Scratch in Space – Coding the Solar System (Grade 6, Grade 9)

Stream of Engineering	Electrical, Software
Cost (per student)	$0
Time (preparation and activity)	1.5 hours
Complexity	Medium

Learning Goal: To use knowledge of electricity, circuits, and coding to explore space concepts and create art that allows students to interact with the solar system.

Materials
- Laptops/Computers
- Internet Connection

Mini Lesson (Theory and Background):

Show this video if you have access to internet and a projector:
https://www.youtube.com/watch?v=cKhVupvyhKk

What is coding?

Code is a set of instructions that a programmer gives to a computer to tell the computer what they want it to accomplish.

Why do we need this set of instructions?

Computers cannot think for themselves, they can only read what they are being told. We must communicate with computers to be able to make them function.

Can we tell a computer what to do out loud?

No, computers only speak very specific languages, that must be given to them through text.

What languages do some of you speak?

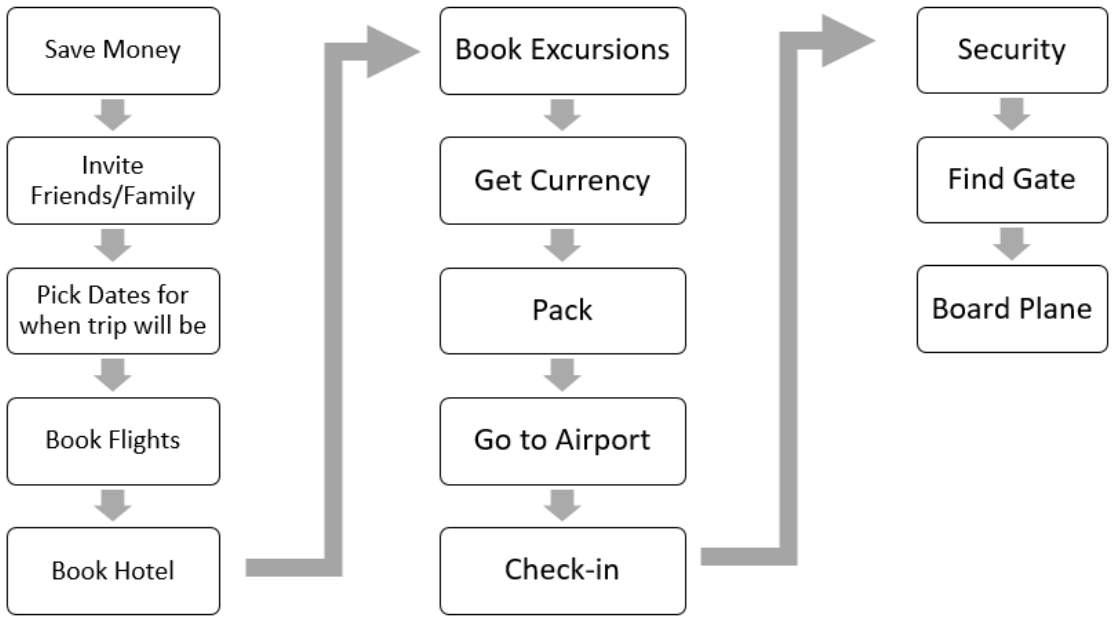
Do we think computers understand these languages? Does anyone know the names of what some of the languages computers do understand are?
- Drag and Drop
- Scratch

- Hour of Code
- Java
- C++
- C#
- Python
- HTML
- etc.

When real programmers are designing a code, before they write a single line of code, they come up with a master plan of how they are going to make their code work. This master plan is called a ***Flow Chart.***

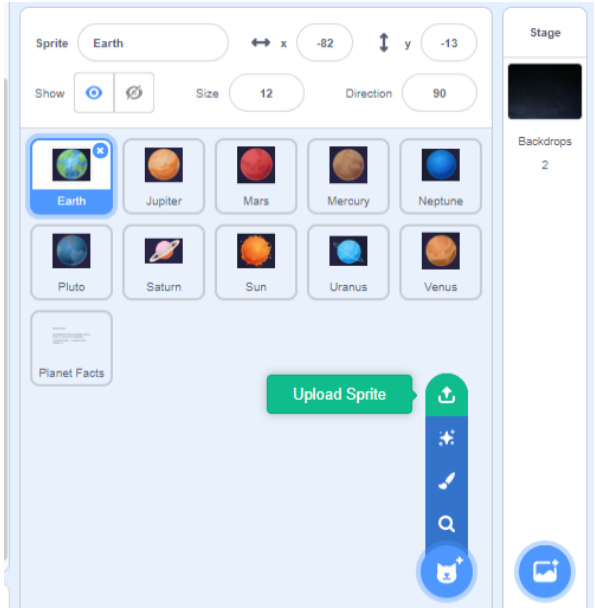
The reason flowcharts are so important in programming is because they help us see the entire picture of what we are trying to accomplish. Coding can include very small details that can sometimes be overlooked if you don't have a clear picture of what your code needs to include.

As a class we are going to make a flowchart together. We are going to write the flowchart for the process of taking a trip. Identify all steps necessary to take a trip. It should look something like this:

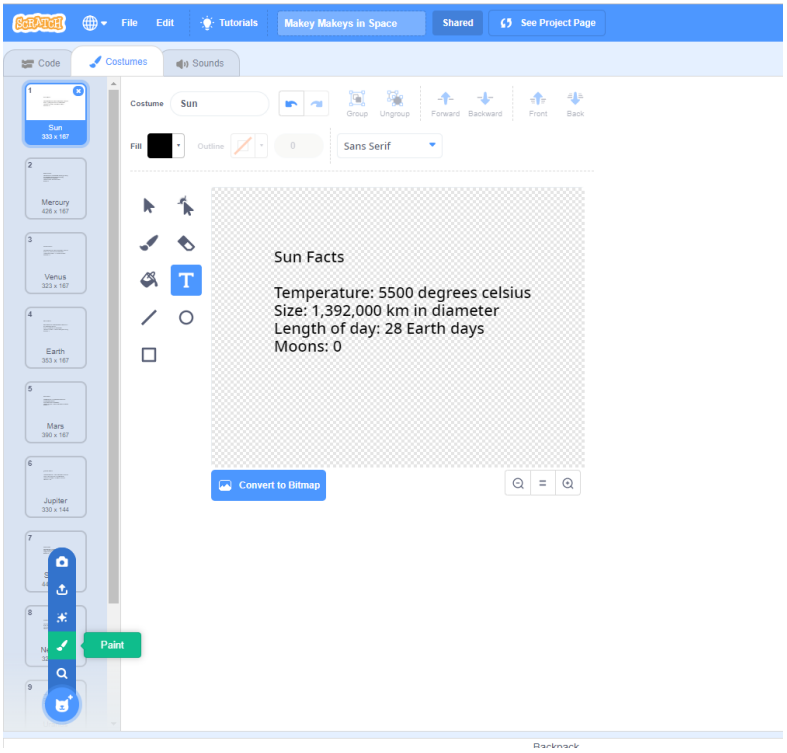


While this flowchart includes the big ideas, it is only the main ideas of this process. That's just like when we write the flowchart for a computer program. We only include the main ideas of what we need our code to do, then more detailed individual lines of code will be added later when we actually begin to write the program.

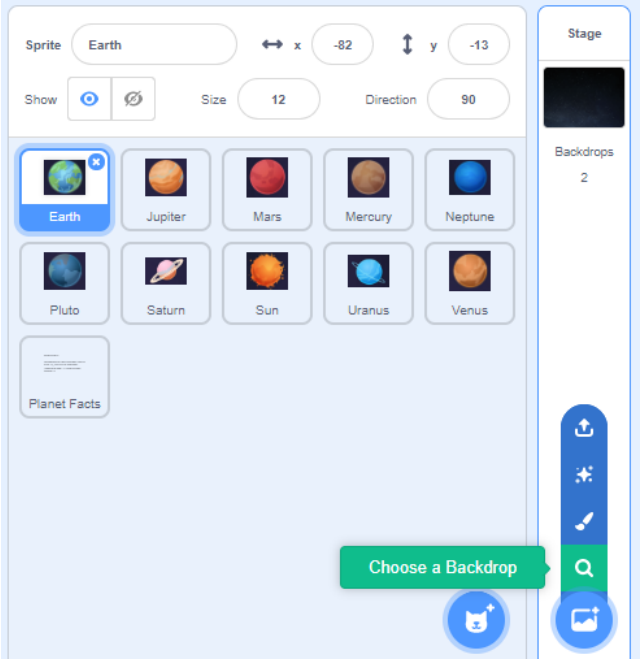
Another thing to note is that we did not necessarily need to do things in the specific order that they are given in the flowchart. There are many different ways to code a program with

the same end goal. Often we will find that 2 codes can do entirely the same thing, but look nothing alike.	
Engineering Heros!	*Elsie MacGill, known as the Queen of Hurricanes, was the first woman in the world to receive an aeronautical engineering degree and the first woman in Canada to receive a Bachelor's degree in electrical engineering. In World War II she worked as an aeronautical engineering and contributed to the success of Canadian aircraft construction. She was later a commissioner on the Royal Commission on the Status of Women in Canada and received the Order of Canada for her work there. She paved the way for women like Julie Payette, Canadian engineer and astronaut.*
Time:	Activity: Students will use Scratch to design and code the solar system **Intro:** Review with students the parts of the solar system and their order (Sun, Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus, Neptune) **Activity:** 1. Open Scratch (https://scratch.mit.edu/) and click create. First we will create all of the components that will then be coded. 2. Find and save images of planets that can be used as sprites (sprites are computer graphics, most often used in 2D games). 3. Upload all the planet images as sprites in the project. Then, use the paintbrush to create a blank sprite. This is where planet information will be written. Label the sprites. As they are added, sprites will appear in the scene. Adjust size and location as desired. 

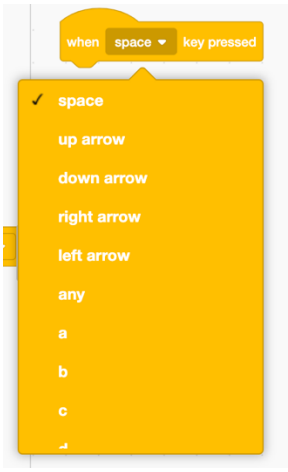
4. Click on the Planet Facts sprite. Add new costumes for each planet using the paint brush. Type desired planet info into each costume. Label the costumes.



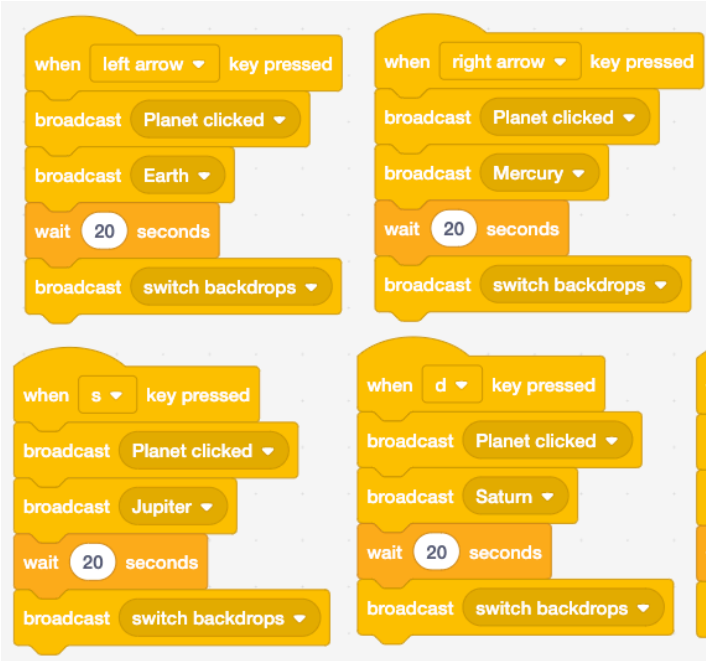
5. Add a backdrop in the bottom right hand corner, so that there are two in total. I choose a plain white background and stars. Label backdrops.



6. Now that all of the components exist, we will code each one. The goal of our code will be to have the correct Planet Facts appear when the corresponding planet is touched in the drawing attached to the Makey Makey. Note, it is important that students know which Makey Makey key is attached to which planet.
7. Click on the Earth sprite. First, use the yellow event block, “when space key pressed”, to code for keystrokes. Use the drop down arrow to change the keys. Add a block for each key (the four arrow keys, the space bar, and the a, w, s, d, f keys). This code can technically go under any sprite, but I put them all in the Earth sprite coding space to keep things organized.

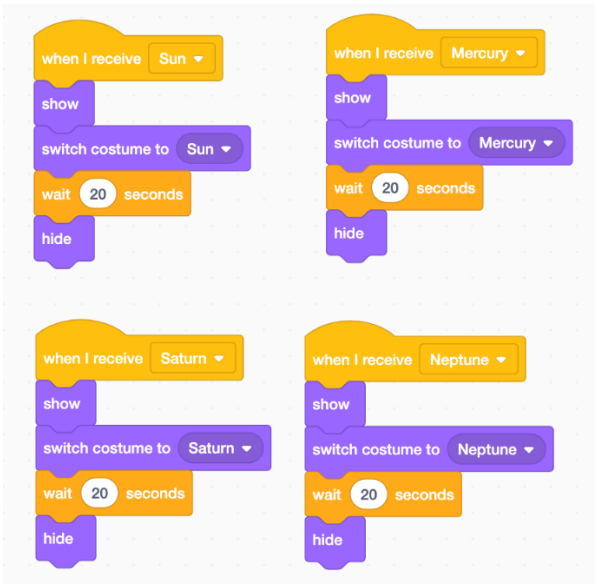


8. When a key is pressed, we want to broadcast a message. Add the “broadcast message1” block to each “when ___ key pressed” block. Use the drop down arrow to create a new message, one for each planet. Each key press should broadcast the name of a different planet. In a few steps, we will code the Planet Facts sprite to be the receiver of this message.
9. Add a second “broadcast message1” block to each “when ___ key pressed.” Add a new message to read “Planet clicked.” Each keystroke should broadcast the same “Planet clicked” message. This message will be received by the backdrops and the planet sprites.
10. Add the orange control block “wait 1 second” to each “when ___ key pressed” block of code. Change the number of seconds to 20 (or long enough to read the info in the Planet Facts sprite).
11. Under each orange control block, add another “broadcast message1” block, entitled “switch backdrops.”



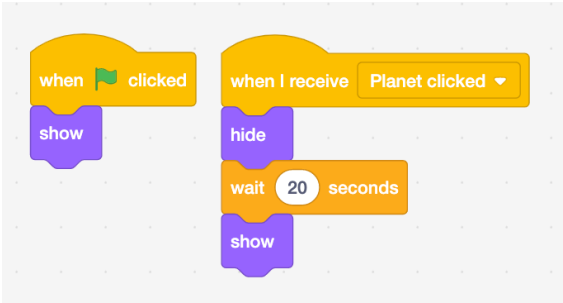
There should be a block of code like this for each planet.

12. Now we will code the Planet Facts sprite to receive these messages. Click on the Planets Facts sprite and add a yellow “when I receive message1” block for each planet. Use the drop down arrow to change ‘message1’ to the name of planets we created in the previous steps.
13. To each one, add the purple “show” looks block. This block will make the sprite appear. Add the purple “switch costume to costume1” block. Use the drop down arrow to select the costume that corresponds with the received message. E.g., if the message received is ‘Sun’, the costume should switch to the Sun costume. Add the orange control block “wait 1 second.” Change the number of seconds to 20 (to match the number selected in step 12). Lastly, add the purple “hide” looks block. This will hide the planet facts after 20 seconds has passed.



There should be a block of code like this for every planet.

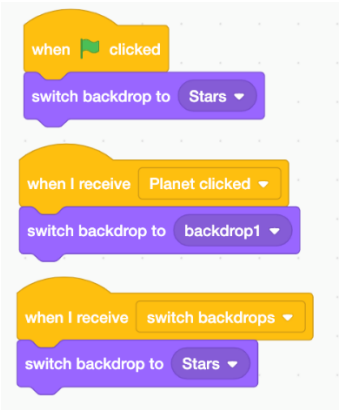
14. Add to the Planet Facts sprite the yellow code block “when [green flag] clicked” and the purple “hide” block. This will ensure that when the code begins, the planet facts are hidden and only planets are visible.
15. Next, we need to code each planet sprite to receive the “Planet Clicked” message. Click on the planet sprites one by one and add the following blocks of code:
16. Add the yellow code block “when [green flag] clicked”, followed by a purple “show” block. This will ensure each planet is visible when the code starts running.
17. Add the yellow code block “when message1 received.” Change the message to the “planet clicked” message created in the previous step. Add the purple “hide” block. Add the orange “wait 1 second” block. Change the number of seconds to 20 (to match step 12 and 15). Add the purple “show” block. This will hide all the planet sprites when the “Planet Facts” sprite is visible.



Each planet sprite needs these two blocks of code.

18. Next, we need to code the backdrops. Add the yellow code block “when [green flag] clicked”, followed by a purple “switch backdrop to ___”

block. Select the star background. This will ensure the star background is visible when the code starts running. Add the yellow code block “when message1 received.” Change the message to the “planet clicked” message. Add the purple “switch backdrop to ___” block. Select the blank background. Add another yellow code block “when message1 received.” Change the message to “switch background” message. Add the purple “switch backdrop to ___” block. Select the star background.



This block of code ensures that the white background is visible behind the Planet Facts and the star background is visible behind the planet sprites.

19. The code is now ready to use!

Debrief:

How could we change the code to produce to same scene?

- Rather than sending and receiving messages, we could code each piece to respond to keystrokes.

What else could we use Scratch to learn about?

- Basically anything! We could code parts of the body, or different animal habitats, or the layers of the Earth. The possibilities are endless.

Accommodations: If time is an issue, have students code all of the pieces first before writing in all of the planet facts. Substitute instead just the names of the planets during the lesson. The costumes for the Planet Facts sprite can always be edited later.

Extensions: If there is time remaining, have students code a back button, rather than having wait times when planets are clicked. Another sprite will have to be added for the back button and sections of code that have the orange “wait 20 seconds” block will need to be altered.

	Turn the activity into a research project where students have to spend time research each planet to include fun facts in their coded solar system.
Notes to Self	