installed in all modern computer devices, including:

- smartphones
- tablets
- smart speakers
- games consoles
- televisions.

How we interact with an operating system

We can interact with a computer through its operating system in any of the following ways:

- **Graphical user interface (GUI):** this is the most common. It allows the user to move a cursor around the screen and interact with icons and menus.
- **Command line interface (CLI):** early operating systems, and some specialist devices, use a command line interface. This means all commands are entered using text commands.
- **Dialogue:** think about how many devices can now be accessed using our voice, from smart speakers to in-car entertainment. Smart devices can be linked to other systems to control other household devices.
- **Gesture/touch screen:** although we can type into our smartphones and tablets, the majority of our interaction involves swipes, finger pinches and taps. These gestures are now becoming standardised across many devices.



Q Further investigation

- Investigate the history of the modern operating system.
- Think about the devices you use regularly. How do you interact with them?
- Discuss how your class might interact with an operating system in the future.

★ Success criteria

- I know what an operating system is for.
- I know why we need an operating system.
- I know about the different types of operating system.
- I know about the different ways we can interact with an operating system.

Utility software

Related topics

- Operating systems
- Computer software
- Impact of technology on society

Keywords

anti-malware
anti-spyware
antivirus

back-up
compression
defragmentation
disk organisation

file transfer
firewall
maintenance

Learning objectives

1. Understand the purpose of utility software.
2. Understand the different types of utility software.
3. Demonstrate an ability to choose the most appropriate utility software for a given scenario.

Why do we need utility software?

There are some functions that the operating systems may not include but are useful in running the computer. **Utility software** performs these useful tasks and they often run in the background. Although designed for a specific operating system, they are often third-party programs, which means they have been designed by a different company. Utility software based around security is effective if kept up to date. New computer viruses, for example, are created every day and so antivirus software must also be set to auto update as often as possible in order to be effective.

Utility software usually falls into one of three categories:

- **Security:** preventing unauthorised access to your computer or repairing any damage done by malicious software.
- **Disk organisation:** ensuring files are organised, disks are correctly formatted and file space isn't wasted.
- **Maintenance:** keeping an eye on system updates, running back-ups or removing unwanted files and applications.

Real-world examples

The first computer virus was written in 1971. It was written as the first 'worm' virus, meaning it could copy itself onto computers connected to one another. It was written as a security test to see if it could work. In 1974 the first malicious computer virus was created and was able to spread between computers, slowing them down until they became unusable.

Types of utility software

The following table outlines some of the most common examples of utility software:

Category	Utility software	Description
Security	Antivirus	Scans for and blocks viruses that will try to access your computer through email attachments or suspicious websites.
	Anti-spyware/ malware	Scans for and tries to remove any malicious software that may be causing damage or spying on your documents.
	Firewall	Blocks unwanted access to your computer from the internet.
	Encryption	Making sure no one can access your documents without a secret key and if they do, it will appear meaningless to them.
Disk organisation	Disk formatting	Operating systems often require disk drives to be formatted in a certain way to work. This also includes any extra storage you may add.
	Disk defragmentation	Files on your computer are rearranged in a logical way to speed up your computer and restore wasted space.
	File transfer	Used to upload and download files to a website or network drive.
	Compression	Useful for saving file space by compressing one or more files. Also used when emailing multiple files as one compressed attachment.
Maintenance	System information	Used to analyse how a computer is running, where memory is being used and if the central processing unit (CPU) is running too fast.
	Back-up	Creating a duplicate version of your files and programs to be stored away from your computer in the event something goes wrong and then your computer can be restored.
	System clean-up	This will look for old un-used applications, hidden installation files and anything that can be safely removed to regain space.

Q Further investigation

- Discuss with your teacher and class, how many of the utility programs do you know?
- Investigate the computers you use, do they have up-to-date security software installed?

★ Success criteria

- I know what the purpose of utility software is.
- I can describe different types of utility software.
- I can choose utility software for a specific purpose.

Internet hardware

Related topics

- Impact of technology on society
- The internet and the WWW

Keywords

data packet
devices
home network
internet
internet protocol address (IP)

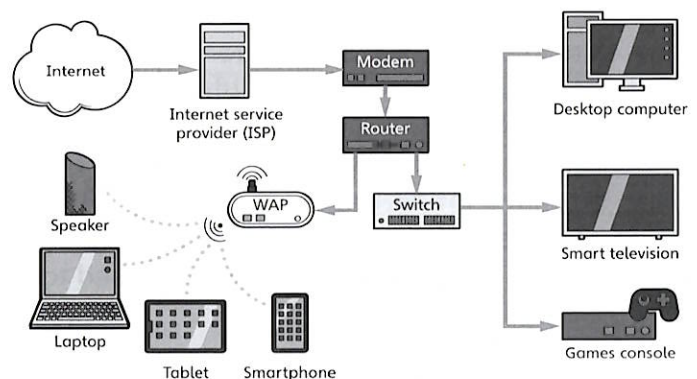
internet service provider (ISP)
modem
network interface card (NIC)
switch
wireless access point (WAP)

Learning objectives

- Understand the hardware required to connect to the internet.
- Understand the role of some of the key hardware devices.
- Demonstrate an ability to choose the required hardware for a given scenario.

How do we connect to the internet?

In Year 7, you considered some of the **devices** required to connect to the **internet**. These devices vary depending on the situation and connection requirements. The diagram below represents a typical home internet connection:



Essential hardware

The table below describes the purpose of the hardware in the diagram.

Hardware	Description
Router	Used to connect networks together, in this case the home network and the internet. It assigns an internet protocol (IP) address to all connected devices .
Modem	A device used to connect a computer to the internet.
Switch	Connects directly to the router and directs data to and from specific devices.
Wireless Access Point (WAP)	Provides a connection for wireless devices to the local network.
Smart devices	These can connect to the internet for additional functionality, for example streaming.
Network Interface Card (NIC)	Allows a computer or laptop to connect to a wired or wireless network

Real-world examples

The modem, router, switch and wireless access point all have individual roles but in many home networks, these may be combined into one or two devices, provided by the **internet service provider (ISP)**.

Data packets

In Year 7, you studied the structure of a **data packet** and how data is transmitted across a network:

- **Packet header:** information about where the data is from, where it is going and its position in the complete message.
- **Packet body:** the actual data content (part of the complete message). The packet body is sometimes called the payload.
- **Packet footer:** this signals the end of the data packet and checks the packet is complete.

Connection scenarios

The table below describes the route that data would take for a given scenario.

Scenario	Description of route
Listening to internet radio on a smart speaker	Internet to speaker via ISP to modem, router, switch and WAP
Sending word processed files between a laptop and desktop computer	Laptop to computer via WAP and switch
Sending video from a tablet to a laptop	Tablet to laptop via WAP and switch
Uploading a new profile picture from a smartphone to a social media site	Smartphone to internet via WAP, switch, router, modem and ISP

Further investigation

- Investigate your own network or a typical home network. How similar is the setup to the one shown in this lesson?
- Revisit the information on data packets in Year 7.
- Research what other hardware devices can connect to the internet in our homes.

Success criteria

- I know how most homes connect to the internet.
- I know about the hardware devices used to connect to the internet.
- I know how data can be transmitted between a home and the internet.

Unit 6 Mid-unit assessment

Typical 4-mark exam question

As a software specialist, you have been employed by a new business to specify software requirements for their new head office.

Explain the purpose of operating and utility software and describe an example of each.



Typical 4-mark answer

Operating system software is essential for every computer to run, it allows the different components inside to talk to each other. An example of an operating system is Windows.

Utility software is designed for supporting tasks, such as defragmenting a hard drive or scanning for viruses.



What good things can we see in this answer?

1. Good selection of key terms used: components, Windows, supporting, defragmenting, hard drive, virus.
2. Descriptions of each software given.
3. An example of both operating (Windows) and utility (antivirus) software are both given.



Which parts of the answer could be better?

1. The phrase components could be expanded to include examples.
2. The phrase supporting tasks could be explained a little more.



How can we improve this answer?

1. Describe some examples of components, such as the motherboard, CPU, etc.
2. Explain that the operating system is also the link between the user and the computer.
3. Expand the term supporting tasks by mentioning running in the background.

Wireless communication technology

Related topics

- Impact of technology on society
- The internet and the WWW
- Privacy and security

Keywords

bluetooth
connection

NFC (near-field
communication)

Wi-Fi

wireless

Learning objectives

1. Understand the purpose of wireless communication technology.
2. Understand the purpose of near-field communication (NFC), Wi-Fi and Bluetooth Describe some of the advantages and disadvantages of each technology.
3. Demonstrate an ability to choose the most appropriate technology for a given scenario.

What is wireless communication technology?

Wireless communication technology allows data to be transmitted and received between compatible devices without the need for a wired connection. The **connection** speed and amount of data needing to be sent determines the most appropriate technology.

Wi-Fi

You studied the many advantages and disadvantages of **Wi-Fi** in Year 7. It is a secure, medium range, wireless technology, with a range of around 50m that is ideal for connecting computers to the internet. Examples of Wi-Fi use include:

- Home wireless networks: connecting laptops and smart devices to the internet.
- Schools and offices: connecting devices and networks to the internet.
- Shops and leisure facilities: providing internet access for smartphones.

Near Field Communication (NFC)

Near-field communication (NFC) is a short range wireless technology with a range of only a few centimetres. It allows data to be communicated between two devices placed next to each other. This makes it ideal for the following:

- Contactless payments: paying for goods and services using a smartphone or smartwatch.
- Transferring personal information between smartphones, such as contact details.
- Checking into an event such as a concert with a smartphone at an entry point.

Bluetooth

Bluetooth is also a short range technology with a range of a few metres. It is used to send and receive data in smaller portable devices. It is ideal for sending instructions or streaming audio files and used commonly in:

- Wireless headphones and speakers
- Wireless keyboards and mice
- Video game controllers.

Advantages and disadvantages of wireless communication technology

	Advantages	Disadvantages
Wi-Fi	■ Ideal in locations ethernet cables cannot be used. ■ Passwords can be used to keep connections secure and private.	■ Signal can be blocked by walls and electrical devices can cause interference. ■ Not as fast as a wired connection.
NFC	■ Can be built into multiple devices for easy payment. ■ Convenience of contactless payments.	■ Some security concerns, such as money being taken without consent (There is very little evidence of this and passkeys can be added to smartphones).
Bluetooth	■ Compatible with many devices, even from different manufacturers. ■ Very little power required to maintain connection.	■ Limited range and only suitable for small amounts of data. ■ Danger of hackers accessing devices, especially if left on at all times.

Choosing a connection

The examples below show which connection type is the most appropriate to different scenarios:

Scenario	Wi-Fi	NFC	Bluetooth	Reason
Checking into a music concert venue with a virtual ticket.		✓		A smartphone containing the ticket can easily be waived over an NFC sensor on entry.
Accessing the internet in a hotel room.	✓			Passwords can allow many devices to access a network.
Connecting a smartwatch to a smartphone.			✓	The connection will not drain the battery and they are close to each other.

Q Further investigation

- Check that the wireless devices you use are using a strong password.
- Investigate the history and uses of NFC.
- Investigate how bluetooth is also used in cars today.

★ Success criteria

- I know why wireless connections are important.
- I know what NFC, Wi-Fi and bluetooth are used for.
- I know some advantages and disadvantages of wireless connections.
- I can describe some scenarios where wireless connections could be used.

The internet and the World Wide Web

Related topics

- Impact of technology on society
- Computer networks

Keywords

data packet
domain name system (DNS)
hosting
Hypertext mark-up language (HTML)

internet
internet protocol address (IP)
internet service provider (ISP)
uniform resource locator (URL)
web browser
web server
world wide web (WWW)

Learning objectives

1. Understand the difference between the internet and the world wide web (WWW).
2. Understand the process of requesting access to and connecting to a website.
3. Understand the key hardware and software components that make the internet possible.

The difference between the internet and the WWW

In Year 7 you studied some of the key differences. In summary:

- **The internet:** The network of interconnected networks all around the world. It allows data to be sent and received from compatible devices anywhere where there is an internet connection.
- **The World Wide Web (WWW):** How we view web pages today. **Hypertext mark-up language (HTML)** is the programming language used to create content that can be viewed and interacted with using a **web browser** on any compatible device.

How do we connect to a website?

Below is the sequence of events that happen when you type a website address into a web browser:

1. An address is typed into a web browser (www.pearson.com, for example).
2. The web browser accesses the internet via a **router**, modem and an **ISP** (internet service provider) as shown earlier in this unit.
3. The web browser contacts a **domain name system (DNS)** server and asks about the website address.
4. The DNS server looks up the **IP address** of the **web server** that is **hosting** the site. It will keep asking other DNS servers until it finds it.
5. The DNS server sends the IP address of the web server back to the browser.
6. The browser contacts the IP address to access the website held on that web server.
7. The requested website is broken into **data packets** that each take the most efficient route possible back to the web browser.
8. The data packets are reassembled and the website is displayed in the web browser.



Real-world advice

The reason we have word-based website addresses, also known as uniform resource locators (URLs) is because they are much easier to remember than an IP address. Many users now don't try and enter the correct web address as it is often quicker to search for the name of the site in a search engine and then click the correct link.

What makes the internet possible?

The following elements make it possible to access websites held on remote web servers:

IP Address	The internet protocol address is the identifying address assigned to every network-connected device.
Internet Service Provider (ISP)	A company that provides internet access to homes, schools and organisations. It will often provide a modem and router to the customer.
Hosting	A term that describes the process of running a website on a web server. The server hosts the website.
DNS server	The domain name server is an international naming service that links internet addresses with an IP address. These links are stored on DNS servers around the world.
Web server	A network-connected computer server that stores website files that can be accessible by users of the internet.

Real-world examples

There are only 13 root DNS servers, named A to M, placed around the world. Run by 12 different organisations, they are at the core of our internet naming system.

Further investigation

- Investigate the history and future of HTML.
- Discuss in a group the question 'How do you get to a site you know?'
- Investigate the history of IP addresses and how they have been designed for the future.

Success criteria

- I know the difference between the internet and the WWW.
- I know how we connect to a website using a web address.
- I know about the key elements that make the internet possible.

Surveillance ethics

Related topics

- Impact of technology on society
- Computer networks
- Hardware and software

Keywords

closed-circuit television (CCTV)
eavesdrop
ethical

geolocation information
ID cards
legal
moral
research

Learning objectives

1. Understand the reasons given for increased surveillance and the tracking of individuals.
2. Understand some of the moral and ethical questions being raised on this topic.
3. Understand some of the arguments for and against surveillance technology.
4. Demonstrate an understanding of the importance of researching controversial topics.

Why is surveillance increasing?

We live in a world where technology is becoming more sophisticated and increasingly accessible. This has led to an increased use of technology-based **surveillance** around the world by both governments and commercial organisations. Examples of the technology used are described earlier in this unit.

Here are some of the reasons given by government for increased surveillance:

- To help solve crime: spot crimes being committed and track those involved.
- To prevent terrorism: identify individuals of interest as they travel and communicate.
- To solve financial crimes: identify criminal money transfers and money being moved.
- To increase personal safety: watch us, so help can be quickly provided if needed.
- To confirm our identity: computer linked **ID cards** can confirm our identity.
- To help us stay safe online: cybersecurity experts monitor the systems we use.

Here are some of the reasons given for increased commercial surveillance:

- Shopping suggestions: by analysing our shopping, products that might be of interest can be suggested to us.
- Social media: by tracking our friends and activities, sites can suggest friends, groups and activities we may find interesting.
- Security: if commercial and financial organisations share our details, any suspicious activity can be spotted and prevented.

Real-world examples

One of the first security **closed-circuit television (CCTV)** cameras was in Germany in 1942. Now around one billion security cameras are active, mainly China.

Moral and ethical questions

Members of any society agree to follow an established set of rules. If these rules are changed, what are the advantages and disadvantages to these changes? Increased surveillance has raised **moral** and **ethical** questions around the world.

- Is it **legal** for us to be observed 24 hours a day?
- Is it OK for others to **eavesdrop** on my conversations?
- What is happening to all the data about me?
- Can I ask to see who is observing me?
- What if I don't want to be observed?
- Why can't my internet searches be private?
- I haven't done anything wrong, so why am I being observed?
- Is surveillance worth losing our freedom?
- Do I want someone to know my **geolocation** information at all times?



Arguments for and against increased surveillance technology.

Here are some of the common arguments used about increased surveillance technology.

For	Against
Increased security and data protection	Lack of freedom
Being able to stop criminal activity and cyberattacks	People may be incorrectly charged with a crime
Better social and shopping experiences	Concerns about data sharing
More secure online shopping and banking	What if data is stolen?
Governments must keep up with criminals	Information being used without our permission

📌 Real-world advice

There are no answers to moral and ethical questions in this book, only knowledge and facts. To make a personal judgement, it is essential to thoroughly **research** the facts behind any subject and the reasons for and against. Then make an informed choice of your own.

🔍 Further investigation

- Discuss in a group some of the issues in this lesson.
- Investigate how surveillance might be used in your school.
- Read some terms and conditions of websites about using your information.

★ Success criteria

- I know why the use of surveillance technology is increasing.
- I know some of the reasons given for increasing surveillance.
- I know some of the moral and ethical questions that are being raised.
- I know some of the arguments for and against increased surveillance.

Data breaches and cyber-crime

Related topics

- Impact of technology on society
- Computer networks
- Hardware and software

Keywords

cyberattack
data breach

phishing

portable storage device

unpatched software

Learning objectives

1. Understand the terms 'data breach' and 'cyberattack'.
2. Understand the effects of data being stolen.
3. Understand the technical weaknesses that leave a computer vulnerable.
4. Understand some of the prevention strategies that can be used.

What is a data breach?

A **data breach** is the access of stored data without the permission of those holding it. This data may be private or confidential and those accessing it do it to either sell it for financial gain or to share it with others for idealistic reasons. Examples of data breaches include stealing:

- Usernames and passwords
- Credit card details
- Personal details
- Social network content
- Cloud-based documents.

What is a cyber attack

A **cyberattack** is an attack on an electronic system with the purpose of causing damage or gaining access to it. Examples of cyberattacks include:

- Crashing a website so it cannot be accessed
- Spreading a computer virus to damage multiple computers
- Causing an online service to become unusable
- Gaining access to a system and stealing data stored within it
- Tricking users into giving up personal information using email or messaging scams.

Real-world examples

In 2013, Yahoo was the victim of a data breach in which all three billion of its accounts were accessed! It wasn't clear if information was stolen or used, but all users were urged to check their accounts and change their passwords.

What can happen if data is stolen?

Following a data breach, these are many possible consequences:

- Usernames and passwords may be shared or even sold online, leading to identity theft.
- Private information can be shared without permission.
- Users can be blackmailed and threatened with their personal information being released unless a ransom is paid.
- People will lose confidence in a system, causing the owners financial loss.

What is a technical weakness?

A data breach or cyberattack is often possible because of weaknesses in the system, including:

- Poor password protection: users with important access may have an easily guessed password.
- **Unpatched software:** operating systems require regular security updates. These prevent known weaknesses and failing to update any computer leaves it open to attack.
- **Eavesdropping:** by intercepting messages, phone calls or even recording conversations, a hacker might gain information that can be used to access a system.
- Use of **portable storage devices:** a USB stick or memory card can contain a virus or spying software that can be activated if plugged directly into a network.

How can we prevent data being stolen?

The following can help prevent such attacks taking place:

- Using strong passwords that cannot be guessed.
- Always keeping computers updated, including operating systems and antivirus software.
- Not sending sensitive information via email.
- Not allowing unknown devices to be plugged into a computer.

Real-world advice

One of the most common ways our data is stolen is via **phishing** emails. These emails look like they are from a familiar organisation, asking the user to update their details. When the user does click, it will use a fake site to steal information. Links within emails should never be clicked.

Further investigation

- Investigate the laws where you live that relate to data breaches.
- Choose a website you are familiar with and see if it has been the victim of a data breach.
- Are all your devices updated? Discuss with your friends and family.

Success criteria

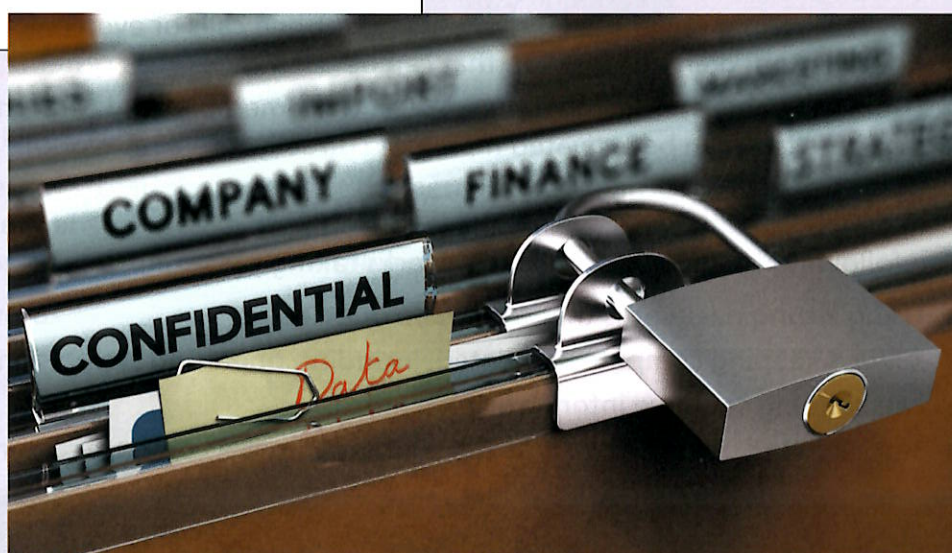
- I know what the terms 'data breach' and 'cyberattack' mean.
- I know what can happen if data is stolen.
- I know what the term 'technical weakness' means and can describe examples.
- I can describe some strategies for protecting systems from cyberattacks.

Unit 6 End-of-unit assessment

Typical 4-mark exam question

You are working as a security advisor to multiple businesses in an online tutorial. You are going to talk about how a data breach can happen.

Describe two examples of how data can be stolen and provide a prevention strategy for each



Typical 4 mark answer

Data can be stolen in a number of ways, including having a poor password, if it can be guessed then someone could access the system. A piece of spyware could also be installed by someone bringing in an infected USB stick.

To prevent data breaches, it is important to have very strong passwords and not allow any external devices to be plugged into the network.

**What good things can we see in this answer?**

1. Some good key words used: poor password, access, spyware, infected, external devices.
2. Two examples of data breaches: poor passwords, spyware.
3. Two preventions: strong passwords, do not allow external devices.

**Which parts of the answer could be better?**

1. Phrases 'poor' and 'strong' password could be explained.
2. Phrase 'infected' could be expanded.

**How can we improve this answer?**

1. Describe what a poor password is: small number of letters or a common password that can easily be guessed.
2. Describe what a strong password is: mixture of eight (or preferably more) random letters, numbers and symbols.
3. Explain that the USB stick could have become infected by accident and the user, unaware it now has spyware on it, could transfer infection to a business computer.

**End-of-unit checklist**

- ☐ I know what the term 'computer surveillance' means.
- ☐ I can describe examples of technology used to observe someone without their knowledge.
- ☐ I know the difference between an application, an operating system and utility software.
- ☐ I know what an operating system is for.
- ☐ I know what the purpose of utility software is.
- ☐ I know how most homes connect to the internet.
- ☐ I know about the hardware devices used to connect to the internet.
- ☐ I know why wireless connections are important.
- ☐ I know what NFC, Wi-Fi and bluetooth are used for.
- ☐ I know the difference between the internet and the WWW.
- ☐ I know about the key elements that make the internet possible.
- ☐ I know why the use of surveillance technology is increasing.
- ☐ I know some of the arguments for and against increased surveillance.
- ☐ I know what the terms 'data breach' and 'cyberattack' mean.
- ☐ I can describe some strategies for protecting systems from attack.

Glossary

ad-blocker	a tool that blocks advertising content when browsing the internet
advertisements	pictures, sets of words or short films that are intended to persuade people to buy a product or use a service
advertising	the activity or business of advertising things on television, in newspapers, online, etc.
algorithm	a set of instructions that are followed in a fixed order and used for making a computer program
alignment	the state of being arranged in a line with something or parallel to something, for example, the text within a column of cells lining up at the left or right margins, or in the centre
alphabetically	relating to the alphabet; for example, when words are listed by their first letter in the order in which they appear in the alphabet, they are said to be ordered alphabetically
anti-malware	software that is designed to block malware (software designed to get onto a computer and cause harm to the information there or problems with the way it works)
anti-spyware	software that is designed to block spyware (software that secretly records information about which websites you visit)
antivirus	software designed to block a computer virus (a set of instructions secretly put onto a computer or computer program, which can destroy information)
application software	programs we use for day-to-day tasks, including word processing, desktop publishing or playing games
artificial intelligence (AI)	the study of how to make computers do intelligent things that people can do, such as think and make decisions
ascending	describing an arrangement of things (e.g. words, numbers) where each thing is higher, or greater in amount, than the one before it; for example, numbers listed in ascending order start at 1
ASCII	stands for American Standard Code for Information Interchange. It is a system used for exchanging information between computers by allowing them to recognize letters, numbers etc in the same way
attract	to make someone interested in something, or make them want to take part in something or buy something
augmented reality (AR)	a situation in which computer-generated information, images, etc. are combined with things in the real world or images of real things
AVERAGE	a spreadsheet function that calculates and displays the average value of all the values in the range of cells

axis	either of the two lines of a graph, by which the positions of points are measured
back-up	a copy of a computer document, program, etc. that is made in case the original becomes lost or damaged
bandwidth	the amount of information that can be carried through a computer connection at one time
bar chart	a picture of boxes of different heights, in which each box represents a different amount or quantity
base 10	a number system that uses the digits 0 to 9
base 2	a number system that uses the digits 0 and 1
bias	an opinion about whether a person, group or idea is good or bad that influences how you deal with it
binary	a system that uses only the numbers 0 and 1 to represent text or any other form of data
binary word	a collection of 1s and 0s that represent a digital object; for example, a simple bitmap image could be represented by the binary word 1001010000101001
bit	the smallest unit of information that a computer uses
bitmap	a computer image that is stored or printed as an arrangement of bits
blogging	the act of writing and publishing a blog (a web page containing information or opinions from a particular person or about a particular subject)
Bluetooth	a system that allows you to connect computer equipment, such as a keyboard or printer, to a computer that is near it by using radio waves instead of wires. You can also use the system to connect a mobile phone to a computer without using wires
border	the lines around a spreadsheet or table cell or a group of such cells
brand	a type of product made by a particular company, that has a particular name or design
bug	an error in an algorithm or program
byte	a unit for measuring computer information, equal to eight bits
cell	a single area within a table or spreadsheet that contains text or data
cell referencing	using values to describe the location of an individual cell or a range of cells in a spreadsheet
central processing unit (CPU)	the part of a computer that controls what it does
chart	a visual representation of information in a graph or simple picture
chatbot	a computer program designed to simulate a two-way conversation between a real person and itself

clear	expressed in a simple and direct way so that people understand
closed-circuit television (CCTV)	a system of cameras placed in public buildings or in the street, used to help prevent crime
code	a set of instructions that tell a computer what to do
coding	the activity of writing computer programs
colour depth	the number of colours an image can contain, represented by the number of bits per pixel
column addition	adding a column of numbers, working from the right and carrying the extra value to the left
columns	lines of numbers or words written under each other that go down a page
command line interface (CLI)	an interface in which a user enters commands using lines of text
commands	instructions to a computer to do something
complementary colours	colours that go together well
compress	to reduce the data size of a file or folder of files into a compressed file format
compression	the act of making a computer file smaller
computer program	a series of instructions that can be put into a computer in order to make it perform an operation
computer-aided design	the process of drawing something to show how you will make it or what it will look like, using computer software
concatenate	to join two or more strings together to create a single string that can then be processed
connection	when two or more things are joined together or when something is joined to a larger system or network
coordinates	a set of numbers that give the exact position of a point on a map, computer screen, etc.
COUNT	a spreadsheet function that counts the number of values in a range of cells
COUNTIF	a spreadsheet function that counts the number of values in a range of cells that meet specific criteria
criteria	standards that you use to judge something or make a decision about something
cyan	a dark greenish-blue colour
cyberattack	the use of the internet to illegally see or damage the information on someone's computer system

cyberbullying	the activity of threatening or insulting someone through email, text messages, social media posts, chat messages etc
data	values in a form that can be stored and used, especially on a computer; on its own data has no meaning - compare with 'information'
data breach	the access of stored data without the permission of those holding it
data packet	a very small piece of a whole data file; a piece of data that contains binary code
data storage	the way that information is kept on a computer
debugging	looking for mistakes in an algorithm or program and fixing them
decompress	to change the information in a computer document back into a form that can be easily read or used, from information that was stored in a form that used less space on the computer's memory
defragmentation	a method of rearranging files on a computer in a logical way to speed up the computer and restore wasted space
denary	a system of numbers used in day-to-day life that uses the digits 0–9; also called decimal
descending	in an order where each thing is lower, or less in amount, than the one before it
design	the art or process of making a drawing of something to show how you will make it or what it will look like
desktop publishing	the work of arranging the writing and pictures for a magazine, small book, etc, using a computer and special software
devices	things made to do a specific job; often electronic objects (e.g. smartphones, tablet computers)
dialogue	speech
dimension	the length, height, width, depth or diameter of something
disk organisation	the process of ensuring that files are organised, disks are correctly formatted and file space isn't wasted
domain name system (DNS)	the system used to link alphabetical website names to numeric internet protocol (IP) addresses
DPI	dots per inch
eavesdrop	to deliberately listen secretly to other people's conversations
efficiency	the quality of doing something well and effectively, without wasting time, money, or energy
electronic waste (e-waste)	the devices that are no longer used and, instead of being recycled, have just been thrown away in landfill sites
encoding	the process of converting original data into a particular form, for example, an MP3 file

equals sign	the sign (=) that you use in mathematics to show that two things are the same size, number or amount
error	a mistake
ethical	relating to principles of what is right and wrong
facial recognition	a way of knowing who someone is using a digital image of their face
file size	the amount of data a computer file contains
file transfer	the act of uploading and downloading files to a website or network drive
filter	a tool in spreadsheet software that allows data to be displayed only if certain criteria are met
firewall	a system that protects a computer network from being used or looked at by people who do not have permission to do so
formatting	the way in which information is organised, arranged or presented
formula	an arithmetic calculation in a spreadsheet
function	1. a tool for carrying out calculations in a spreadsheet, for example, SUM (to find the total of a range of numbers) and AVERAGE (to find the average of a range of numbers) 2. a specific tool in an electronic device, for example, the camera in a smartphone
gaming	playing or watching online games with international players
geolocation information	data that identifies exactly where you are in the world
geometric shapes	shapes drawn in two or three dimensions (2D and 3D) on paper or on screen (a triangle, a cube, etc.)
gesture/touch	a screen that responds to gestures and touch (e.g. swipes, finger pinches, taps) to perform functions, for example, a smartphone screen
gibibyte (GiB)	a unit for measuring computer information, equal to 1024 mebibytes
gigabyte (GB)	a unit for measuring computer information, equal to 1000 megabytes
goal seek	a function in spreadsheet software that can adjust a variable until a desired outcome is created
gradient	the change from one colour to another, for example, fading from black to white with the pixels moving from black to grey to white
graph	a drawing that uses a line or lines to show how two or more sets of measurements are related to each other
graphical user interface (GUI)	an interface that allows the user to move a cursor around the screen and interact with icons and menus

graphics plotter	a device that follows a series of instructions to create a graphic, for example architectural drawings. The pen is lowered onto the paper as required and lifted when it needs to start another shape
graphics software	applications that allow users to design images, edit photographs, etc.
hack	to secretly find a way of getting information from someone else's computer or changing information on it
hard drive	the part of a computer where information and programs are stored, consisting of hard disks and the electronic equipment that reads what is stored on them
hardware	computer machinery and equipment, as opposed to the programs that make computers work
hazardous	dangerous, especially to people's health or safety
hexadecimal (HEX)	hexadecimal numbers are based on the number 16 and are mainly used on computers
hibernate	to sleep for a long period of time (for example, some animals hibernate all winter); an electronic device that 'hibernates' saves power when not in use
home network	a set of computers in a single home that are connected to each other so that they can share information
hosting	running a website on a web server
HyperText mark-up language (HTML)	a computer language used for producing pages of writing and pictures that can be put on the internet
ID cards	a card that proves who you are; short for identity card
image	a picture (e.g. photograph, drawing)
image editing	the work of changing the appearance of an image, for example, cropping a photograph or applying a filter to alter its appearance
image quality	how good or bad an image is
index	a way to keep track of an item's position in a string
infographic	a way of presenting data, key terms or timelines in a visually appealing way
information	data that has meaning, for example, 3 out of 10 computers use the Apple operating system
input	information that is put into a computer
instructions	statements telling someone or something what they must do, for example, the commands in a computer program
integer	a whole number
interactivity	the ability of a computer program to respond to you directly and to do things in reaction to your actions

intercept	to stop something or someone that is going from one place to another before they get there
internet	the network of computer systems that allows computer users around the world to exchange information
internet protocol address (IP)	a special number that is used to identify a computer, and which the computer needs in order to be able to connect to the internet
internet relay chat (IRC)	introduced in 1988, it allowed people to send instant text messages to each other around the world. It is still in existence today and many current social networks developed from this system
internet service provider (ISP)	a company that provides the technical services that allow people to use the internet, including the internet connection to your home, school or place of work
iteration	repeating an instruction until a specified result is reached
kibibyte (KiB)	a unit for measuring computer information, equal to 1024 bytes
kilobyte (KB)	a unit for measuring computer information, equal to 1000 bytes
knowledge	the information, skills and understanding that you have gained through learning or experience
labels	words or phrases that are used to describe a picture
landfill	a place where waste is buried under the ground
layout	the arrangement of the text, graphics and photos on a page or slide
leading question	a question that makes someone give you the answer that you want
legal	if something is legal, you are allowed to do it or have to do it by law
line graph	a graph on which the relationship between two things is shown by a line
list	a group of objects in Python, separated by a comma and held within square brackets
logic error	a fault with the design of a computer program (as opposed to an error just in a line of code)
logo	a small design that is the official sign of a company or organisation
loop	a set of operations in a computer program that are continuously repeated
lossless	a form of file compression that makes a file smaller without losing any data
lossy	a form of file compression that makes a file smaller but loses some of the original data, so for example, an image will be less detailed than the original
magenta	a dark reddish purple colour

maintenance	the work of keeping a computer, a program, a network, etc. running well, for example, by keeping an eye on system updates, running back-ups or removing unwanted files and applications
marketing	the activity of deciding how to advertise a product, what price to charge for it, etc.
MAX	a spreadsheet function that displays the highest value in a range of cells
mebibyte (MiB)	a unit for measuring computer information, equal to 1024 kibibytes
media sharing	the act of sharing original or internet-sourced photos, videos, music or animation with others
megabyte (MB)	a unit for measuring computer information, equal to 1000 kilobytes
memory	the amount of space that can be used for storing information on a computer
memory management	the work of allocating memory space to programs and data
merging	joining together two or more cells in a table or spreadsheet to create a single cell
messaging	the process of sending messages using electronic equipment
MIN	a spreadsheet software that displays the lowest value in a range of cells
models	computer representations of something; for example, a spreadsheet can be used to model the potential cost of decorating a home by pricing materials and equipment
modem	a piece of electronic equipment that allows information from one computer to be sent along telephone wires to another computer
moral	relating to the principles of what is right and wrong behaviour
motherboard	the main circuit board that all other components connect to
multi-tasking	a computer's ability to do more than one job at a time
network	a set of computers that are connected to each other so that they can share information
network interface card (NIC)	a card that allows a computer or laptop to connect to a wired or wireless network
networking	connecting with others with similar interests anywhere in the world
NFC (near-field communication)	a short range wireless technology with a range of only a few centimetres; it allows data to be communicated between two devices placed next to each other
nibble	four binary digits or half of an 8-bit byte
numerically	relating to numbers; for example, when numbers are listed in order of size, they are said to be ordered numerically

office applications	computer software typically used in an office, for example, word processing, spreadsheet and database software
open source	software designed to be free to use, edit and distribute
operating system	a system in a computer that helps all the programs in it to work together
output	the information produced by a computer
overflow	the result when an 8-bit binary addition is carried out that exceeds the maximum value that an 8-bit binary sequence can represent (256); this means that the sum cannot be carried out
pattern	the regular way in which something happens, develops or is done
peripherals	peripheral equipment can be connected to a computer and used with it - a mouse, a keyboard, etc.
persuade	to make someone decide to do something, especially by giving them reasons why they should do it
phishing	the criminal activity of sending emails or having a website that is intended to trick someone into giving away information such as their bank account number or their computer password; this information is then used to get money or goods
pictogram	a picture that represents a word or phrase
pie chart	a circle divided into parts by lines coming from the centre to show how big the different parts of a total amount are
pixel	the smallest unit of an image on a computer screen
pixel size	the measure of how many pixels in total make up a digital image
pixels per inch (PPI)	a measure of image resolution and size
place value table	a table that identifies the value of a particular digit in a number based on its position
platform	the type of computer system or software that someone uses
portable storage device	a device such as a USB stick or memory card that stores computer folders, files, etc. and can be used to transfer files between different devices
power of	a term that describes multiplying a number by itself a defined number of times; for example, 2^2 (2 'to the power of' 2) means to multiply 2 by 2
power supply unit (PSU)	a PSU provides the correct power supply to all components
primary data	information you have collected yourself, for example, as part of a survey
procedure	a subprogram that carries out a specific task before the main program continues

profit	money that you gain by selling things or doing business, after your costs have been paid
program	a set of instructions given to a computer to make it perform an operation
pseudorandom	a computer-generated output (e.g. a pseudorandom number) that appears to be random but is not truly random
Python	an open source programming language
qualitative	relating to the quality or standard of something rather than the quantity
qualitative data	data that is based the quality or standard of something rather than the quantity
quantitative	relating to amounts rather than to the quality or standard of something
quantitative data	data that is based amounts rather than the quality or standard of something
question	a sentence or phrase that is used to ask for information or to test someone's knowledge
questionnaire	a written set of questions which you give to a large number of people in order to collect information
RAM (random access memory)	the part of a computer that acts as a temporary store for information so that it can be used immediately
random	1. happening or chosen without any definite plan, aim or pattern 2. a function within Python that can also be incorporated to allow the random selection of an object from a list
rechargeable	a rechargeable battery is one that is able to receive a new supply of electricity
recycled	put through a special process so that they can be used again
reliability	the quality of being able to be trusted or believed because of working or behaving well
represent	to be a symbol of something
research	to study a subject to discover new facts or test new ideas
resolution	a measurement of the number of pixels in a digital image; the higher the resolution, the clearer the image
response	something that is done as a reaction to something that has happened or been said
return	(of a computer program) to perform a response to a user's command
router	a piece of electronic equipment that makes sending messages between different computers or between different networks easier and faster

rows	lines of numbers or words written next to each other that go across a page
run length encoding (RLE)	a lossless compression method; it works by analysing and reducing any repeated characters in a string by grouping them together
sample size	the number of people who take part in a survey questionnaire
scale down	to decrease the size of an image while keeping its dimensions in proportion
scale up	to increase the size of an image while keeping its dimensions in proportion
scaling	increasing or decreasing the size of an image while keeping its dimensions in proportion
scatter graph	a graph that displays two sets of data to see if there is a correlation, or connection, between them
Scratch	a visual coding application that uses colourful blocks
secondary data	information that has already been collected by someone else that you are using
security	monitoring and controlling access to programs and data and preventing any unauthorised access to the system
selection	the process of adding a question to an algorithm and taking action based on the result
sentence case	is a style of capitalisation where only the first letter of the sentence and proper nouns are capitalised eg The car belongs to Marta
sensors	pieces of equipment used for discovering the presence of light, heat, movement, etc.
sequence	a step-by-step series of instructions that follow each other in a logical order
sequencing	a logical series of instructions, one followed by another from start to finish
social marketing	advertising on free social media sites; advertisers will target users with specific advertisements based on their profile and interests
social media	ways of sharing information, opinions, images, videos etc. using the internet, especially social networking sites
social media platforms	websites that enable the sharing information, opinions, images, videos, etc.
software	the sets of programs that tell a computer how to do a particular job
sort	a function that allows data to be sorted in a particular order
sound card	a circuit board that can be added to a computer so that it is able to produce sound

sound editing	the preparation of a recorded sound or collection of sounds (speech, sound effects, music, etc.) prior to an audio file being finalised
spectrum	the set of bands of coloured light into which a beam of light separates when it is passed through a prism
spreadsheet	a computer program that can show and calculate financial information
sprite	an on-screen character
spyware	computer software that secretly records information about which websites you visit
storage	the way that information is kept on a computer
string	any combination of characters (letters, numbers and symbols) in a computer program or entered as input
subprogram	a block of code that can be reused either within the same program or in several different programs
subroutine	a part of a computer program containing a set of instructions that will be followed every time the main program calls for it
SUM	a spreadsheet function that displays the total of a range of selected cells
surveillance	when the government, police, etc. watch a person or place carefully because they may be connected with criminal activities
survey	a set of questions that you ask a large number of people in order to find out about their opinions or behaviour
survey data	the responses to a set of questions that you ask a large number of people in order to find out about their opinions or behaviour
switch	a piece of equipment that connects directly to a router and directs data to and from specific devices
syntax error	a mistake within a program that causes the program to stop running
target audience	the group of people for whom a digital or physical product is designed
test table	a table used to check the functionality of a program
timeline	a line showing the order in which events happened
toxic	containing poison, or caused by poisonous substances
transfer	uploading and downloading files using the internet
transferability	how easily something can be used in different places, for example, a logo in different type of files
trends	general tendencies in the ways situations are changing or developing
Turing Test	a test created by Alan Turing to challenge the ability of a computer to simulate a real person in questioning

turtle	a Python module that allows simple commands to be sent to a virtual pen to create graphics
type	one member of a group of people or things that have similar features or qualities
uniform resource locator (URL)	a website address
unpatched software	software that has not been updated with the latest security updates
unreliable	unable to be trusted or depended on
user	someone who uses a computer, computer program, online service, etc.
user interface	the on-screen display that allows the user to interact with the computer, move the mouse and see applications
user-friendly	easy to use, understand or operate
user-generated	created by the user
utility software	software that supports the operating system and performs specific tasks, for example, checking for and removing viruses or making sure the hard disk is running efficiently
value	the content of an individual cell in a spreadsheet
variables	values that may be different in different situations, so that you cannot be sure what will happen(e.g. bandwidth, file size)
vector	an image that is made up of lines, curves, etc. joined by points; unlike other image formats such as JPEGs or GIFs, vector images contain no pixels
video card	a circuit board that can be added to a computer so that it is able to produce sound
virtual reality (VR)	an environment produced by a computer that looks and seems real to the person experiencing it
wearable technology	any computer device that is connected to the internet worn by someone (e.g. a smart watch, smart jewellery, smart glasses)
web browser	a computer program that finds information on the internet and shows it on your computer screen
web server	a network-connected computer server that stores website files that can be accessible by users of the internet
what if scenarios	questions posed to see what might happen when you do something before you actually do it
Wi-Fi	a way of connecting computers or other electronic machines to a network or the internet by using radio signals rather than wires
wireless	a system of communications that does not use electrical or telephone wires

wireless access point (WAP)	a piece of equipment that provides a connection for wireless devices to the local network
worksheet	an individual collection of cells arranged in rows and columns within a spreadsheet; a spreadsheet can contain multiple worksheets
world wide web (WWW)	a system that allows computer users to easily find information that is available on the internet, by providing links from one document to other documents, and to files with sound, pictures, etc.

INSPIRE COMPUTING

International

YEAR 8

Student Book

Inspire Computing International helps learners of all ages to become confident in core computing skills, both in the classroom and at home!

Student books present and embed key computing knowledge and skills in an engaging but supportive and structured progression.

Workbooks are linked closely to the learning and provide the opportunity to apply, consolidate, and extend skills.

Create students who are fully prepared for the future - with Inspire Computing International!

Key features include:

- Complete coverage of the Pearson Computing Curriculum and the English National Curriculum in Computing
- Clear and structured progression towards GCSE and International GCSEs in Computer Science and ICT
- Flexible and inspiring topic based learning - everything from understanding algorithms to creating your own animations!
- Key skills and knowledge embedded through a variety of engaging activities
- A combination of supportive teacher-led assessments and self-assessments help learners understand their progression.



Other Student book titles in the series:

Year 1	978-1-292-40421-9
Year 2	978-1-292-40422-6
Year 3	978-1-292-40423-3
Year 4	978-1-292-40424-0
Year 5	978-1-292-40425-7
Year 6	978-1-292-40426-4
Year 7	978-1-292-40427-1
Year 9	978-1-292-40429-5

www.pearson.com/international-schools