



## Maths Quiz

**Program Aim** Can the users create and improve a Maths quiz

**Maths Concepts** -Equals means the same as

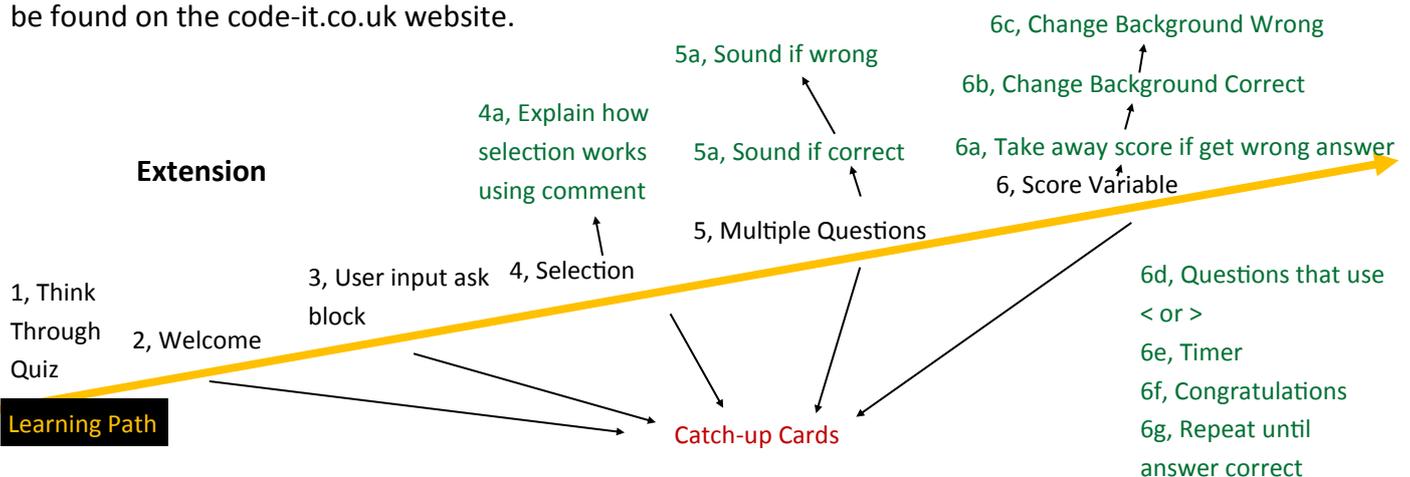
### Computer Science Concepts

- if else selection
- Broadcasting to trigger other blocks
- Variable for score
- Algorithm to code

### Computing Program of Study

design, write and debug programs that accomplish specific goals, use sequence, selection, and repetition in programs; work with variables and various forms of input and output

**Differentiation and Assessment for Learning** This planning should be used alongside the *catch-up cards* supporting SEN pupils and the *extension cards* supporting the more able, although in this module a lot of the extension tasks can be given orally. At the beginning of each session the *learning intention sheet* is shared and the learning journey expanded through success criteria. Pupils feed their progress back to the teacher through annotating this sheet with smiley faces at the end of the lesson. Teachers can also annotate the sheet to indicate those who need more or less help in future lessons. These extra resources can be found on the code-it.co.uk website.



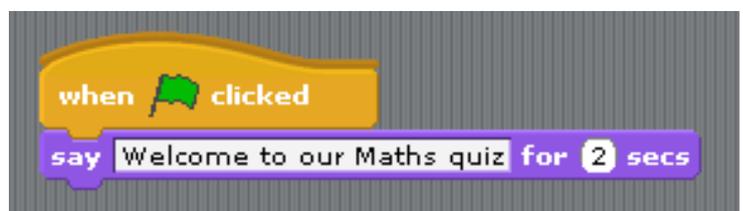
### 1, Think through steps (algorithm) to ask a quiz question

Ask a simple maths question to the class. Get them in pairs to work out the steps. Draw out that you need to think of the question and then think of the answer before you can ask the question. Once the user has answered the question ask them how they will know if the question is right or wrong. Indicate that you have got your answer and their answer. You are looking for a pupil to indicate that they need to be compared. Push for type of comparison as same. You can then ask what the maths symbol for the

same is (=). Then go through quiz pdf which highlights the process graphically.

### 2, Welcome

Explain that this program will mostly be one sequence of instructions from start to finish. Demonstrate a green flag starting block and welcome say block and give pupils a few minutes to complete. Have they tested their code?





Junior

omputer



code-it.co.uk

cience

## Maths Quiz P2

### Common Mistakes

Pupil types in word answer instead of using the blue answer block

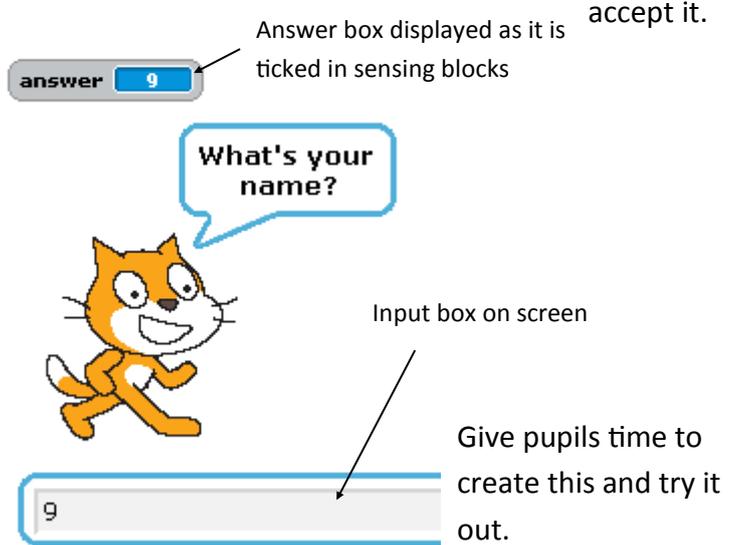
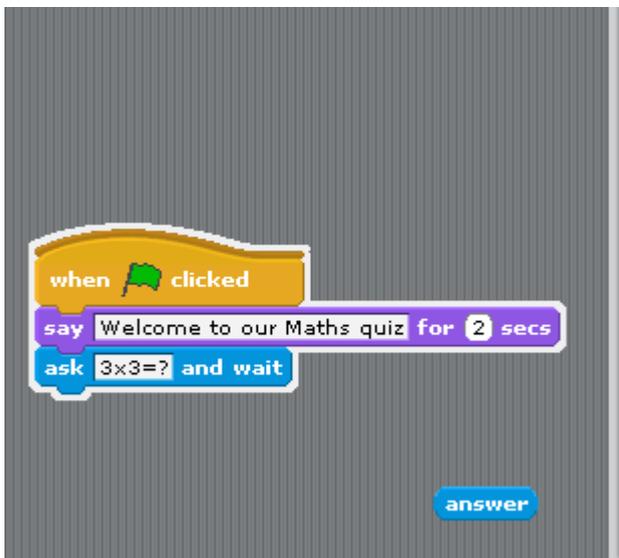
Ticked answer box



### 3, User Input Ask Block

Drag the ask and the answer blocks out. Explain that this is a user input block. It **puts in** information from the person taking the quiz. Now tick the answer block so that it is visible on the screen. Explain that whatever is typed into the ask input block goes inside the answer block. Type in a Maths question that everyone will know the answer to and watch as it appears in the answer block after you tick to

accept it.

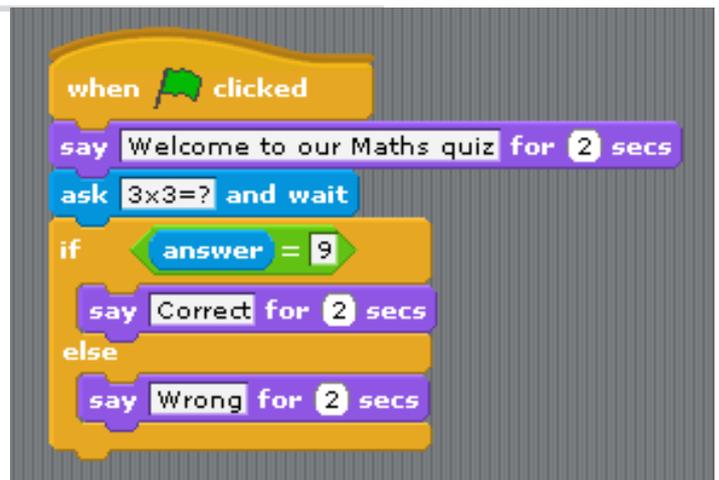


### 4, Selection

If this is the first time that pupils have met the idea of selection I always use a number of [real world examples](#) before demonstrating how it works in a quiz. The last examples is important as it paves the way for the code. Make sure in the last example you draw out that equals means the same as.

Drag out these blocks and arrange them like this. Make sure you draw attention to the order especially when inserting the equals block. Talk

through this before running your program. Pretend to be a user who gets an answer right then. Pretend to type the answer and explain that it goes inside the answer block (point) then the program checks to see if it is the same (point to equals sign) as the right answer this means that the condition has been met and only the top correct block is run. Do the same for a wrong answer pointing out that the answer is not the same as the right answer so only the else block is run.





Junior

omputer



code-it.co.uk

cience

## Maths Quiz P3

### 4a, Explain how selection works using a comment box

Pupils could also explain to a friend and record using mobile devices.

### 5, Multiple Questions

Demonstrate how to drag out another question underneath the first and ask pupils to create as many questions as they can in the time.

Don't show pupils how to duplicate the blocks until they have done a few more block by block.

Now demonstrate how to duplicate blocks by dragging out a single question and right clicking on the top most blue ask block and choosing duplicate. You must right click on the top block or not all the code will be duplicated.

#### Common Errors

Once pupils have started to duplicate it is very easy to snap the question inside the bottom else part of the selection. Can they tell you when the second question would be asked? ANSWER Only if you got question 1 wrong.

```

ask 4x2= and wait
if answer = 8
  say Correct for 2 secs
else
  say Wrong for 2 secs
ask 10x10 and wait
if answer = 100
  say Correct for 2 secs
else
  say Wrong for 2 secs

```

```

when clicked
say Welcome to our Maths quiz for 2 secs
ask 3x3=? and wait
if answer = 9
  say Correct for 2 secs
else
  say Wrong for 2 secs
ask 4x5=? and wait
if answer = 20
  say Correct for 2 secs
else
  say Wrong for 2 secs
ask 3x7=? and wait
if answer = 21
  say Correct for 2 secs
else
  say Wrong for 2 secs

```

#### Common Errors

Another common error is to use say blocks without timings. This runs the say block so fast that the user can't see them at all. Comparing their code with yours is normally enough for most pupils to identify the error.

```

ask 10x10 and wait
if answer = 100
  say Correct
else
  say Correct

```

### 5a, Sound if Correct

This is good to identify if pupils really understand what selection is really doing and some pupils will create this automatically without asking.

### 5b, Sound if incorrect

```

ask 10x10 and wait
if answer = 100
  play sound Fairydust
  say Correct for 2 secs
else
  say Wrong for 2 secs

```



## Maths Quiz P4

### 6, Score Variable

Ask pupils what possible scores the user could get from a three question quiz if each question is worth one mark each? (0,1,2 or 3) Ask them if we know what score a user will get before they take the quiz? This is important as many pupils won't have considered the range of possibilities and may have already awarded a 'mark' via a say command at the end of each question.

Ask pupils what it means if we say that the weather is variable? Draw out that it means changeable. Explain that we can get the program to keep score for us by creating a variable which will change depending on the users answers.

Now create a variable called score

Ask pupils what score the user should start with? Drag out the **set score to 0** block and ask pupils where it should go?

Now drag out multiple **change score by 1** blocks and ask pupils to decide where they should go to put a point into the variable?

If pupils come up with the wrong location go with it and run the program to show them otherwise. A common one is after each question. Get each answer wrong to demonstrate that this is the wrong location.

When pupils have come up with the right location talk through the code to explain it before running it. A great way to do this is with a box containing 4 pencils labelled with score that a pupil holds. Model setting the score to 0 by removing all the pencils. Model adding a pencil into the box if you get it right.

Finally run the code getting some answers right and some wrong before encouraging them to add a variable.

```
when clicked
set score to 0
say Welcome to our Maths quiz for 2 secs
ask 3x3=? and wait
if answer = 9
change score by 1
say Correct for 2 secs
else
say Wrong for 2 secs
say join Your score so far is score for 2 secs
ask 4x5=? and wait
```

You may also wish to show pupils how you can get the sprite to say the score through the use of a say, join and score variable. Pupils love the cleverness of this.

### Common Mistakes

Inserting **set score to 1** blocks instead of **change score by 1**. This sets the score back to 1 every time it is used.

Missing out a **set score to 0** block

This means that when the quiz is run the second time the old score is still inside the variable and will be added to instead of having an empty score when you start.

### 6a Takeaway score point if user gets answer wrong

Adding a change score by -1



Junior

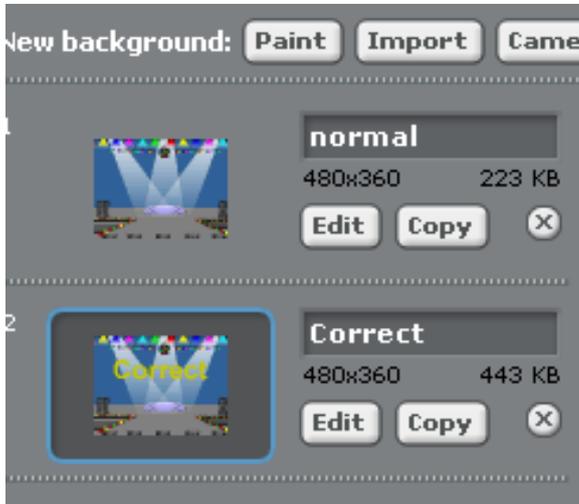
omputer



code-it.co.uk

cience

## Maths Quiz P5



### 6b, Change Background to Correct if Answer Correct

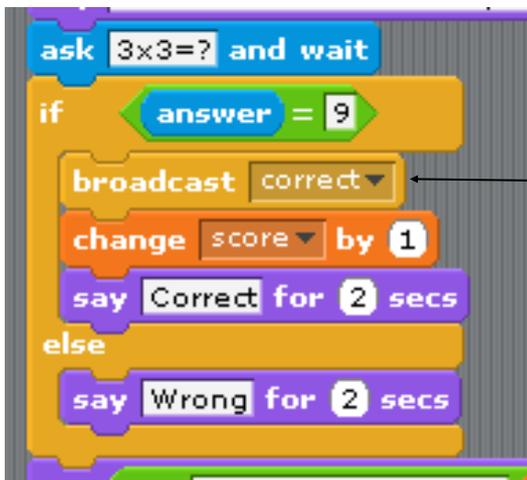
As this might be their first experience with broadcasts I normally take a break away group aside to go through this.

Get pupils to import two backgrounds and change one by adding the word correct on it. Change the names of the backgrounds to normal and correct. Give pupils a few minutes to do this.

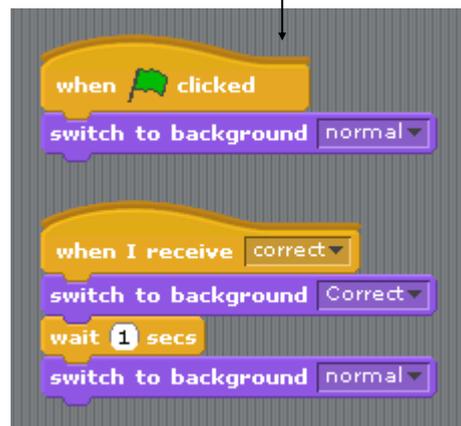
Gather pupils back and whisper in two pupils ears that is you say the word correct they must jump up and down twice and then sit down.

Explain to the group that we are going to use a broadcast command to change the background when the users gets a question correct. When you say correct the two pupils will jump up and down. Make sure you use correct more times. Explain that this is like our quiz code. We can all hear the broadcast command but only those who have the code attached will know what to do with it.

Now setup the code as shown starting with broadcast in sprite and then moving on to stage/screen code.



Broadcast in sprite



Stage/Screen code

### 6c, Change Background if User Gives Wrong Answer

This is a good independent task to see who understood the use of a broadcast

#### Assessment Statements

**Well above average** 6e with variable, 6f or 6g completed

**Above average** 6a or 6b completed correctly

**Below average** basic variable not completed correctly after debugging

### 6d, Code that uses less than or greater than

Can pupils create questions such as type a number less than 56.

Use a > or < block instead of an = one



## Maths Quiz P6

My thanks to [Vikki Dodd](#) Head of Computing Queen Elizabeth's Grammar School Blackburn for these extra challenges

### 6e, Timer

Can pupils create a timer for the quiz which ends the game after a set period of time. This could be further extended so that the user could select a difficulty level which results in a longer or shorter time period. Pupils could use simple wait for x seconds time blocks although this will not show the user the time counting down or a time variable inside a loop which is the better solution.

```
when clicked
wait 28 secs
say 2 seconds to go for 1 secs
say 1 seconds to go for 1 secs
stop all

when clicked
set timer to 30
repeat 30
wait 1 secs
change timer by -1
stop all
```

### 6f, Congratulations

Add code at the end of the quiz that congratulates the player IF they got all of the questions correct ELSE it tells them their score. Pupils would need to use an if else selection blocks.

```
if score = 10
say Congratulations you got all the answers correct for 2 secs
else
say join join Your score was [score] out of 10 for 2 secs
```

```
repeat until answer = 9
ask 3x3=? and wait
if answer = 9
say Correct for 2 secs
else
say Wrong for 2 secs
```

### 6g, Repeat until answer correct

Can pupils get a question to repeat until the user answers it correctly?  
Ask if this solution would affect a scoring variable?