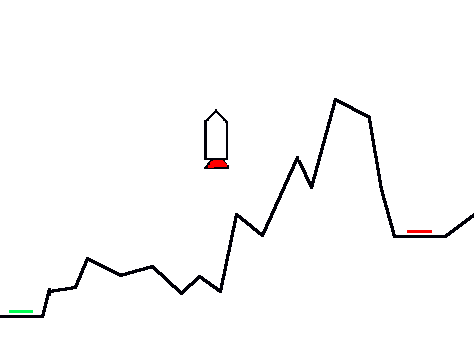
# Moonlander

In this game the player must guide a rocket from the launch pad (green) to the landing pad (red) without crashing into the landscape.



# Step 1: Getting the rocket to move up and down – 1 point

We want the rocket:

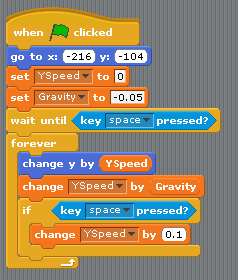
* To start off stationary on the launch pad.
* To accelerate upwards when the player holds down the space bar.
* To slow down and eventually fall when the player stops holding down the space bar.

The first thing we need to do is to make some variables to control the rocket’s speed. A variable is a way of storing information in the computer’s memory, for example: how fast the rocket is currently going, or the rocket’s current horizontal position.

1. Click on the “Variables” button.
2. Click on “Make a variable”, call it “Gravity” and make available “For this sprite only.” This variable will control the strength of gravity on the rocket.
3. Make another variable called YSpeed, available “For this sprite only.” This variable will control how fast the rocket moves vertically.
4. Tick the box next to YSpeed so that it appears in the stage.



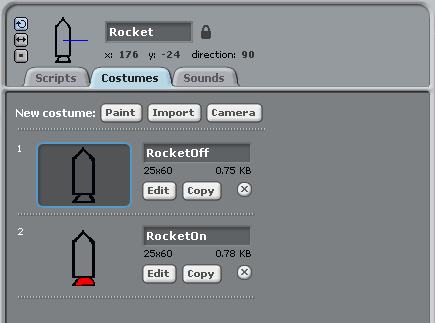
Now, copy this code into the Rocket’s script:



Now press the green flag to start your script, and test the rocket’s movement by pressing the space bar. Make sure that it falls down when you let go of the space bar!

# Step 2: Displaying the thrusters when the rocket is accelerating – 1 point

The rocket has two costumes so that the player can see when the thrusters are on:



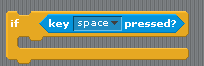
We want the rocket:

* To start with the thrusters turned off.
* To turn on the thrusters when the player is pressing the space bar.
* To turn off the thrusters when the player is not pressing the space bar.

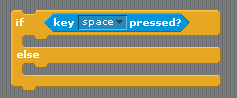
Can you work out how to make the rocket do this?

## Hints:

You want to replace



with



Now you need to figure out where in the script to insert these two blocks:

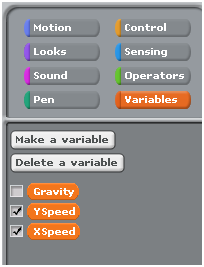




Now press the green flag and test that the rocket’s thrusters work correctly.

# Step 3: Getting the rocket to move left and right – 2 points

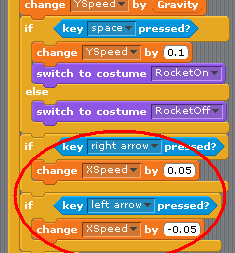
The first thing we need to do is to make a variable called XSpeed to control the rocket’s horizontal speed.



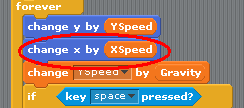
We want the rocket to start off with zero horizontal speed, so that it is standing still. Add the following block into the start of the script:



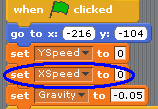
Now add in the following blocks to allow the player to control the value of the XSpeed variable by using the left and right arrow keys:



and the following block so that the rocket moves horizontally at the speed of the current value of XSpeed:



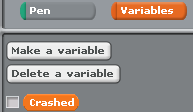
Finally, we want the rocket’s horizontal speed to start off as zero:



Now press the green flag to test out your game. Does the rocket move correctly?

# Step 4: Game over when the rocket crashes – 1 point

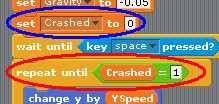
When the rocket hits the landscape, we want the game to end. To do this, we need create a new variable to store whether or not the player has crashed yet. Make a variable called “Crashed”:



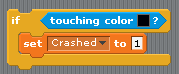
With this new variable, we will use:

* “Crashed = 0” to mean that the player has not crashed yet,
* “Crashed = 1” to mean that the player has crashed.

Now replace the “forever” loop with a “repeat until” loop. This new loop will make the game stop when the player has crashed. We also need to make sure the “Crashed” variable is equal to zero when we start the game:



Now you will need to add this block in somewhere:



Can you figure out where?

# Step 5: Game ends when the player reaches the landing platform – 2 points

Can you work out how to do this by yourself?

## Hints:

You will want to create another variable to store whether or not the player has landed, similar to the “Crashed” variable we just made. Your finished script should look a bit like this:



# Extra features

Now that you’ve made the game, you could try adding in these extra features:

* Some more costumes to the rocket so you can see when the left and right thrusters are on
* Have only a certain amount of fuel that can be used for the thrusters
* Have only a limited amount of time to complete the course
* Add some moving asteroids
* Keep a score based on how quickly you land and how much fuel you use
* Make a random moon surface each time