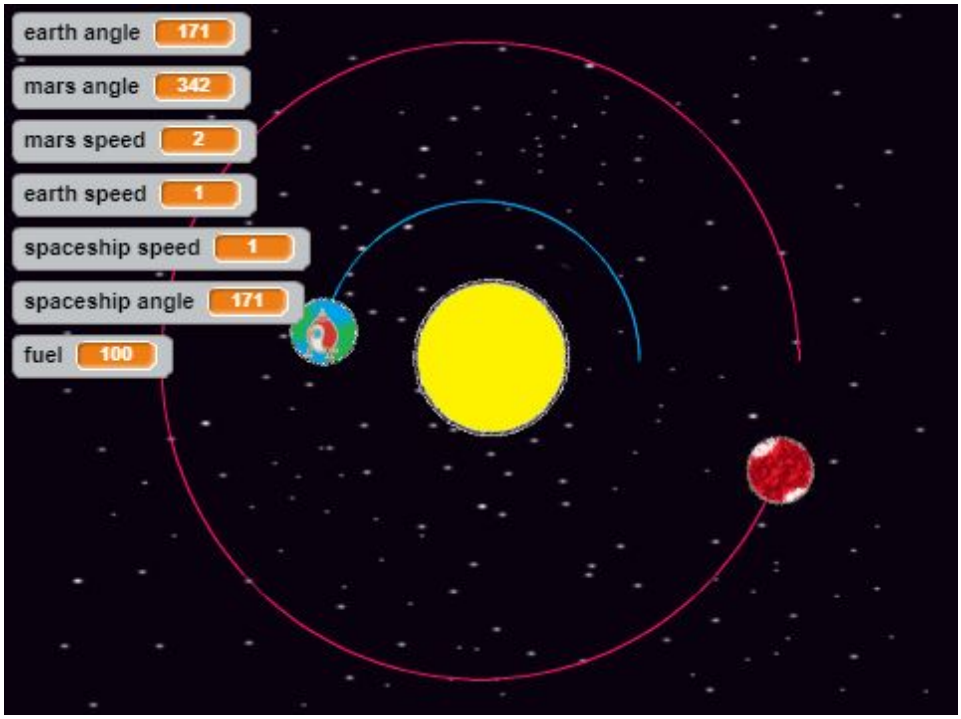
Session 2 - Hohmann Transfer Orbits

Introduction

In this session you are going to learn how to programme an animation which simulates how and when spaceships are able to fly from Earth to Mars. When we send spaceships to Mars we use a Hohmann transfer orbit as it uses the least amount of energy to transport a spaceship all the way there!



Step 1 : Earth and Mars Orbit the Sun

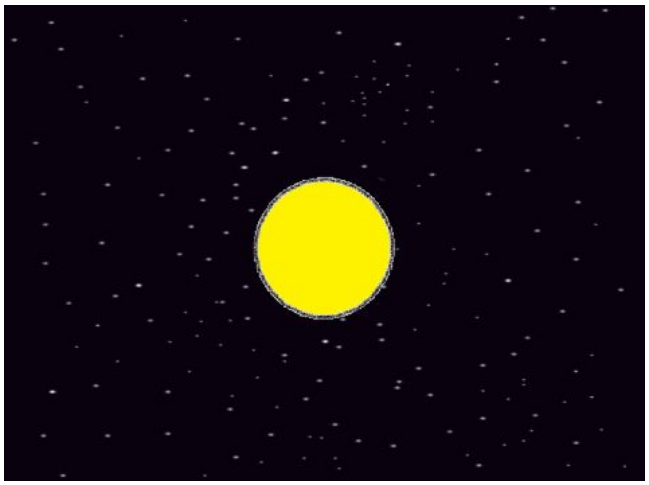
Let's make the Earth and Mars orbit the Sun!

Activity Checklist

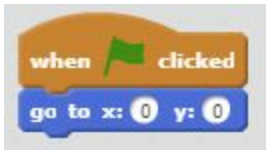
- ❏ Start by opening this link to the Scratch editor https://scratch.mit.edu/projects/205044178/#editor
- ❏ Delete the cat sprite from the project stage.
- ❏ Add the 'stars' backdrop from the backdrop library to the stage.

FOR THE SUN SPRITE:

- Add a Sun sprite. You can download this from the dropbox or create it yourself! Your project will now look like this:

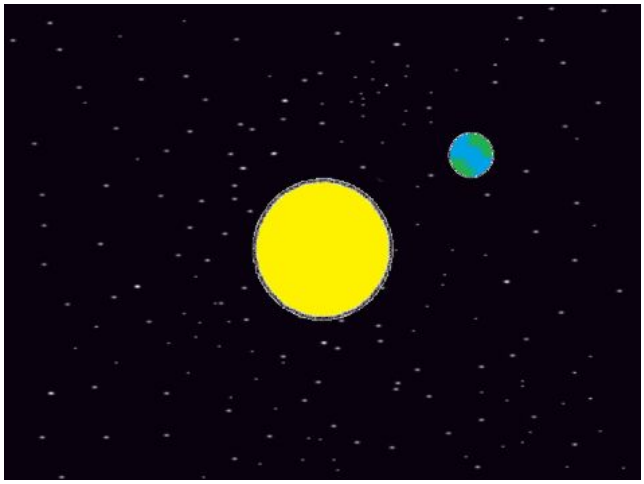


- To get the 'Sun' to sit in the middle of the stage we need to add the following code to the Sun sprite:



FOR THE EARTH SPRITE:

- Add an Earth sprite to the stage like so:



- To get the Earth sprite to orbit the Sun, firstly we need to make 2 new variables, which you should name ‘earth angle’ and ‘earth speed’

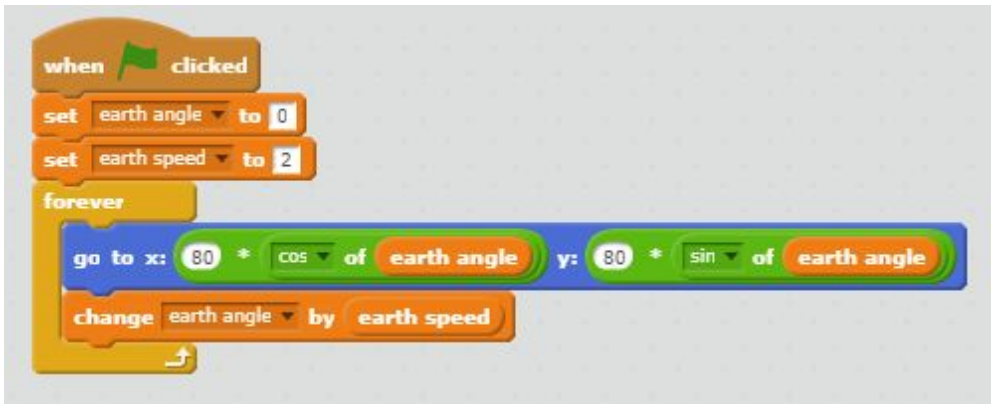
- When the green flag is clicked we need to set the ‘earth angle’ variable to 0 and the ‘earth speed’ variable to 1, so need to add the following code to the Earth sprite:

- We want the Earth to orbit around the Sun forever so we need to add a forever loop to the code like so:

- To get the Earth to move around the Sun we need to add 2 blocks into the forever loop. Firstly we need to add a 'go to block' and secondly a 'change by block'. The 'go to block' should look like this:

- EXPLAIN!
- Then underneath this inside the forever loop you need to put the following block:

- Therefore your code so far on the Earth sprite should look like so:

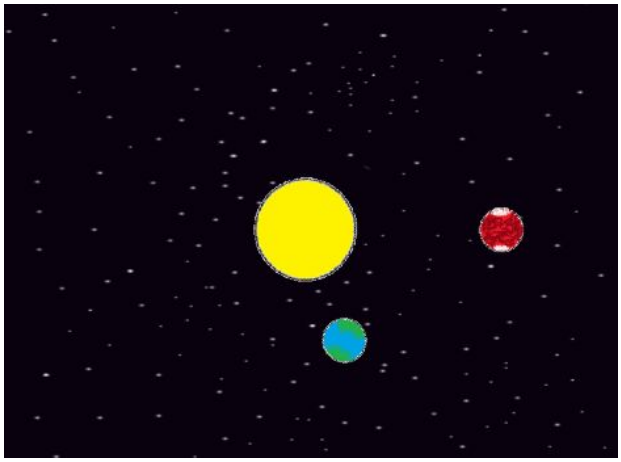


➔ **TRY THIS!**

- **Click the green flag! What happens?**
- **The Earth should orbit around the Sun!**
- **The earth angle variable should increase as the Earth moves around the Sun**

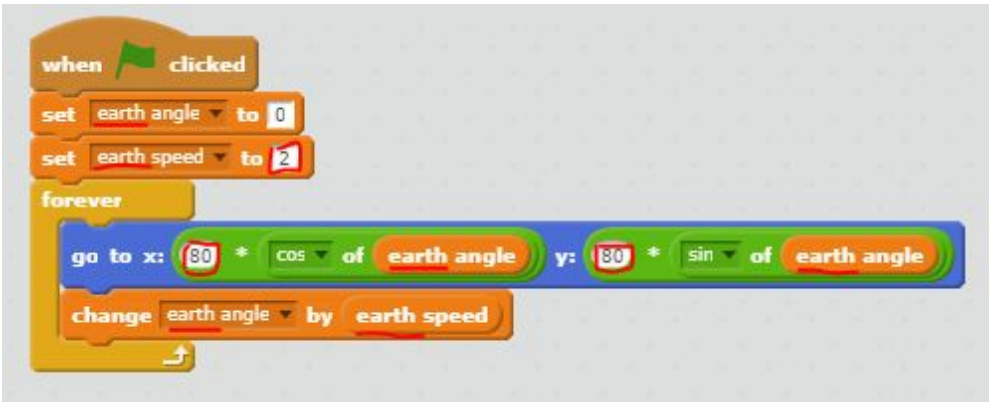
FOR THE MARS SPRITE:

- Now you need to add a Mars sprite, so that your stage looks something like this:



- Firstly you need to make two new variables which you should name ‘mars angle’ and ‘mars speed’ like so:

- You can duplicate all the code from the Earth sprite onto the new Mars sprite and you just need to change the following bits of the code (the parts highlighted in red):



- You need to change any variable that has the word earth in and change it for the corresponding variable just with the word mars in, so that the whole code on the Mars sprite looks like this:

- Finally you need to change the numbers previously circled in red pen like so:

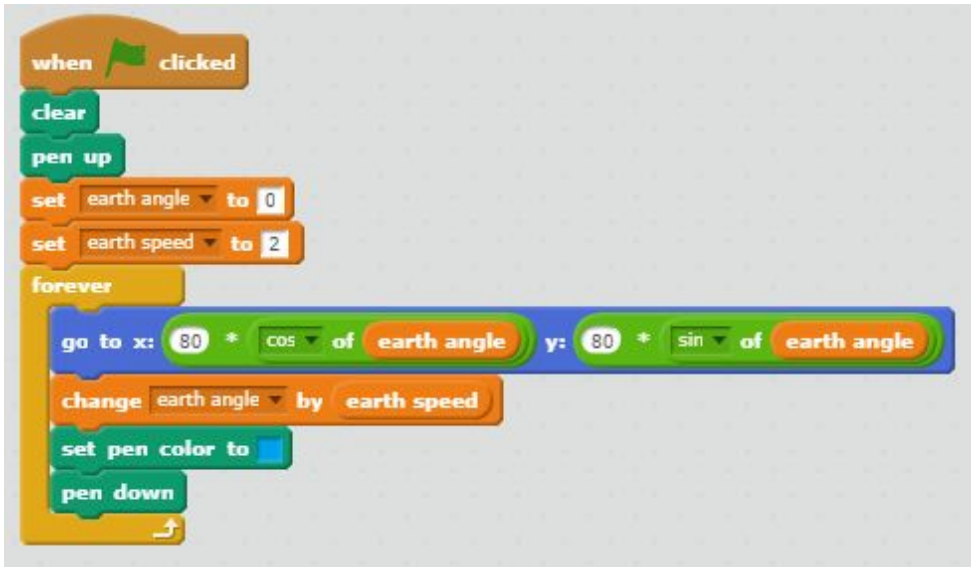
➔ **TRY THIS!**

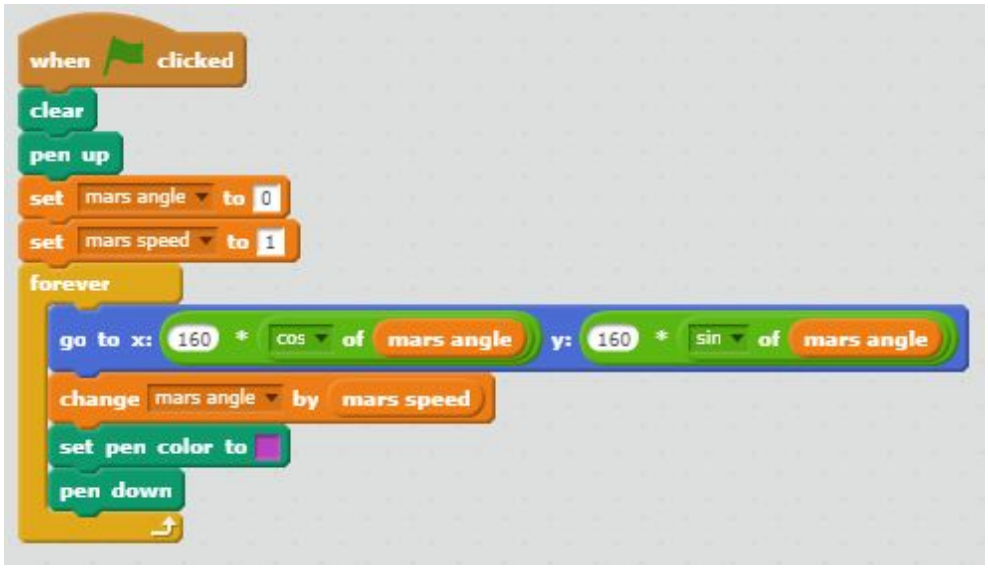
- **Click the green flag, do both Earth and Mars orbit the Sun?**
- **Is Earth orbiting closer to the Sun than Mars?**
- **What do you notice about the speeds of two planets? Earth should be moving twice as fast as Mars.**

EXTENSION

- **Can you get the Earth and Mars sprites to draw out the paths they follow around the Sun.**
- **To do this you need to use the following blocks:**

- This is what your code should look like if you want the Earth and Mars to draw their orbits out as they move:



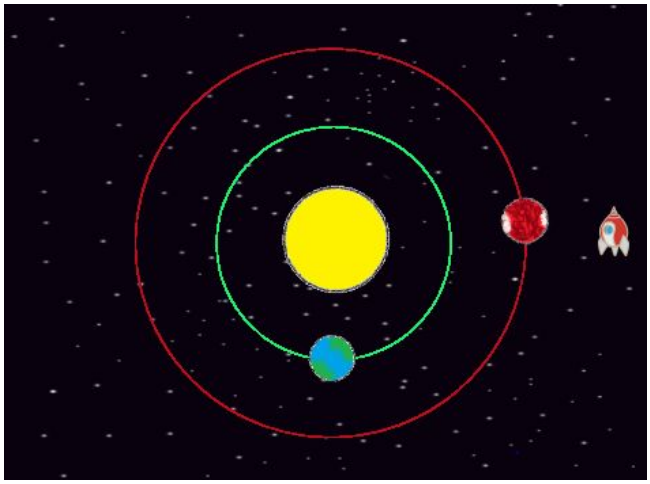


Step 2 : Spaceship Set Up

Activity Checklist

FOR THE SPACESHIP SPRITE:

- We will now code for the 'Spaceship' to sit on the Earth



- To get the 'Spaceship' to sit on the Earth while it orbits the Sun by coding it to forever go to Earth.

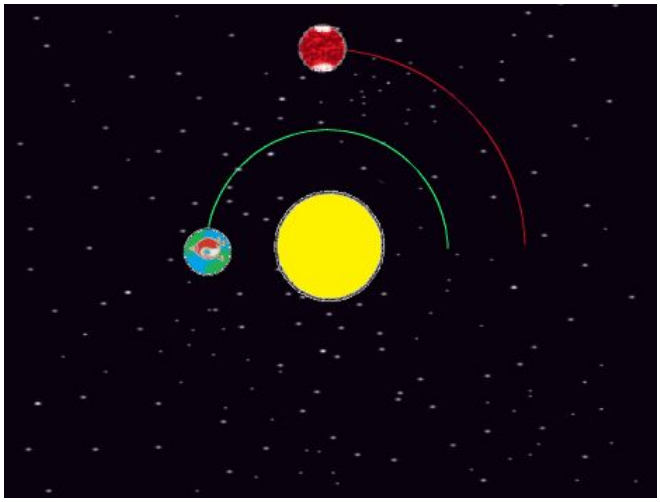
- To make the spaceship sprite look like it is sitting on the Earth before it takes off we need to add the 'switch costume to spaceship-b' to the 'Spaceship' sprite like so:

- The spaceship must point point to the Sun which is at x = 0 and y = 0, so add 

- To get the 'Spaceship' to sit on the Earth while it orbits the Sun by coding it to forever go to Earth.

➔ **TRY THIS!**

- **Does the 'Spaceship' sit on the Earth as it orbits?**



Step 3 : Spaceship Launch

FOR THE SPACESHIP SPRITE:

- For the spaceship sprite these variables have been pre-made:
  - Launch angle
  - Spaceship x
  - Spaceship y
  - Spaceship speed
  - Position

- We also have 3 blocks defined on the spaceship sprite:

- Like so:

1) Launch Angle

2) Spaceship x

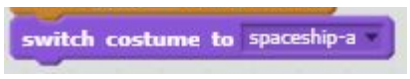
3) Spaceship y

Activity Checklist

- ❏ We need a trigger for the spaceship to launch For example the space key 

- ❏ 

and the spaceship is launched to make the spaceship move add 

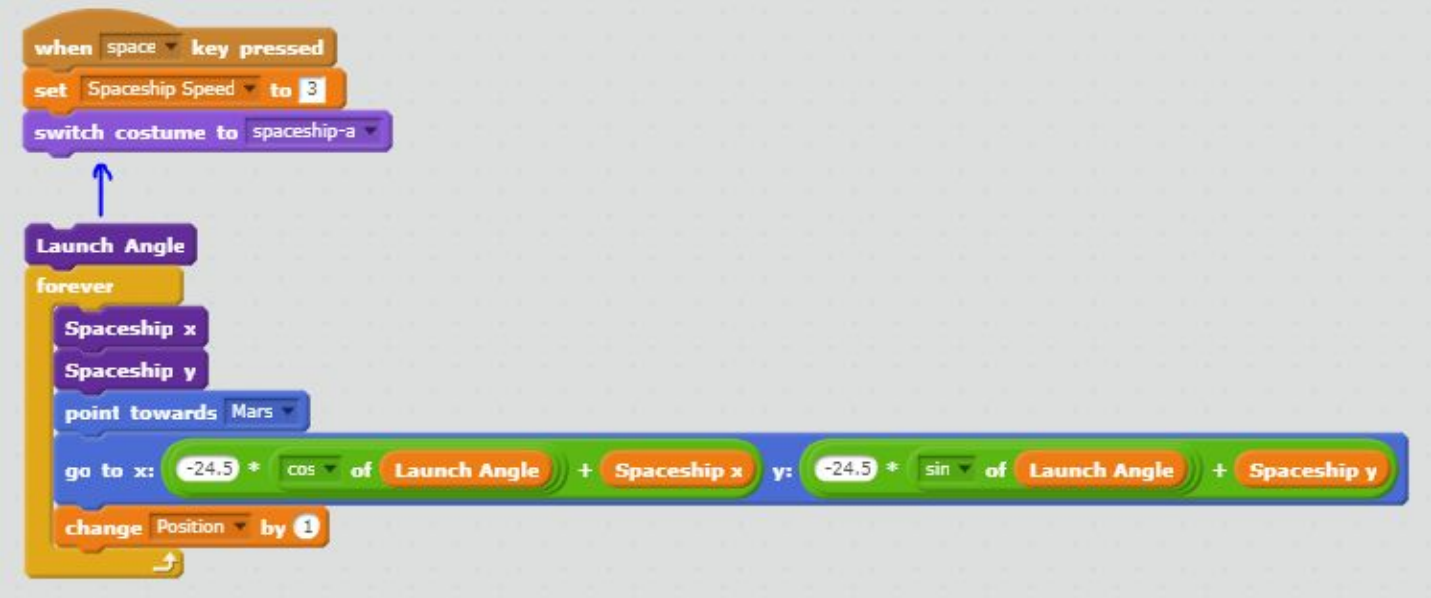
- ❏ When the spaceship has launched to make the spaceship look like it is flying switch the costume to spaceship-b 

.

- Your code should look like:

- We will now use the blocks made earlier to make:

- Put this block here:

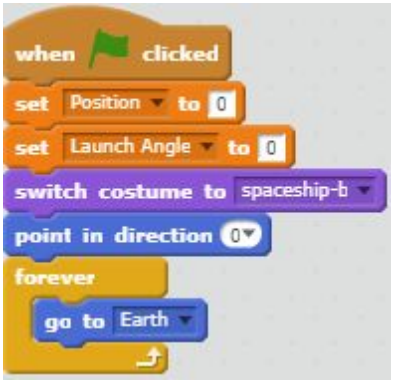


- To make the spaceship reset after each Launch, under 

we need to add

.

- Your code should now look like the following:



- **TRY THIS!**
  - **When the space key is pressed does the spaceship Launch?**
  - **Does it change costume?**
  - **What happens if you change the 'Spaceship Speed'?**

Step 4 : Spaceship Landing on Mars!

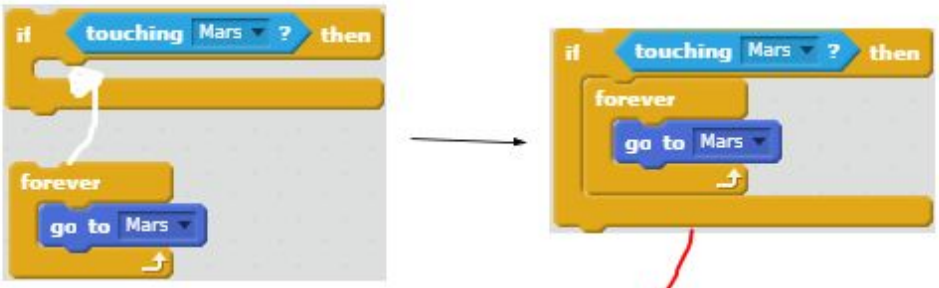
Activity Checklist

- We now need to get the spaceship to land on Mars!

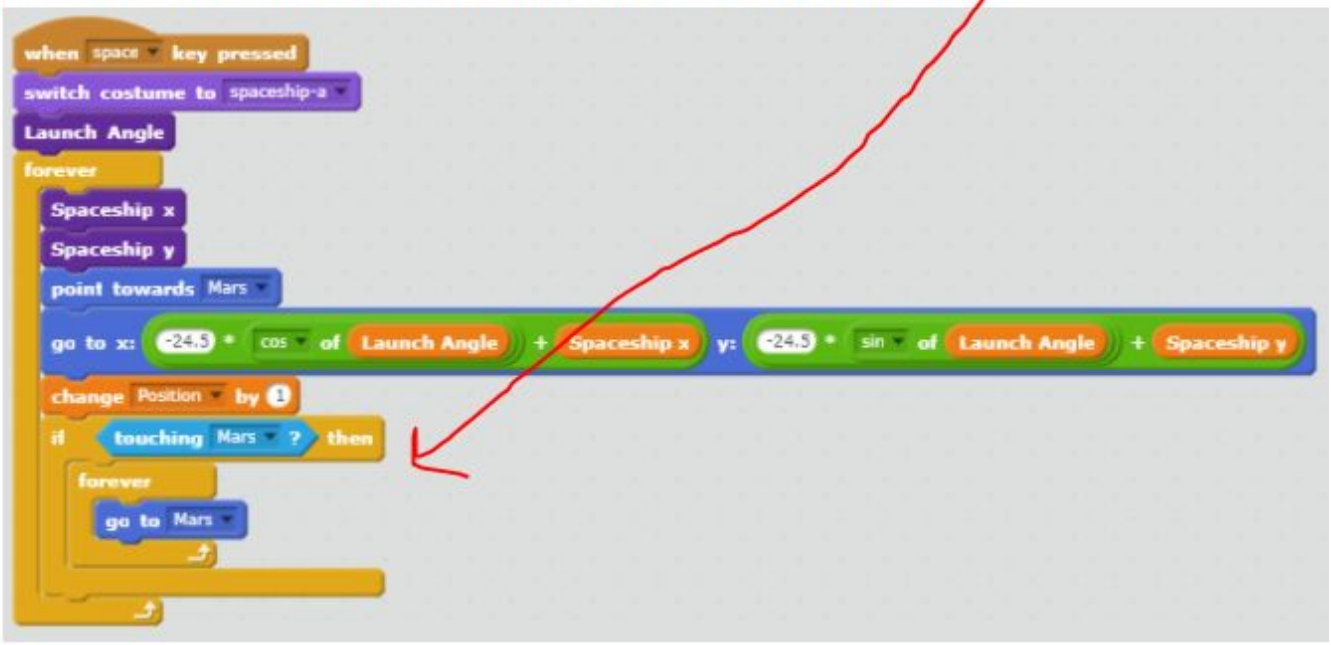
- We need to know if it is touching Mars using the **sensing block** and an **if loop**:

- To make the stay on Mars you must make the spaceship forever go to mars using:

- Put this inside the **if** loop



- Place this new block inside the forever loop.Your code should look like this:



- **TRY THIS!**
  - **When the spaceship hits Mars does it land on Mars and stay on it?**

Experiment!

If we change the set 'Spaceship Speed'. What happens to:

1) Launch angle
2) How easy it is to land

- Is it easier to land the spaceship on mars when it launches closer or further?
- If the 'Spaceship Speed' is bigger on Mars when it launches closer or further?
- Is the spaceship easier or harder to land on Mars when the 'Spaceship Speed' is higher?

Step 4 : Running out of Fuel?

What if we run out of fuel?

Timing is important when launching a spaceship to Mars as you want it to travel the least distance. **The less distance, the less chance of running out of fuel**. This is a really important consideration and we will now code it!

Activity Checklist

FOR THE SPACESHIP SPRITE:

Firstly we need to create a new variable called 'Fuel'. (see bottom of guide on how to create a new variable).

- Chose the amount of Fuel you want at the beginning of the game by putting 

under 

.

- We only want the fuel to go down once the spaceship has launched so under a new

command you need to add the following:

- To make the fuel go down we will use the 

block. What number do you need to type in to make the fuel go down?

- We need the fuel to continually go down, so you will need to put 

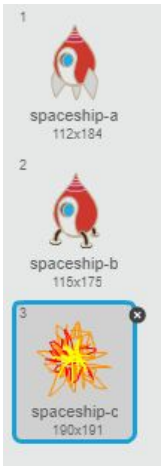
inside a repeat loop.There are two repeat blocks, which shall we use?

- The Fuel to stop going down when it lands on / touches Mars as the engines would be off

- Try this! When you launch the spaceship by pressing the space key, does the Fuel go down?

- You should notice the Fuel will keep decreasing and become negative! **This is not possible!**

- We have already stopped the fuel going down when it lands on mars but it must also stop going down when it runs out, this is when the 

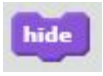


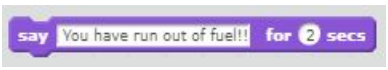
- **TRY THIS!**
  - **Now test your code out! Does the fuel variable stop going down when …**
    - **The spaceship lands on Mars?**
    - **The fuel reaches 0 and has run out?**

- However when you run out of fuel you will not just keep flying! Your mission will be over! To show this we need to create a new costume on the 'Spaceship' sprite like so:

- Now **if** the 'Fuel' variable runs out 

  - we want the costume to switch to the explosion costume 

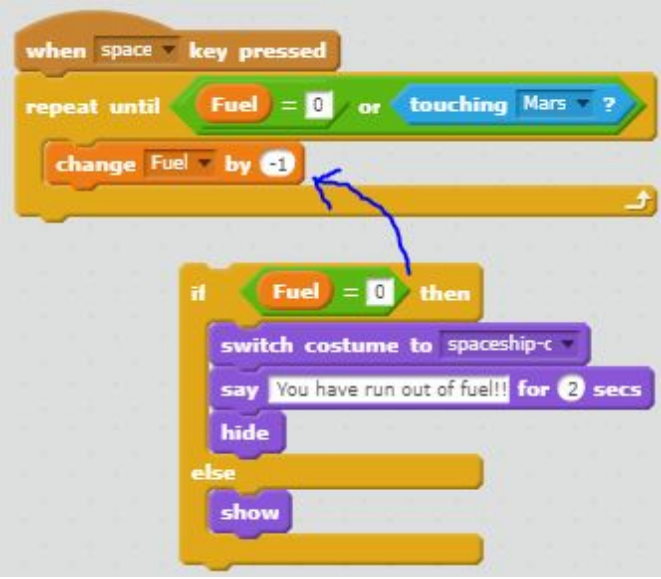
  - The spaceship to 

  - And the spaceship to say a message that it has run out of fuel 

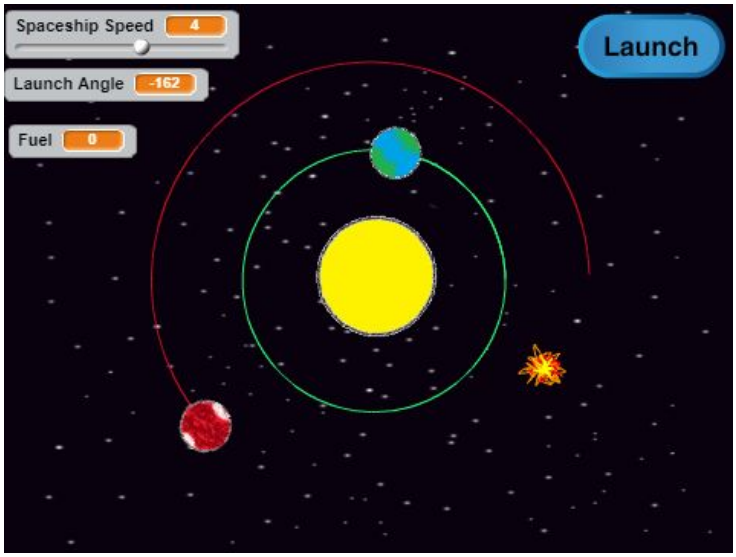
- Because we have made the spaceship hide when the fuel has run out, else when it has not run out of fuel we must make it show

- Your code should look like this:

- And you need to put it into the ‘repeat until loop’ like so:



- **TRY THIS!**
  - **Now test this out to see if this works! Does the ‘Fuel’ reduce until it hits 0 and then explode?**

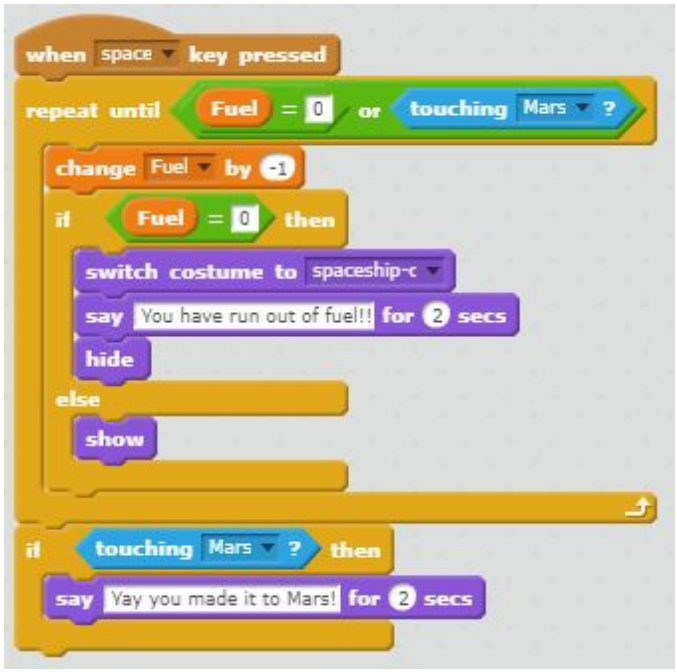


- Finally we can add a message for **if** you land on Mars by adding the following

- Place this **under** 

.

- So this is how all that code should look:



- TRY THIS!
  - Now your code is done!!
  - Test it to make sure the message comes up when you successfully land on Mars!

Challenge:

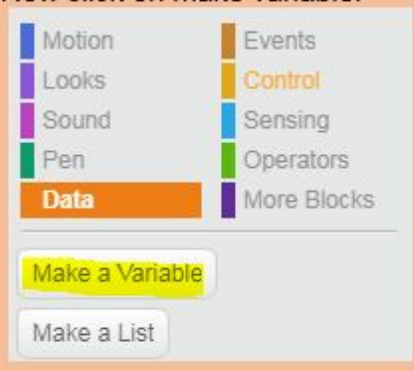
- As you can see from your game, the Earth orbits around the Sun about twice as fast as Mars orbits the Sun.
- Can you create two variables to count the number of ‘Earth Years’ and the number of ‘Mars Years’ so you can compare the two?
- The year counter should increase by one after the planet has made one full orbit and it should look something like this:

Save your project

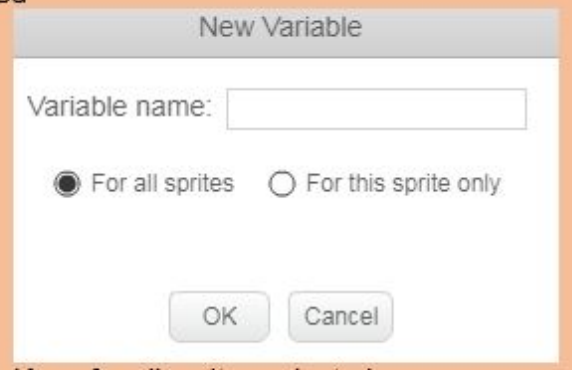
SCRATCH HELP: HOW TO CREATE A VARIABLE

1. Click on Data

2. Now click on make variable:



3. This window will appear and you can name your variable and keep for all sprites selected



Keep for all sprites selected