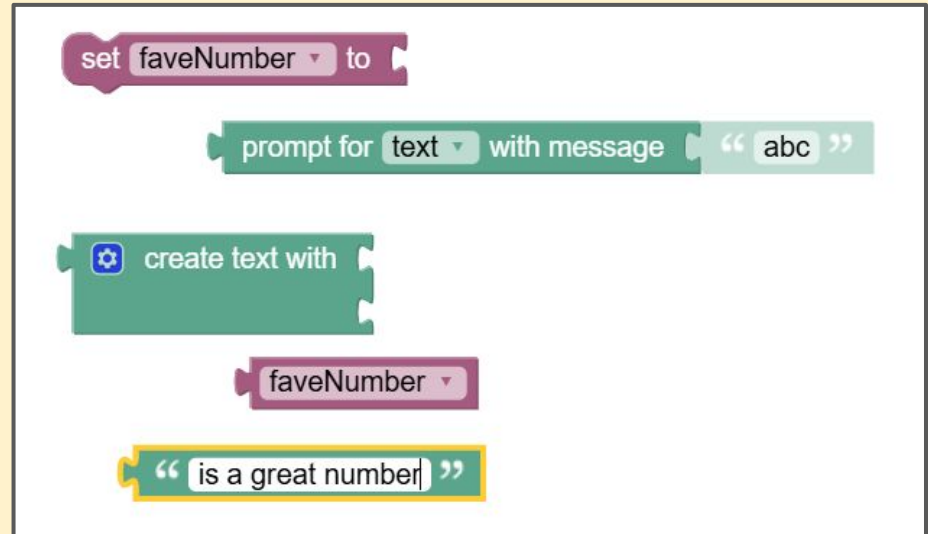


Challenge 1:

Write a program that:

- asks the user for their favourite number
- stores the number in a variable called faveNumber
- prints a message that says faveNumber “is a great number”

You should only need these blocks:



Hint: create the variable faveNumber first

Challenge 2:

Write a program that:

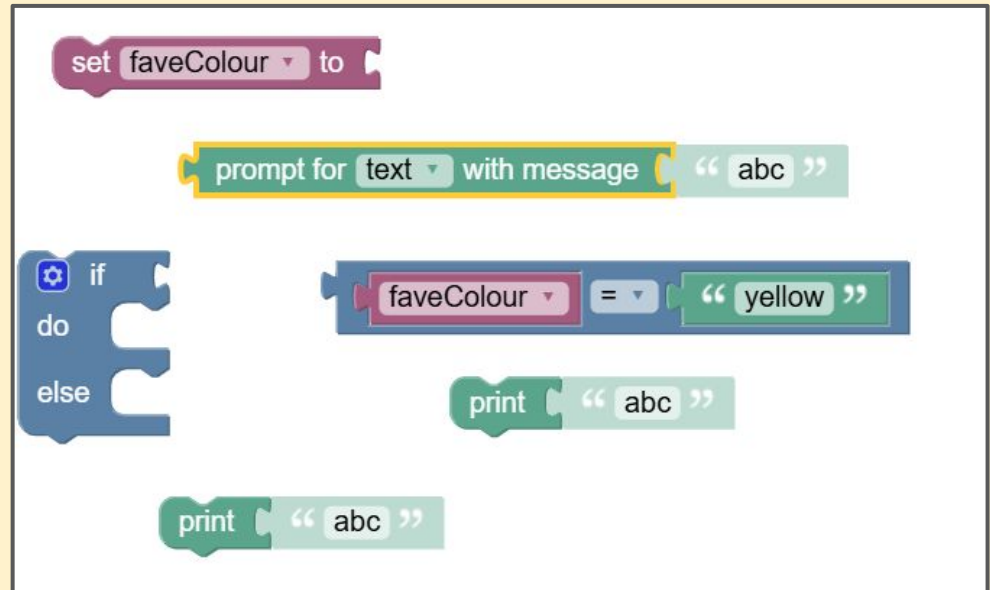
- asks the user for their favourite colour
- stores the colour in a variable called faveColour
- if the colour is yellow says "That's the best colour"
- otherwise says "Nice, but yellow's better"

You should only need these blocks:

Building the if – do – else block is tricky

The next slide shows you how to do that

Hint: create the variable faveColour first



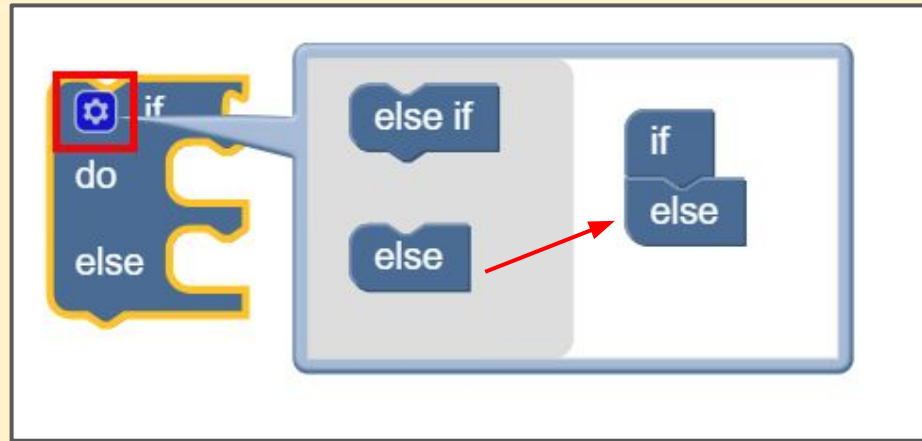
The image shows a collection of Scratch code blocks arranged in a sequence. At the top is a purple 'set faveColour to' block. Below it is a green 'prompt for text with message' block with the message 'abc'. To the left is a blue 'if do else' block. To the right of the 'if' block is a blue 'faveColour = 'yellow'' block. Below the 'if' block is a green 'print 'abc'' block. At the bottom is another green 'print 'abc'' block.

Building an IF – DO – ELSE block:

The if – do – else block is in the logic section
Building it is tricky

Here's how it works:

1. **Drag** an if – do block out to the coding area
2. Click the **blue cog** at the top left
3. A pop up section appears
4. **Drag** the **else** to underneath the do in the little pop up section
5. The block on the coding area should change
6. **Click** somewhere on the coding window to get rid of the pop up section



Hint: ask for help if you need it here – this is properly tricky

Challenge 3:

A problem...

For challenge 2, if you type “yellow” the program agrees with you

But if you type “Yellow” (or “yeLLow” or even “yellow “ (add a space) the program disagrees

The computer thinks “yellow” and “Yellow” are two different things because of the capital letter

Computers often do this. It can be annoying

Here’s one way to fix that



Challenge: Fix the program so that it will accept capital or lowercase letters

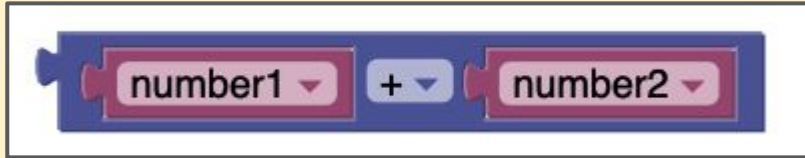
Hint: add the block shown straight after the user enters their colour

Challenge 4:

Write a program that:

- asks the user to enter two numbers (one at a time!)
- adds the two numbers together
- prints a message that gives the answer

You can add numbers together using this block:



Hint: you will need to make at least two variables. You might want a variable called answer as well

Challenge 5:

Write a program that:

- asks the user to enter two numbers (one at a time!)
- takes the second number away from the first number
- prints a message that gives the answer

Then **multiply** the two numbers

Then **divide** the first number by the second number

Then try and work out what the other maths operator (^) does (ask if you need help)

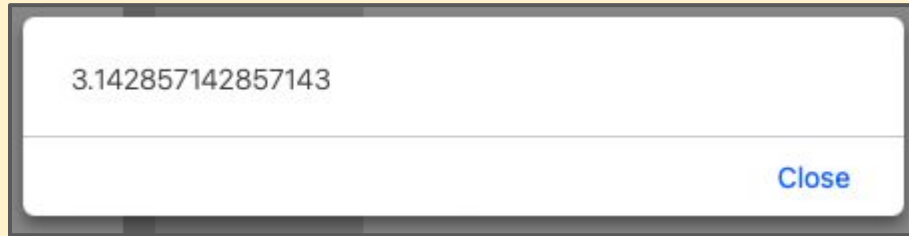


Challenge: try and make the final output as useful as possible (“The result of taking X away from Y is...”)

Challenge 6:

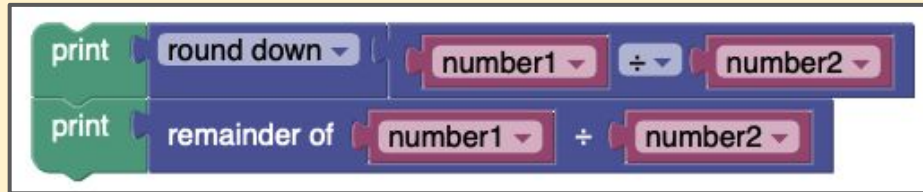
Write a program that gives the answer and the remainder when you divide

When you divide two numbers this sometimes happens:



I don't want that. I want to know that the answer to $22 \div 7$ is **3 remainder 1**

You can do this using this method:



Challenge: Make the answer go on a single line and make the output more helpful (“X divided by Y is Z remainder W”). **Hint:** You may want to use variables for this

Challenge 7:

Write a sweets challenge program that could be used to help teach division to primary school children

- ask how many sweets there are
- ask how many people there are
- give the answer to how many sweets each person gets
- give the answer to how many sweets are left over

Challenge: Make the output from the program easy for a primary school child to understand

Challenge 8:

Use code blocks from the text and maths folders to write a program to throw a six sided die. You will need to use the block shown:



Key knowledge: an **integer** is a whole number. This block will create a random number between two integers

Challenge: make sure the output is shown nicely

Challenge 9:

Write a program to throw a twenty sided die:



Key knowledge: an **integer** is a whole number. This block will create a random number between two integers

Challenge: make sure the output is shown nicely

Challenge 10:

Write a program to throw two six sided dice and show the result of the numbers added together

- use two variables to hold the numbers thrown on each die
- use the skills from challenge 4 to help add them together



Key knowledge: an **integer** is a whole number. This block will create a random number between two integers

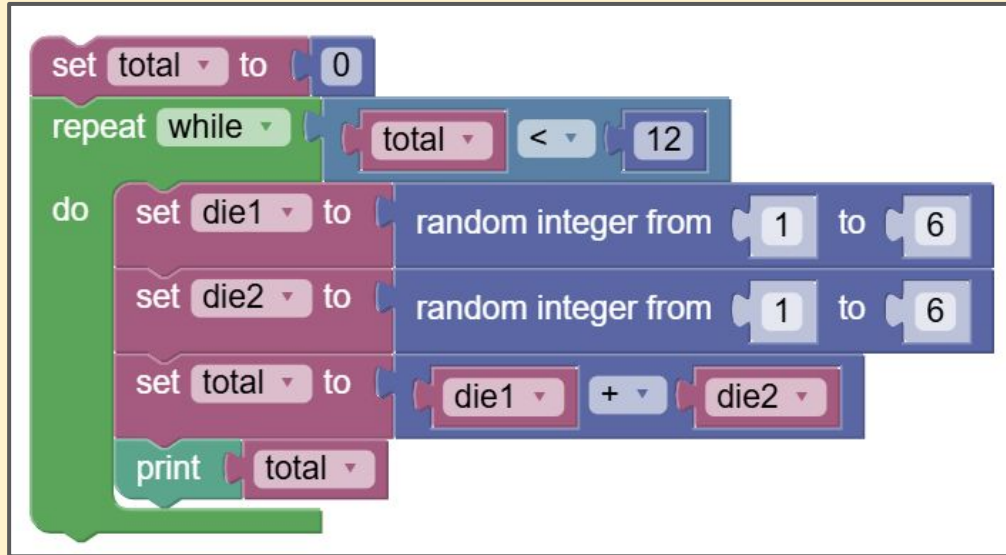
Challenge: make sure the output is shown nicely

Challenge 11:

Write this program

It throws two six sided dice until the total is 12

Count how many times you have to throw the dice before the total is 12



Interesting: the average number of times to throw the dice until you get a 12 is 36 times!
You can stop the program by using the reset button!

Challenge 12:

Change your program from challenge 11 to count the number of throws needed before you get a 12

You will need to:

- create a new variable called throws
- set throws to 0 at the top of the program
- add one to throws each time you throw the dice
- add a print throws block right at the bottom of the program - outside the loop

Run the program a number of times to get the average number of throws needed

Interesting: take out the print total blocks to speed things up

Hint: speed the program up!

Challenge 13:

Change your program from challenge 12 to use three six sided dice and count the throws needed to get 18

The average number of throws needed is 216!

Run the program a number of times to get the average number of throws needed

Hint: take out the print total blocks to speed things up

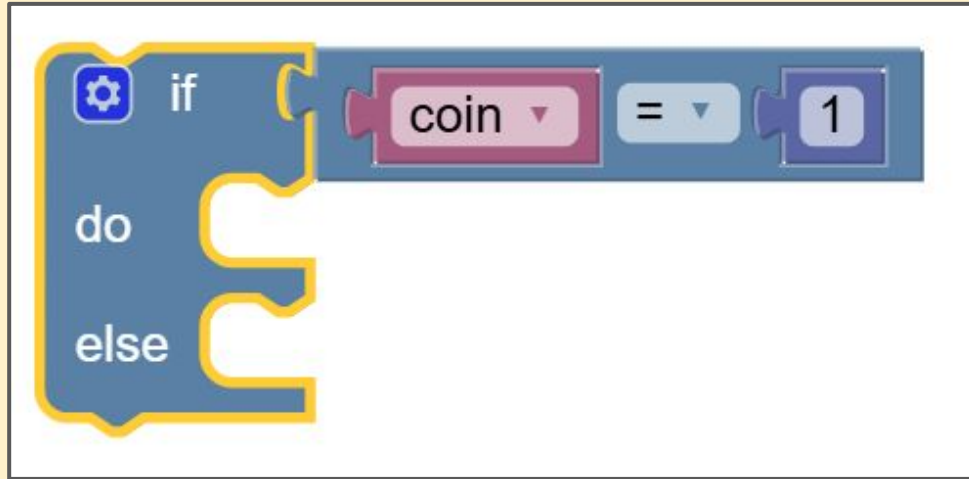
Hint: speed the program up!

Challenge 14:

Write a program to throw a coin

- use the random integer block
- make a 1 mean a head and a 2 mean a tails
- use an if - do - else set of blocks to report the result

Slide 3 tells you how to set up if - do - else blocks



Challenge: can you write a program to repeat this until you get five heads in a row?