Summer

Computer Science

ICT - Year 8

Topic: Python & Data Representation

Name: _____

ICT, Business and Computer Science Curriculum Pathway

Digital Literacy Computer Science ICT Creativity

Year 7

- Introduction to Computing and Computer Hardware
- Programming in Scratch and Flowol
- E-safety and Website Development

Year 8

- Digital Tools for Business
- Programming in Python and Data Representation (YOU ARE HERE)
- HTML and Comic Creation

Year 9

- Analysing, Planning and Creating Digital Graphics
- History of Computing, Algorithms and Logic

KS4 Options

GCSE Computer Science

- Systems Architecture
- Memory and Storage
- Networks and Protocols
- Network Security
- Systems Software
- Ethical, Legal & Environmental
- Algorithms & Logic

OCR Creative iMedia

- R081 Pre-Production Skills
- R082 Digital Graphics
- R084 Storytelling with Comics
- R085 Creating a website

A Level Computer Science

- Theory of Computer Science
- Programming and Algorithms
- Computing Practical Project

BTEC Level 3 ICT

- Information Technology Systems
- Using Social Media in Business
- Website Development

BTEC Level 3 Business

- Exploring Business
- Developing a Marketing Campaign
- Personal and Business Finance

KNOWLEDGE ORGANISER
BIG IDEA: Computer Science
TOPIC: Python & Data Representation

Key Word	Definition
Algorithm	A set of rules/instructions to be followed by a computer system
Sequence	Parts of the code that run in order and the pathway of the program reads and runs very line in order
Selection	Selects a pathways through the code based on whether a condition is true
Iteration	Code is repeated (looped), either while something is true or for a number of times
Variable	A value that will change whilst the program is executed. (eg. temperature, speed)
Data type	This indicates how the data will be stored. The most common data types are integer, string, and float/real
Syntax	The punctuation/way that code has to be written so that the computer can understand it. Each programming language has its own syntax.
Binary	The language of computers, made up of 0s and 1s
Pixel	The smallest part of an image
Sound sampling	The amplitude of the wave is measured at regular intervals which creates a digital representation of the wave.

Comparative Operators	
==	Equal to
!=	Not equal to
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to

Denary is the decimal number system that we are used to. It uses the numbers 0-9 and the column headings go up in powers of 10.

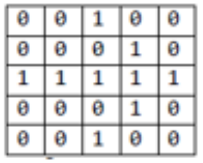
100 (Hundreds)	10 (Tens)	1 (Units)
2	3	8
2 lots of 100	3 lots of 10	8 lots of 1

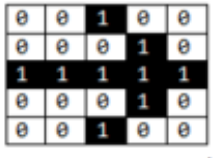
Binary uses the numbers 0 and 2. The column headings go up in power of 2:

128	64	32	16	8	4	2	1
0	1	0	0	0	1	1	1

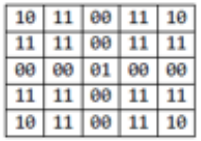
64 + 4 + 2 + 1 = 71

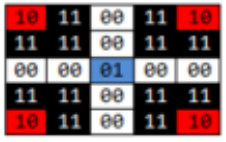
Images are made up of pixels
The colour of each pixel is represented by a binary number. If an image uses 1 bit to represent each colour then it will only have 2 colours:





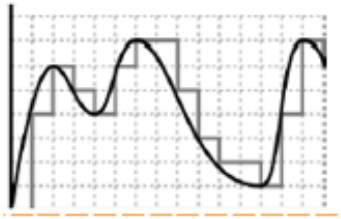
This is a 1-bit image so it uses 2 colours.
0=white and 1=black





This is a 2-bit images so it uses 4 colours.
00=white,
01=blue, 10=red,
11=black

The analogue wave is smoother and shows continuous data. The digital sampling shows the amplitude of the wave at different points.



Data Type	Definition
String	Text eg: "Hello"
Integer	Whole number eg: 32
Float/Real	Decimal number eg: 1.2
Boolean	Two values eg: true or false
Character	A single character eg: b

```
name = input("Enter Name")#A
age = 14                  #B
users = ["John","Jane"]   #C
length = len(users)       #D
valid = False             #E
                          #F
for i in range(length):   #G
    if name == users[i]   #H
        valid = True      #I

if valid == True:         #J
    print("Valid user")
else:                     #K
    print("Invalid user")
```

A: Stores user inputted text to a variable called name.
B: Stores an integer value 14 to a variable called age.
C: Creates a list which contains 2 string values. Stores to users.
D: Calculates the length (how many items are in) of users.
E: Stores the Boolean value False to variable valid.
F: Use of white space to make code clearer.
G: Creates a loop that will iterate for every element in users.
H: Use of selection to determine if the entered value match with any values from the list

This binary addition gives an overflow error as the total does not fit in 8 bits (a byte).

```
  1 0 0 1 0 1 0 1
+ 1 1 0 1 1 0 1 1
1 1 1 1 1 0 0 0 0
1     1 1 1 1 1
```

Lesson 1: Introduction to Python

Objective: Be familiar with the Python environment and write a simple program

DO NOW:
This PC→StudentRO (K:\ Drive) → Subjects → ICT → 1 Year Groups -> 2 Year 8 ICT -> Python
Right click → Copy and paste the file called “template” into your Y8 ICT, Computer Science folder

Task 1: What is an algorithm?

_____

_____

_____

Task 2: Hello World
Right click on ‘template’ → Edit with IDLE
In your template type the words below, exactly as they appear (remember to use lower case). Press **F5** to run the code:

```
print(“Hello World!”)
```

Try writing a few different statements of your own to the screen using this command.

Task 3: Syntax rules

Code	Error? (Y/N)	What caused the error?
print(Hello, world!)		
print("Hello, world!)		
print "Hello, world!"		
print("" Hello, world!)		
print ("Hello, world!")		
Print("Hello, world!")		
_print("Hello, world!")		

What have you learnt from the activity above? What will create syntax errors in your code? Identify some 'rules' here:

Task 4:

1. Write a program that will output a message of your choice to the screen.
2. Write a program that will output a picture to the screen. E.g.:

```
print(5 + 5)
print(5 * 5)
print(12 / 3)
print(8 - 4)
print("Hello" * 5)
```

```
      .--.
   .---' '--.
  '-()-------()-'
```

3. Write a program that will display your email signature. Your signature should include your name, position and a picture that represents who you are.
4. Write a program that prints out your name 10,000,000 times.
5. Make a variable called dogs and assign the number 3 to it
6. Make a variable called cats and assign the number 5 to it
7. Make a variable called total_pets and make it equal to cats + dogs
8. Make a variable called total_pets_in_town and make it equal to total_pets * 100
9. Print total_pets_in_town

Lesson 2: Variables and Inputs

Objective: Learn how to store inputs into variables in Python

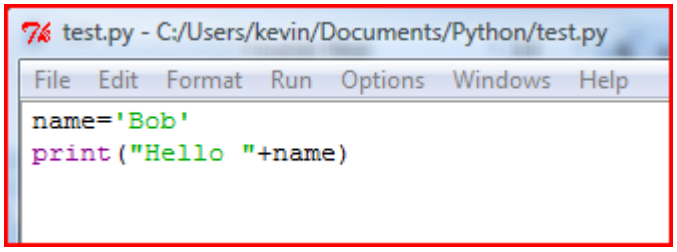
DO NOW: Will it work? Yes or no?

a) Print("Hello World!")		b) print("Hello Year 8")	
c) print("I like Python")		d) print("I like Python)	
e) print("Computers")		f) print(I like Python)	

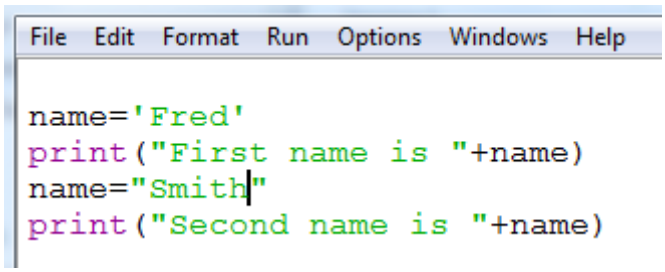
Task 1: Variables

In programming, inputs and data are stored in something called a ***variable***
It can simply be thought of as a storage box.

Try this…..



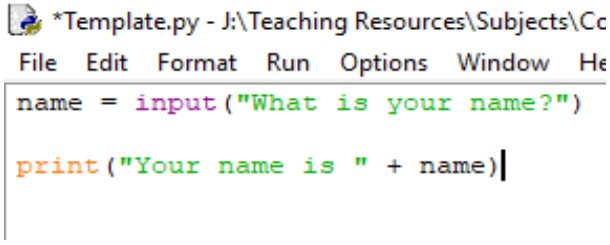
Try running the code above and explain what happened:



We can change the value of a variable within the program. Explain what happened…

Task 2: Inputs

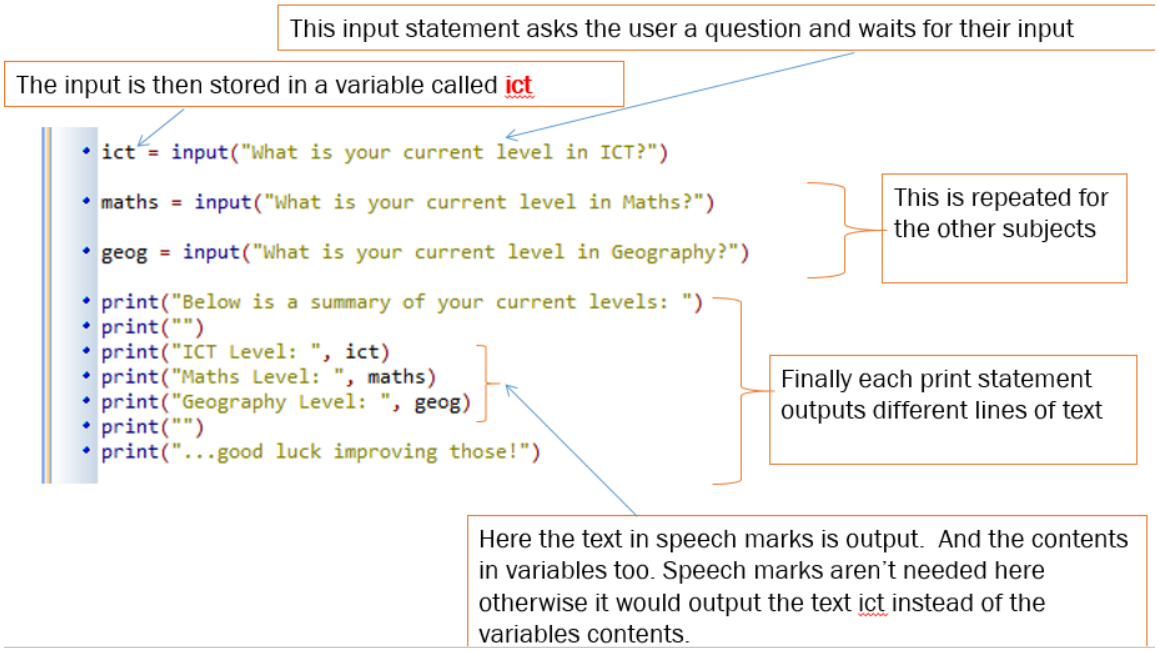
Inputs allow us to get information from the user and store it in a variable. This is shown in the example below



Write this code and add **5+** more questions afterwards and display the answers in a sentence.

Make sure you choose sensible variable names!

Task 3: Individual task



Write a program, similar to the one above, which does the following:

- Asks for your current level in all of your subjects
- Stores the inputs in different, suitably named variables
- Displays your current levels in an organised, well presented report on the screen

Lesson 3: Data Types

Objective: Understand the different data types and their uses

DO NOW: Python and Numbers

Open the Python template file (Right click, edit with IDLE save as -> data types)

Try each of the commands Write what the output is for each command and think about why that output has happened.

print("OK Joe!")	
print(5 + 11)	
print("5 + 11")	
print(“Joe" + “Bloggs")	
print(“Joe", “Bloggs")	
print("5" + "11")	
print("Python is Ace" * 5)	

Task 1: Data types

For each of the four data types give a descriptions and 3 examples.

String		
Description:		
Examples:		

Boolean		
Description:		
Examples:		

Integer		
Description:		
Examples:		

Real		
Description:		
Examples:		

Task 2: Programming activities

You now have a number of different challenges to complete.

The first one has been completed below to help you – try this code first

1) Ask the user to input two numbers and add those numbers together

For this program you need:

Two variables – each with an input that is converted to an integer

```
firstNumber = int(input("Please enter the first number"))
secondNumber = int(input("Please enter the second number"))
```

A calculation to work out the answer and store it in another variable called answer

```
answer = firstNumber + secondNumber
```

Finally a print statement to show the result.

```
print(answer)
```

2) Write a program that will store two inputs from the user and subtract one from the other, then display the result

3) Write a program that will input three numbers and multiply them together, then display the result.

4) Write a program which will store two inputs and divide them, then display the result.

5) Weights can be converted from pounds to kilograms using the following formula

(note that “kilo” is simply a variable):

kilo = pound * 0.45

Write a program to input a weight in pounds and output the equivalent weight in kilograms.

6) Enter the length, width and depth of a rectangular swimming pool.

Write a program to calculate the volume of water required to fill the pool and output the volume.

Extension Task: Write a program to work out your pocket money spending by carrying out the following steps:

- Input a variable pocketmoney to the amount of money you get each month.
- Output the title ‘Pocket Money’
- Output the number of pounds at the start of the month
- Input the number of pounds spent on food
- Input the number of pounds spent on your mobile phone
- Output the amount of pocket money left

Plenary: Data Types quiz
Go to this website and complete the quiz
tinyurl.com/L3quizdata

Lesson 4: Conditions

Objective: Understand how to write conditional statements in Python

Do Now: Complete the data types match-up on your PC

Task 1: Boolean logic

Give the meanings of the Boolean logic symbols

Boolean Logic	
Symbol	**Meaning**
<	
>	
==	
!=	
<=	
>=	

Task 2: Conditionals

Create your own age checking program using IF Statements and the Boolean logic symbols in the table above.

```
age = input("What is your age?")

age = int(age)

if age > 17:
    print("What can I get you young sir?")
else:
    print("Go home boy!")
```

Examples:

- Can you learn to drive? (17+)
- Seeing an age rated film (12, 15, 18)
- Achieving enough marks to pass a test (out of 60 marks)

Use the template for help

Extension: Try coding a different IF Statement program that asks for two inputs from the user and displays different messages depending on the answers.
E.g. Are they over 17? **and** do they have a car?
You could also try coding a program to combine with the work you completed in previous lessons – e.g ask the user which shape they would like to be drawn – if the user enters "square" then draw a square to the screen using the Turtle

Plenary: Correct the mistakes in the code

```
age = imput("Enter your age")

if age < 17:
    print("You can drive
else
    printYou can't drive
```

Lesson 5: Conditions 2

Objective: Understand how to write conditional statements in Python

Do Now: Decisions recap
Find the ICT → Python → template file on your drive
Right click → edit with IDLE
Write a program that will allow the user to input a value and output whether the value is zero or above *(number >= 0)* or below zero *(number < 0)*.
REPLACE

Task 1: Password checker

```
answer = input("Do you like school? (y/n) ")

if answer == "y":
    print("You answered YES!!")
else:
    print("You answered NO!!")
```

- Write a program and immediately create a variable which contains a password:
  E.G. ***set_password = "5idm0uth"***
- Then ask the user to try and guess the password and store this in another variable:
  E.G. ***user_guess = input("Guess the password!")***
- The program should display a suitable error message if the password is incorrect and a welcome message if the password is correct.

Task 2: 10 Question Quiz
Using Python, create a 10 question general knowledge quiz.
You need to use elif to give multiple outputs for different answers.
Use the template for each of your questions.
Note: You need to use a new, different variable for each question!

```
city = input("What is the capital city of France?")

if city == "Paris":
    print("Correct, Well Done")
elif city == "Lyon":
    print("Right country, wrong city")
elif city == "London":
    print("Wrong country, wrong city")
else:
    print("Sorry, wrong answer")
```

Lesson 6: While Loops

Objective: Understand how to repeat lines of code using while loops

Do Now: Dingbats

Guess the Python keywords from the pictures.

Task 1: Drawing Shapes

Copy the code below into a Python file. Then finish the code to draw a square.

```
from turtle import*

shape("turtle")

forward(100)
right(90)
forward(100)
right(90)
```

Task 2: Drawing Shapes using loops

The code below draws a square using a loop. Try the code below and then adapt your code to solve the challenges.

```
from turtle import*

i = 0
while i < 4:
    forward(100)
    right(90)
    i = i+1
```

Challenge 1: Modify the code to draw:

1. A triangle
2. A hexagon

Challenge 2:

1. Modify your code from the previous challenge to draw a star. Hint: 144°!

2. Draw three stars in different locations around the screen.
   Hint: You will need to:
   a. Draw a star
   b. Pick the pen up
   c. Move the pen
   d. Put the pen down
   e. Draw a star
3. Finally:
   a. Change the background colour to black
   b. Change the pen colour to yellow

Extension:

Open the file "6. While Loops - Extension Challenges". Open each of the links and try to complete the challenges. Use this code to help you.

Plenary: Spot the mistakes

Circle all the mistakes in the code below

```
from turtle import
shape("turtle)
forward100)
ri ht(90)
Forward(100)
```

Lesson 7: Python Assessment

Objective: Complete a mini project including information about all the things you have learnt about Python this half term

Task: Python Poster

You are going to create a poster demonstrating all of the things you have learnt this half term

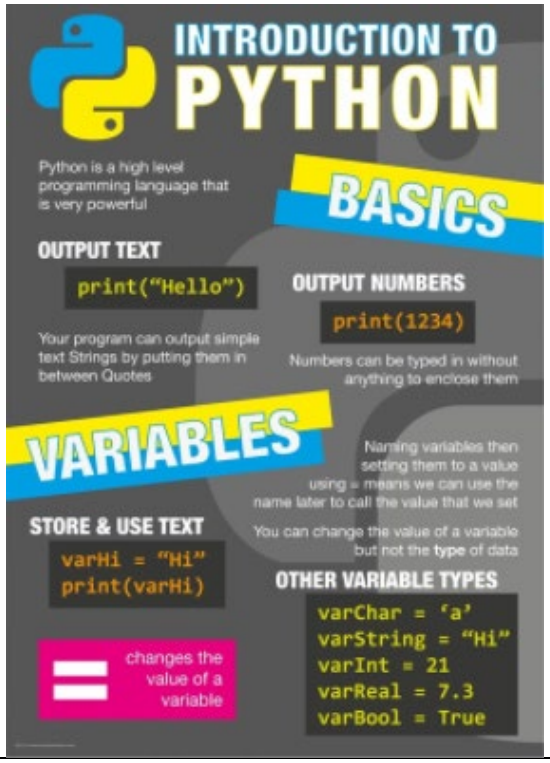
You need to use your existing Python code to help you create a poster called "The How-To Guide of Python" informing people how to create basic programs.

Your poster should include information about:

- Inputs and outputs
- Python turtle
- Data types
- If statements
- While loops
- Syntax errors

You should explain how to write commands and give examples (screenprints) of your own code for each bullet point.

E.g



When you have finished your poster you should print it and stick it in your booklet in the box below.

<Stick your poster in here>

Mini Project PRIDE

Effort 1 2 3 4

Strength

- ❑ You have explained how inputs and outputs work
- ❑ You have demonstrated Python turtle commands
- ❑ You know the main data types – string, int, Boolean and real
- ❑ You can write if statement commands
- ❑ You can use while loops to repeat sections of code
- ❑ You can identify errors in code
- ❑ You have given thorough explanations of your code

Target

- ❑ Write the code for an input command
- ❑ Write the Python turtle code to draw a triangle
- ❑ Add 2 data types to your poster and give examples of the data
- ❑ Write an if statement that checks if someone is older than 17, and if so ouputs “Can drive”
- ❑ Write a while loop to print out the numbers 1 to 10
- ❑ Give the definition of a syntax error
- ❑ Add some explanations to the different code discussing what they do

Presentation and SPaG

- ❑ Correct any missing capital letters
- ❑ Correct your spellings
- ❑ Add in missing full stops/ apostrophes
- ❑ Write in black pen
- ❑ Draw diagrams in pencil

Student Improvement:

_____

_____

_____

_____

Lesson 8: Introduction to binary

Objective: Be able to convert from binary to denary numbers and from denary to binary

Task 1: Convert each 'Binary' number into Denary

128	64	32	16	8	4	2	1

Convert 10010111 and write the Decimal number here

128	64	32	16	8	4	2	1

Convert 01110110 and write the Decimal number here

128	64	32	16	8	4	2	1

Convert 01111111 and write the Decimal number here

128	64	32	16	8	4	2	1

Convert 01001100 and write the Decimal number here

128	64	32	16	8	4	2	1

Convert 10100010 and write the Decimal number here

128	64	32	16	8	4	2	1

Convert 00110011 and write the Decimal number here

Task 2: Convert each 'Denary' number into Binary

128	64	32	16	8	4	2	1

Convert 92 and write the Binary number here

128	64	32	16	8	4	2	1
1	1	1	1	1	1	1	1

Convert 255 and write the Binary number here

128	64	32	16	8	4	2	1

Convert 196 and write the Binary number here

128	64	32	16	8	4	2	1

Convert 51 and write the Binary number here

128	64	32	16	8	4	2	1

Convert 23 and write the Binary number here

128	64	32	16	8	4	2	1

Task 3: Create a Powerpoint to that explains what binary is and gives a step-by-step tutorial on how to convert numbers from denary to binary.

Plenary: Go to the website below and complete the binary game
tinyurl.com/Y8binarygame

Lesson 9: Binary Addition

Objective: Be able to add binary numbers together and know the rules for binary addition

Do Now: Conversions recap

Convert this binary number to denary: 00100110

128	64	32	16	8	4	2	1

Answer:

Convert this denary number to binary: 187

128	64	32	16	8	4	2	1

Answer:

Task 1: Binary addition practice

Using rules (a)-(e) of binary addition below as a guide, work out the answers to questions 1-7.

(a) **0 + 0 = 0**

(b) **0 + 1 = 1**

(c) **1 + 0 = 1**

(d) **1 + 1 = 0** *carry* ***1*** **= 10**

(e) **1 + 1 + 1 = 1** *carry* **1 = 11**

1. Start with this simple addition. (You can use rules (a) and (b) to help you if necessary.) Calculate the denary equivalent to check that it is correct.

	8	4	2	1			Denary equivalent
	0	0	1	1	=		3
+	0	1	0	0	=	+	
=					=	=	

2. Use the same techniques as you did in the last question to find the binary result.

	0	0	0	1
+	1	1	0	0
=				

3. Now use rule (d) to help with this problem. Use the carry row at the top for the carried 1.

	0	0	1	1
+	1	0	1	0
=				

4. This one will carry into a new column. Remember that like in denary addition, the last carry just makes the number bigger and is added on to the left of the number.

		1	0	1	0
+		1	0	1	1
=					

5. Use rule (e) in this question. Use the carry row again and remember: 1+1+1 = 1 carry 1.

		0	1	1	0
+		1	1	1	0
=					

6. Now try a full 8-bit binary number. Apply the same rules as before.

	1	0	0	0	1	0	1	1
+	0	1	0	0	1	0	1	0
=								

7. Here is another 8-bit number that requires you use all of the rules.

	1	0	0	1	1	1	1	0
+	0	1	0	1	1	0	1	0
=								

Task 2: You need to create a quiz in Powerpoint including:

- ✓ Binary -> Denary conversions
- ✓ Denary -> Binary conversions
- ✓ Binary addition.

You must include a set of answers at the end!

Plenary: Binary Recall Quiz

1. Convert this binary number to denary:

128	64	32	16	8	4	2	1	Denary
1	1	0	1	0	1	1	1	

2. Convert this denary number to binary:

128	64	32	16	8	4	2	1	Denary
								149

3. What is 1+1 in binary?

_____

4. What is the biggest number you can represent in 8 bits?

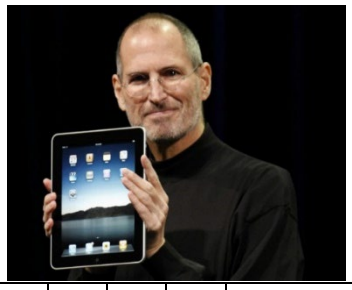
_____

Lesson 10: Images

Objective: Understand how computers store images in binary

Do Now: Conversions recap

- Calculate the year Steve Jobs was born and the year he died
- NO CALCULATORS!



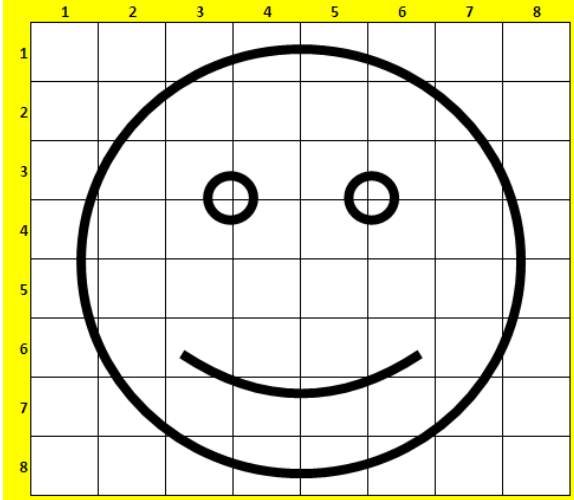
	1024	512	256	128	64	32	16	8	4	2	1	Denary
Born	1	1	1	1	0	1	0	0	0	1	1	
Died	1	1	1	1	1	0	1	1	0	1	1	

Task 1: Practice drawing in pixels

Copy the ***Bitmap Images*** spreadsheet from the school network into your own work area and then open the file.

Draw an 8 x 8 image

Look at the image on the left. You are going to replicate this image on the right. If part of the image is showing in box on the grid on the left then type in a 1 on the same box in the grid on the right. If the box is empty on the left then type in a 0 on the same box on the right. The first row has been done for you.



	1	2	3	4	5	6	7	8
1	0	0	1	1	1	1	0	0
2								
3								
4								
5								
6								
7								
8								

Explain what “pixelated” means:

_____

_____

_____

Task 2: 16 x 16 image

- Move onto the next worksheet and repeat the process with a 16 x 16 grid.
- Have a look at the final 16 bit image on the next worksheet.
- What happened to the quality of the image compared with when we had an 8 x 8 grid?

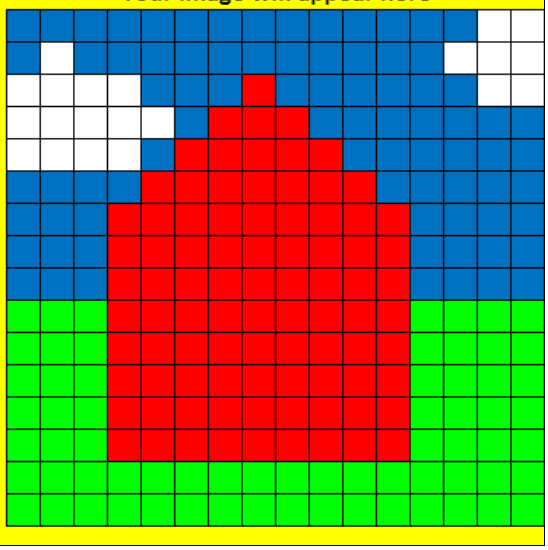
Task 3: Colour images

We are going to use a wider range of numbers to make a colour image. Use the numbers 0 to 3 inclusive. Make up your own image using these colours.

Type in your numbers in this grid

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	3	3	3	3	3	3	3	3	3	3	3	3	3	3	0	0
2	3	0	3	3	3	3	3	3	3	3	3	3	3	0	0	0
3	0	0	0	0	3	3	3	1	3	3	3	3	3	3	0	0
4	0	0	0	0	0	3	1	1	1	3	3	3	3	3	3	3
5	0	0	0	0	3	1	1	1	1	1	3	3	3	3	3	3
6	3	3	3	3	1	1	1	1	1	1	1	3	3	3	3	3
7	3	3	3	1	1	1	1	1	1	1	1	1	3	3	3	3
8	3	3	3	1	1	1	1	1	1	1	1	1	3	3	3	3
9	3	3	3	1	1	1	1	1	1	1	1	1	3	3	3	3
10	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2	2
11	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2	2
12	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2	2
13	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2	2
14	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2	2
15	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
16	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

Your image will appear here



Extension: Create your own bitmap image using the “Bitmap Maker” excel file

Plenary: Go to the link and try out the pixelation widget goo.gl/tCUKyF

Lesson 11: Text & Sound

Objective: Understand that text and sound is represented by binary numbers in a computer system

Do Now: Images in binary: Let's get started!
What image is being drawn here? Shade in the correct pixels

0	**1**	**0**	**0**	**0**	**0**	**1**	**0**
0	**0**	**1**	**0**	**0**	**1**	**0**	**0**
0	**1**	**1**	**1**	**1**	**1**	**1**	**0**
1	**1**	**0**	**0**	**0**	**0**	**1**	**1**
1	**1**	**1**	**1**	**1**	**1**	**1**	**1**
1	**0**	**1**	**0**	**0**	**1**	**0**	**1**
0	**0**	**1**	**1**	**1**	**1**	**0**	**0**
1	**1**	**0**	**0**	**0**	**0**	**1**	**1**

Task 1: Open the sound_task excel file and convert the analogue sound file into binary.
When you are finished click on the "digital" tab to see the result.
Explain what sound sampling is in the box below:

_____

_____

_____

_____

Task 2: Crack the Secret Coded Message

Character	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Denary	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90

128	64	32	16	8	4	2	1

c	01010011	01000101	01000011	01001111	01001110	01000100				

2	01001110	01001001	01001110	01010100	01001000					

3	01000110	01001111	01010101	01010010	01010100	01000101	01000101	01001110	01010100	01001000

4	01000110	01001001	01010010	01010011	01010100					

5	01000101	01001001	01000111	01001000	01010100	01000101	01000101	01001110	01010100	01001000

6	01010100	01010111	01000101	01001110	01010100	01011001	01000110	01001001	01000110	01010100

Extension: Your own secret message
Write your own secret coded message in binary and get your friend to crack it.

Lesson 12: End of Topic Test

Objective: Complete the end of unit test to assess our understanding of the topic.

Assessment Criteria:
Can write commands in Python including input, output and if statements to make programs

Q1) Use the instructions to draw the path for the robot. The robot starts at the square marked by the arrow.

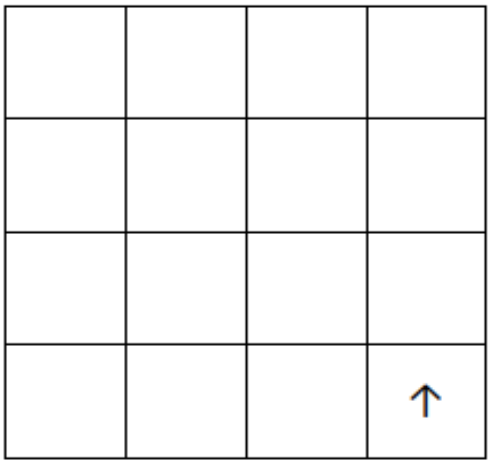
Forward(3)

Turnleft()

Forward(2)

Turnleft()

Forward(1)



/ 3

Q2) What data types would you use for the following?

	Data Type
Name	
Light on?	
Age	
Pi	

/ 4

Q3) A syntax error is where you've broken the rules of the language.

Answer the following questions about syntax errors.

i) Tick the line of code which contains a syntax error:

- ☐ name="Alex"
- ☐ print (name)
- ☐ print ("hello Alex)

```
name="Alex"
print (name)
print("hello Alex)
```

ii) **Tick** the line of code which contains a syntax error:

- ☐ print (“How are you”)
- ☐ name=input(“What is your name”)
- ☐ Print (name)

```
print("How are you")
name=input("What is your name")
Print(name)
```

i) **Tick** the line of code which contains a syntax error:

- ☐ age 20
- ☐ print(“What is your age?”)
- ☐ print (age)

```
age 20
print("What is your age?"
print(age)
```

iv) **Tick** the line of code which contains a syntax error:

- ☐ age = imput(“Enter your age
- ☐ print (age)
- ☐ print(“Thank you”)

```
age = imput("Enter your age")
print(age)
print("Thank you")
```

v) **Tick** the line of code which contains a syntax error:

- ☐ name = “Tom”
- ☐ if name == “Tom”
- ☐ else:

```
name = "Tom"
if name == "Tom"
    print("Hey Tom")
else:
    print("Who are you?")
```

/5

Q4) Evaluate what each of the programs are going to say

i) Tick the output of the code:

- ☐ hobby
- ☐ Football
- ☐ print
- ☐ Error

```
hobby="Football"
print(hobby)
```

ii) **Tick** the output of the code:

- ☐ Alex
- ☐ Sam
- ☐ name
- ☐ Alex & Sam

```
name="Alex"
name="Sam"
print(name)
```

iii) **Tick** the output of the code:

```
name = "Alex"
if name=="Alex":
    print("Welcome")
else:
    print("What is your name?")
```

- ☐ name
- ☐ Alex
- ☐ What is your name?
- ☐ Welcome

iv) Tick the correct answer:

```
game = "Fifa"
if game == "Fifa":
    print("Good game")
else:
    print("Bad game")
```

- ☐ Fifa
- ☐ game
- ☐ Good game
- ☐ Bad game

v) Tick the correct answer:

```
X = 12
Y = 8
answer = X + Y
print(answer)
```

- ☐ 12
- ☐ 8
- ☐ 20
- ☐ answer

vi) Tick the correct answer:

```
X = 11
Y = 2
answer = X - Y
print(answer)
```

- ☐ 11
- ☐ 2
- ☐ 9
- ☐ answer

/6

Q5) Indicate whether the statement is true or false.

Example: 5 > 1 is **True**.

5 == 5	True / False
2 != 3	True / False
9 < 5	True / False
4 >= 8	True / False

/4

Assessment Criteria: Can use commands in Python including loops and functions to make programs	
Q6) Read through the given code. There are 5 mistakes in the code. Circle the mistakes	

``
fro turtle import *
turns = 0
while turns < 6
    orward(100)
    right60)
    turns = turns +
``
	/5

Q7) A In the box below. Write the code to display the numbers 1-10 on screen using a while loop	
	/4

Assessment Criteria: Can convert binary and denary numbers

Q8) Convert each of the following binary numbers into denary

128	64	32	16	8	4	2	1

Convert 00001111 and write the Decimal number here

128	64	32	16	8	4	2	1

Convert 10011111 and write the Decimal number here

128	64	32	16	8	4	2	1

Convert 11010100 and write the Decimal number here

/ 3

Q9) Convert each of the denary numbers into binary

128	64	32	16	8	4	2	1

Convert 22 and write the binary number here

128	64	32	16	8	4	2	1

Convert 57 and write the binary number here

128	64	32	16	8	4	2	1

Convert 247 and write the binary number here

/ 3

Q10) using the binary addition rules below, add the following binary numbers

0 + 0 = 0

0 + 1 = 1

1 + 0 = 1

1 + 1 = 0 *carry* ***1*** **= 10**

1 + 1 + 1 = 1 *carry* **1 = 11**

		1	1	0	0
+		1	0	1	0
=					

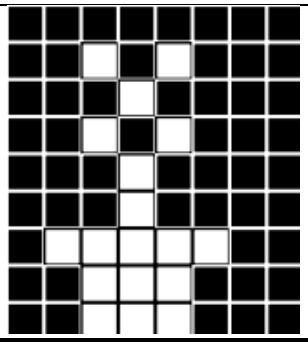
		0	1	1	0
+		1	1	1	1
=					

		1	0	1	1	1	1	1	1
+		0	1	0	0	1	0	1	0
=									

/7

Assessment Criteria: Know how text, images and sound are stored in computers

Q11) Convert this black and white image into it's binary equivalent. The first two rows are done for you

Row 1		1111 1111
Row 2		1101 0111
Row 3		
Row 4		
Row 5		
Row 6		
Row 7		
Row 8		
Row 9		

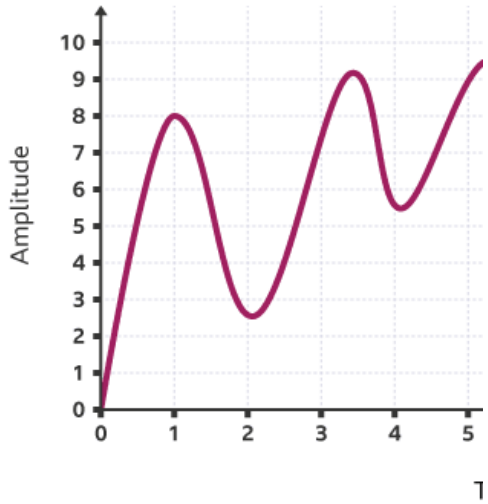
/ 4

Q12) What is the smallest part of an image

_____

/ 1

Q13) A small section of a sound wave is shown in a graph below. The sound recorded at each sample point is converted into binary. The first has been done for you. Convert the rest of the sound file into binary.



Sample	1	2	3	4	5
Denary	8				
Binary	1000				

/4

Q14) Explain what sound sampling is. Explain what happens to the sound file if I increase the sample rate of the recording

_____

_____

/3

Self Reflection

Beginner	Progressing	Embedded
I am still learning this and don't feel confident about my knowledge of the topic	I am getting there but don't understand everything about the topic	I have a really good knowledge of this topic and can answer questions confidently

Assessment Criteria	Beginning	Progressing	Embedded
Can write commands in Python including input, output and if statements to make programs			
Can use commands in Python including loops and functions to make programs			
Can convert binary and denary numbers			
Know how text, images and sound are stored in computers			

Complete the boxes below, identify things you have done well this term and things you want to improve on. This can be specifically about the topic itself or your own working practices

Things that I have done well are:

Things I need to improve are:

Assessment PRIDE

Effort 1 2 3 4

Strength

- ❑ You know the 4 main data types used in programming
- ❑ You can identify errors in Python programs
- ❑ You can follow and interpret simple programs
- ❑ You can write Python code that uses loops
- ❑ You can convert from binary to denary and from denary to binary
- ❑ You can add binary numbers
- ❑ You understand how images are stored in a computer
- ❑ You understand how sound is stored in a computer

Target

- ❑ Explain the difference between a Boolean and a string
- ❑ Give two ways how errors can occur in Python
- ❑ Correct your answers to Q4
- ❑ Write the code to solve Q7. Use your existing programs on the computer for support
- ❑ Convert 01110101 to denary by drawing a table
- ❑ Add together 11010 and 01110 using the rules
- ❑ Draw your own image in binary
- ❑ Explain how analogue sound is converted into digital
- ❑ Explain what ASCII is and what it is used for

Presentation and SPaG

- ❑ Correct any missing capital letters
- ❑ Correct your spellings
- ❑ Add in missing full stops/ apostrophes
- ❑ Write in black pen
- ❑ Draw diagrams in pencil

Student Improvement:

_____

_____

_____

_____