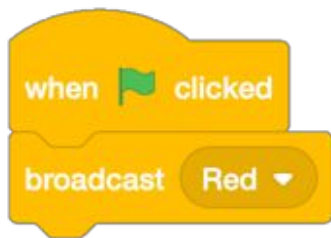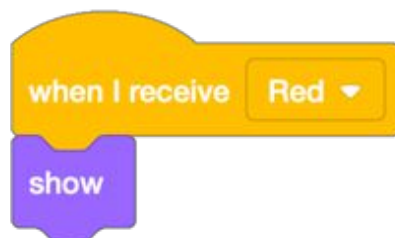# Computational Thinking Cheat Sheet

Broadcast

An important block used in Scratch is the Broadcast block. This allows different sprites to communicate with each other.

For example if we want a Red light sprite to show and other lights to hide at a certain time we could use the following blocks:

In the background code: In the Red Light Sprite: In the other lights:

We could then use this same idea for the next stage, when the red and yellow lights are on. Though we would want to wait for a certain amount of time before broadcasting that new message.

Waiting

Occasionally in a program we would like a period of time between actions. A useful block in Scratch to do this is a Wait block. There are two types, both with different uses.

Wait for an amount of time: Wait until something happens:

Computational Thinking Cheat Sheet

Using Variables For Buttons

We can create buttons in Scratch by creating a sprite and the "When this Sprite Clicked" block.

Combining this with a variable we can set it to be a value of 1 when it is pressed, and then set it back to 0 when it is reset.
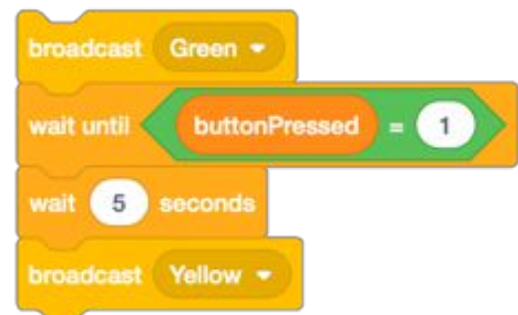
To create a variable, click on the orange variables category and click the "Make a Variable" Button. Then we can call our variable something sensible such as "buttonPressed".
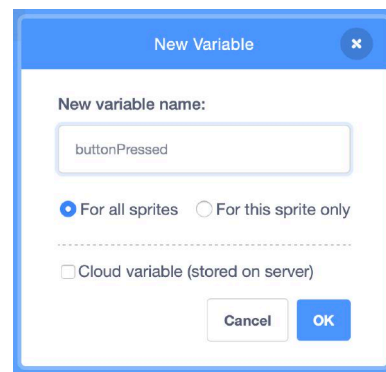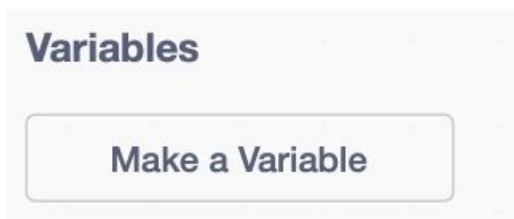
When the sprite is clicked we can set our variable to be 1:

Now that a variable is keeping track of when the button is pressed we can use this to allow another algorithm to continue:

Using Variables For Movements

If we want a car or person to stop as if reacting to a traffic light, we need to make it approach the crossing and then depending on whether the light is red or not, to stop or carry on.

We need them to move until they reach a point on the screen. For example, if a car is driving from the left of the screen to a crossing:

We are repeating until the car has reached an x position (across the screen) which is more than 50.

It then waits until a variable named carStop is 0 i.e. the light is green. If the variable was 1 i.e. red light, then the car would wait in that position.

Once carStop = 0 i.e. green light shows, the car moves on again until it is passed the right side of the screen (x = 240).

You could use this idea to make a car drive in the opposite direction too. You would need to adjust the conditions and the values but the algorithm itself is performing the same task.

You could also use this for pedestrians crossing, however they would be moving in the y-direction instead of the x-direction.

Computational Thinking Cheat Sheet

Changing Costumes

We would want more than one car to drive down our street so we can repeat the instructions once it has driven off the screen.

However, we don't want the same car driving past every time so we can select some other cars as costumes.

To do this we can select our car and click on "Costumes":

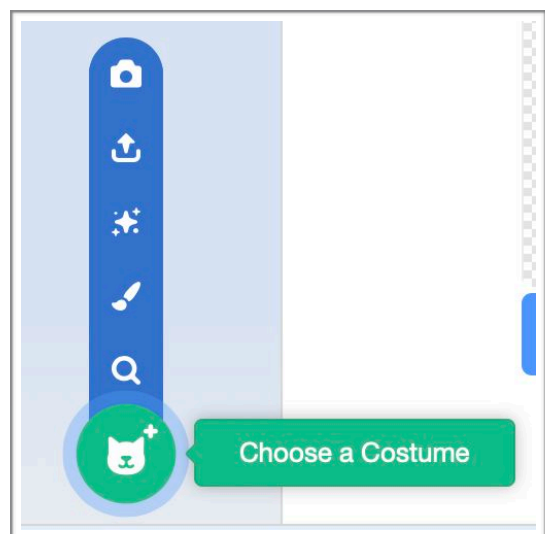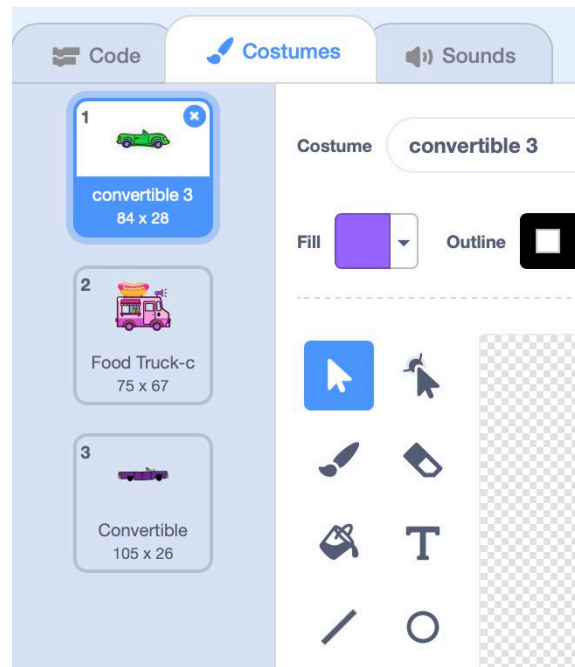We can then choose other vehicles as costumes for our sprite.

We can then use the

block and combine this with our car algorithm already and repeat the whole process forever using a forever block.

The same idea can be applied to different pedestrians.

Computational Thinking Cheat Sheet

Key Words

Algorithm
Decomposition
Abstraction

Pattern Recognition
Sequence
Iteration

Algorithm – is a set of simple instructions that are done in a certain order to solve a problem.

Decomposition – is the process of breaking a complex problem into smaller component parts.

Abstraction – is the process of removing unnecessary detail and simplifying. Abstraction is used to remove unnecessary detail from a real-world situation and to model the simplified result in an algorithm or program.

Pattern Recognition - Is the process of finding similarities in data or small problems in order to complete more complex tasks.

Sequence: an action, or event, leads to the next ordered action in a predetermined order.

Iteration: An iteration is a single pass through a set of instructions. Most programs contain loops of instructions that are executed over and over again. The computer repeatedly executes the loop, iterating through the loop.