Enriched Module on Coding Education for Upper Primary Level

Technology Education Section
Curriculum Support Division
Education Bureau
June 2023

Primary Four 4
Student Version
Booklet 1

Preface

The Education Bureau actively promotes innovation and technology (I&T) education for all students. Continuous incorporation of I&T learning elements into both the primary and secondary curricula helps strengthen the cultivation of students' interest in and capability of learning information technology and I&T from an early age, equip students with 21st century skills, and unleash their creativity and potential.

To enhance I&T education, the Education Bureau has launched the "Enriched Module on Coding Education for Upper Primary Level" for schools to adopt. Designed in accordance with the revised "Computational Thinking - Coding Education: Supplement to the Primary Curriculum" published in 2020, the curriculum module helps teachers integrate I&T elements into classroom learning more systematically. Schools should conduct appropriate curriculum planning with reference to the content of the "Enriched Module on Coding Education for Upper Primary Level", and incorporate 10 to 14 hours of enriched coding education for all upper primary students every year in order to further develop their computational thinking and strengthen their I&T learning.

The "Enriched Module on Coding Education for Upper Primary Level" is adapted from learning and teaching resources of the "CoolThink@JC" project initiated and funded by The Hong Kong Jockey Club Charities Trust and co-created by The Education University of Hong Kong, Massachusetts Institute of Technology, and City University of Hong Kong. The Education Bureau is grateful for the collaboration with The Hong Kong Jockey Club Charities Trust in consolidating and drawing on the experience accumulated by the schools in the project to develop the "Enriched Module on Coding Education for Upper Primary Level" for adoption by all publicly-funded schools in Hong Kong. The Technology Education Section, Curriculum Support Division of the Education Bureau and Department of Mathematics and Information Technology of The Education University of Hong Kong co-developed the curriculum module based on the deliverables produced and experience gained in the project. Views on the content of the curriculum module were collected from the Committee on Technology Education of Curriculum Development Council and their support was sought.

The "Enriched Module on Coding Education for Upper Primary Level" covers basic coding and computational thinking concepts, namely abstraction, algorithm and automation, as well as connection with physical objects, the use of sensors and actuators to interact with the environment, etc., allowing students to develop their computational thinking as well as interest in and ability to learn I&T through the learning of coding.

This Primary 4 curriculum module, the first of three to be developed for upper primary levels (Primary 5 and 6 forthcoming), focuses on establishing a solid foundation for students' in the above basic concepts of coding and computational thinking; through

coding activities, logical thinking and problem solving skills are developed, and computational thinking is cultivated. There are a total of 8 units in the curriculum module, including 6 core units, and 2 optional extension units for schools to provide opportunities for students with a higher ability or strong interest in coding to enrich their learning and deepen their understanding of coding and innovative technology. The curriculum module also includes a project-based component that allows students to apply their computational thinking and creativity, and make good use of programming and innovative technology in different contexts, thereby formulating solutions to everyday problems for the benefit of society.

The recommended lesson time of the curriculum module (excluding the extension units) for each upper primary year level is 14 hours. Please refer to Table 1 and the Appendix for the arrangement of this Primary 4 curriculum module, the recommended lesson time, as well as the pedagogy to be adopted.

Table 1: Arrangement of the Primary 4 curriculum module and recommended lesson time

		Core Unit		**Extension Unit**	
Unit	**Unit Title**	**Recommended Lesson Time (in minutes)**	**No. of Lessons (35 minutes for each lesson)**	**Recommended Lesson Time (in minutes)**	**No. of Lessons (35 minutes for each lesson)**
1	Introducing Scratch Programming	70	2		
2	Exploring Under the Sea	70	2		
3	Storytelling	70	2		
4	Space Traveling	105	3		
5	Creating a Maze Game	140	4		
6	Creating a Maze Game with micro:bit			70	2
7	Drawing Shapes in Scratch	105	3		
8	Designing Line Pattern Art			70	2
	Final Project	280	8		
		840 (14 hours)	24	140	4

Views and suggestions on the "Enriched Module on Coding Education for Upper Primary Level" are always welcome. These may be sent to:

Chief Curriculum Development Officer (Technology Education)
Curriculum Support Division
Education Bureau
Room W101, 1/F, West Block
Kowloon Tong Education Services Centre
19 Suffolk Road, Kowloon Tong
Kowloon, Hong Kong

Fax: 2768 8664
E-mail: teched@edb.gov.hk

Appendix

Pedagogy

Teachers may make reference to the seven-step guide introduced in the Technological Pedagogical Content Knowledge (TPACK) framework for the teaching of computational thinking (CT). Technological content knowledge (TCK) refers to the knowledge of using block-based programming environments for coding. Content knowledge (CK) refers to the knowledge of CT concepts, practices, and attitudes to be taught. Pedagogical content knowledge (PCK) refers to pedagogies that do not involve the use of programming environments for teaching CK. TPACK refers to the integration of the use of technology and pedagogy to teach CK in context.

Based on the four dimensions of the TPACK framework above, teachers may adopt the seven-step guide in the instruction of each unit with a view to developing students' problem solving skills and digital creativity. The last three steps emphasise applying TCK to exploring the possible use of tools in the programming environments for the cultivation of digital creativity; revisiting and reviewing CK for consolidation; and reflection on PCK to engage in the improvement of teaching practices relevant to CK (Kong, Lai & Sun,2020; Kong & Lai, 2022; Kong, Lai & Li, 2023).

Step 1: TCK (Introducing features of the programming environment in a specific context)

Step 2: CK (Introducing computational thinking concepts, practices and attitudes to be taught)

Step 3: PCK (Adopting pedagogy such as allowing pre-coding access to games or apps to pave the way for reflection on the design of games or apps; and engaging in unplugged activities to enhance understanding of more difficult coding-related concepts, practices and attitudes)

Step 4: TPACK (Applying knowledge of using programming environments for teaching CK with appropriate pedagogy in a specific context)

Step 5: TCK (Encouraging students to suggest applications of relevant features of the programming environment in other contexts, thereby inspiring their digital creativity)

Step 6: CK (Helping students reflect on CT concepts, practices and attitudes to consolidate their learning)

Step 7: PCK (Conducting self-reflection on the pedagogy adopted in the unit with a view to improve the next round of teaching)

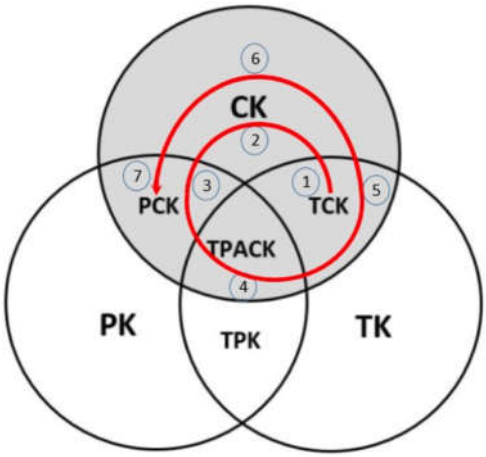


Figure 1 The seven steps in the shaded areas (CK, TCK, PCK, and TPACK) indicate those steps needed for teachers to teach content knowledge of CT. (Kong, Lai & Sun, 2020)

References

Education Bureau. (2020). *Computational Thinking - Coding Education: Supplement to the Primary Curriculum*. Hong Kong: Author.

Kong, S. C., & Lai, M. (2022). A proposed computational thinking teacher development framework for K-12 guided by the TPACK model. *Journal of Computers in Education, 9(3)*, 379-402.

Kong, S. C., Lai, M., & Sun, D. (2020). Teacher development in computational thinking: Design and learning outcomes of programming concepts, practices and pedagogy. *Computers & Education, 151*, 103872.

Kong, S. C., Lai, M., & Li, Y.G. (2023). Scaling up a teacher development programme for sustainable computational thinking education: TPACK surveys, concept tests and primary school visits. *Computers & Education, 194*, 104707.

Enriched Module on Coding Education for Upper Primary Level (Primary 4)
Editorial Board Members

Professor KONG Siu Cheung
Research Chair Professor of E-Learning & Digital Competency
Department of Mathematics and Information Technology
Director, Artificial Intelligence and Digital Competency Education Centre
Director, Centre for Learning, Teaching and Technology
The Education University of Hong Kong

Ms SIU Ka Yuk Cora
Educational Development Manager
Centre for Learning, Teaching and Technology
The Education University of Hong Kong

Ms MA Yunsi Tina
Assistant Educational Development Manager
Centre for Learning, Teaching and Technology
The Education University of Hong Kong

Miss TING Wan Yee
Assistant Educational Development Manager
Centre for Learning, Teaching and Technology
The Education University of Hong Kong

Mr LO Hoi Lam Ken
Educational Development Officer
Centre for Learning, Teaching and Technology
The Education University of Hong Kong

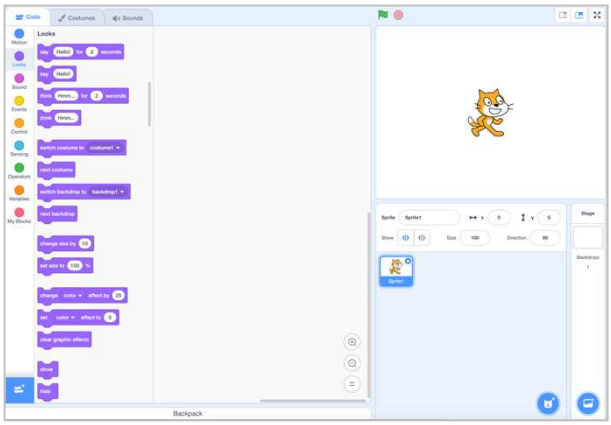
Content for Booklet 1

Unit 1: Introducing Scratch Programming Student Guide

Content

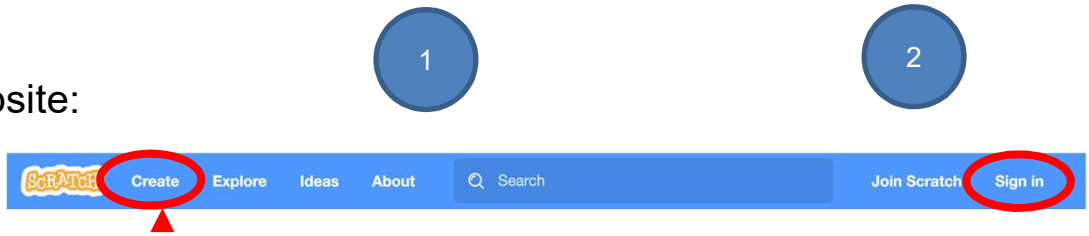
Introducing Scratch Programming

In this activity, you will learn to sign in and out of the Scratch website and learn how to make the cat move and play music .



Start Here

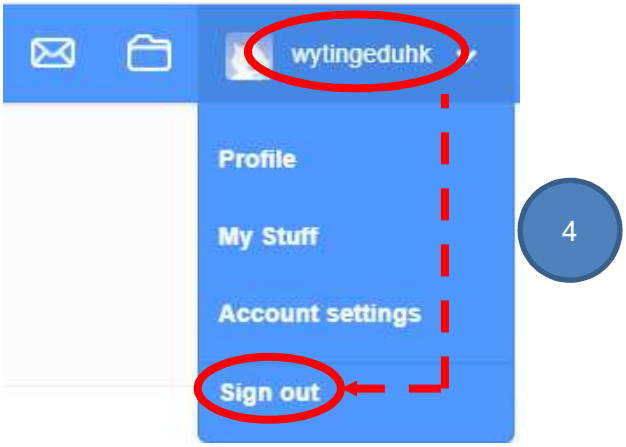
❑ Go to the Scratch website: http://scratch.mit.edu



❑ Sign into your account.

❑ Click on the **Create** tab located at the top left of the browser to start a new project.

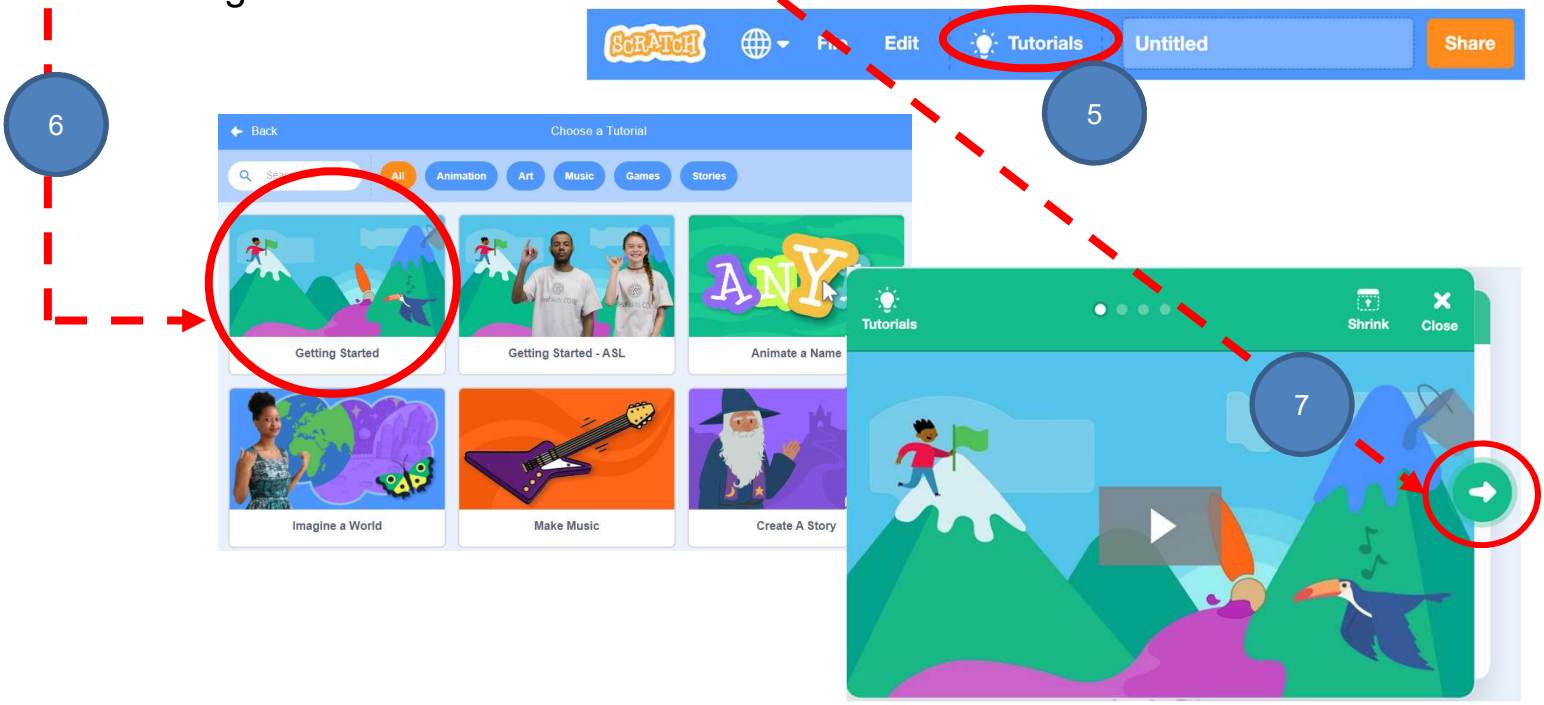
❑ Sign out by clicking on your name and clicking **Sign out**. Then sign in again!



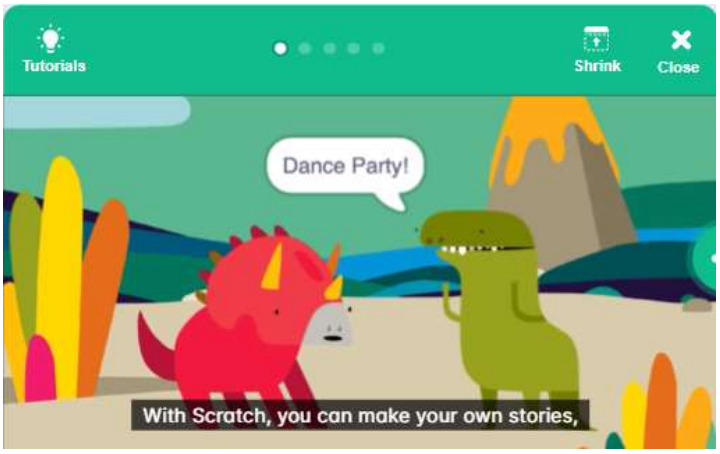
Introducing Scratch Programming

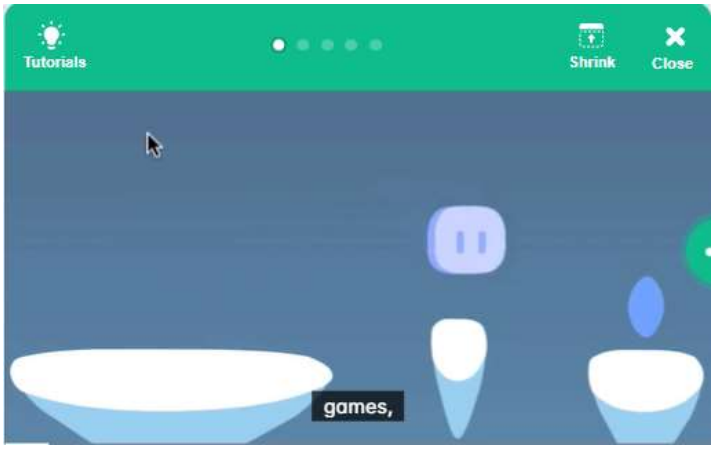
To Play

❑ Time to explore! After clicking "Create", you will see "Tutorials" on the menu bar. Click on "Tutorials" and choose "Getting Started". After watching the video, you can click the green arrow on the right side of the window to step through the tutorial.



❑ With Scratch, you can make your own stories, games and animations!

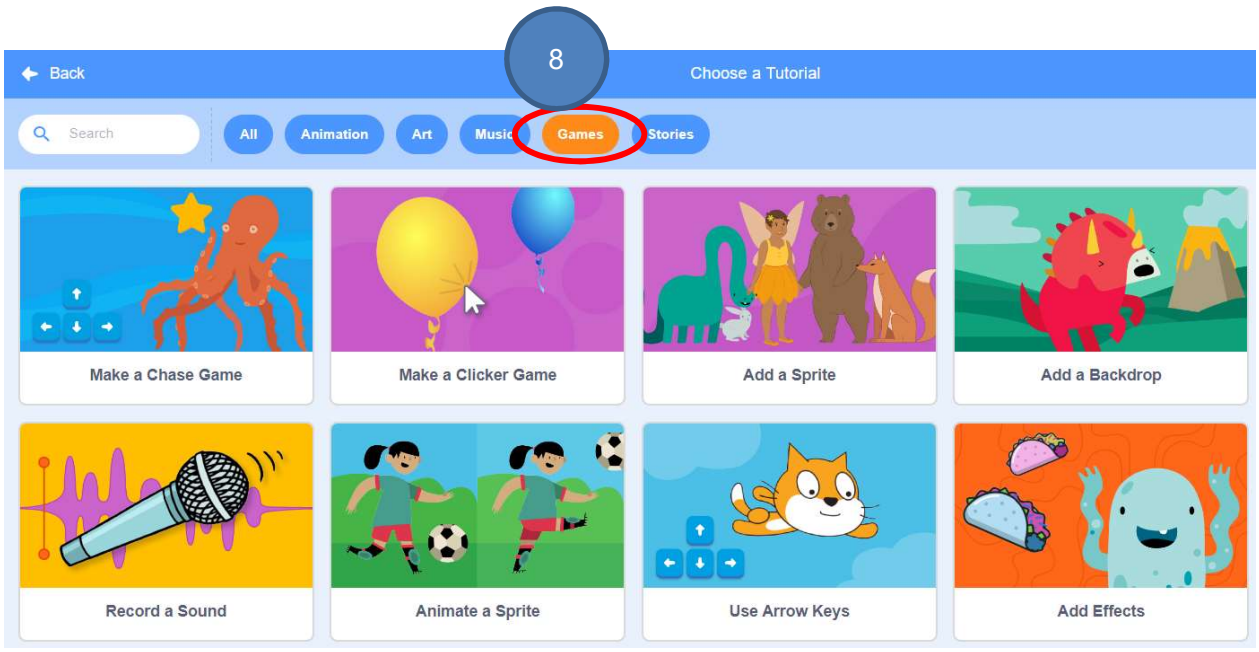




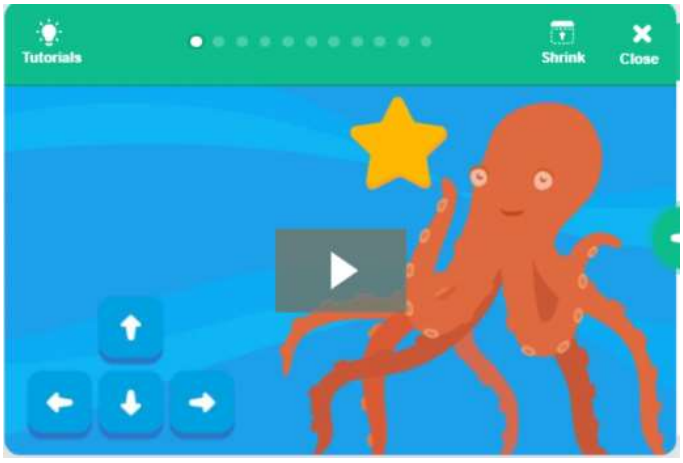
Introducing Scratch Programming

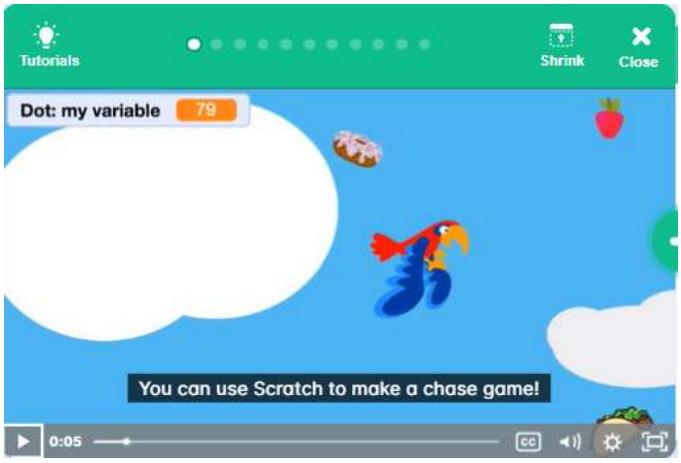
To Play

- ❑ Let's try some games! Select "Games" from the category on the top to see a list of games. Choose the games you like and try it!



- ❑ Try different games and see what can be created by using Scratch!





- ❑ Jot down the games you like and think about why you like them.

Introducing Scratch Programming

To Learn

Coding Environment in Scratch

Tutorials

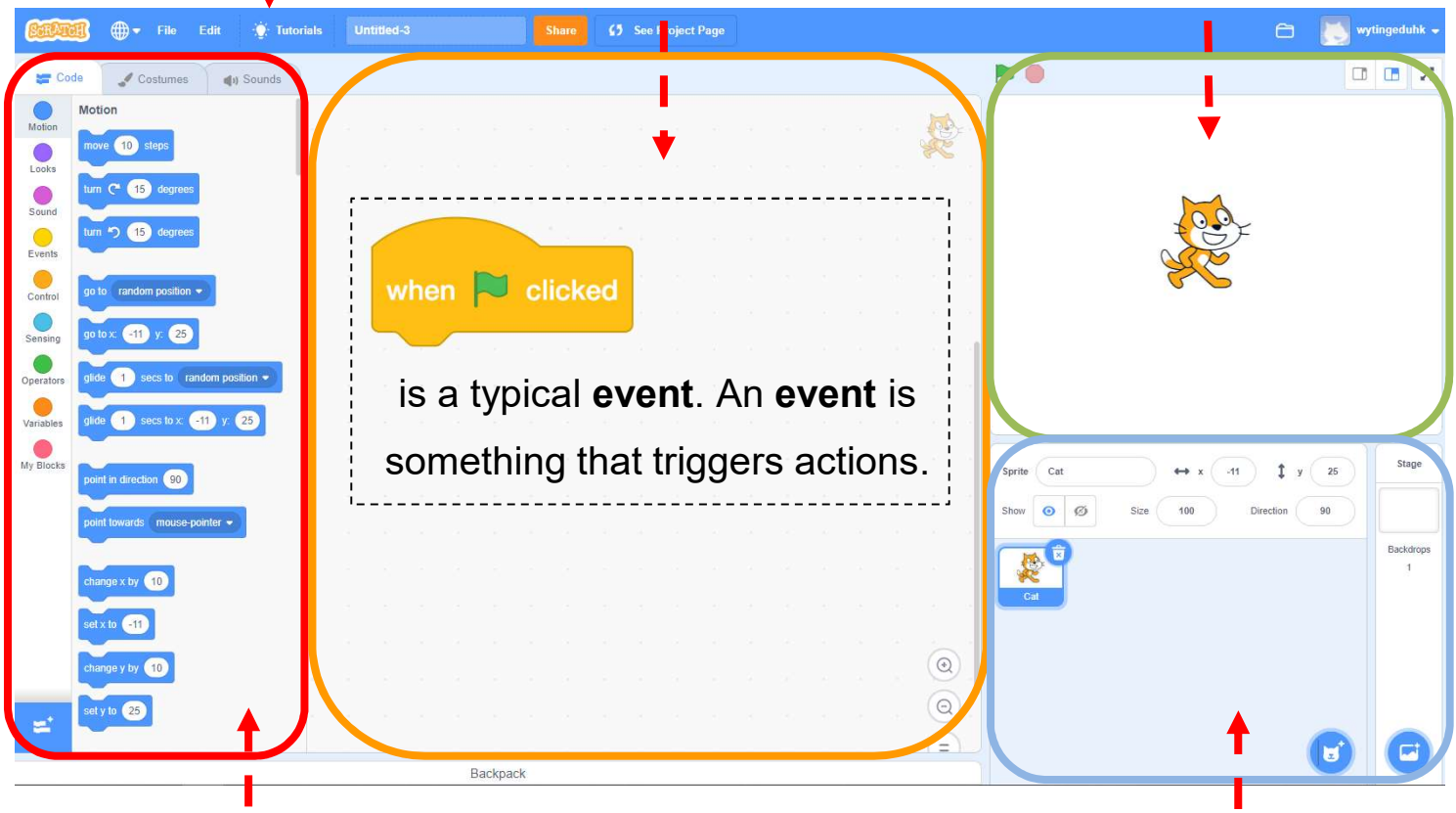
Watch tutorials to learn about Scratch.

Code Area

An area for storing blocks that run the project.

Stage

An area where the sprites are displayed and perform their actions.



Block Palette

Code: Drag and drop the coding blocks to the Code Area
Costumes: Changing sprite's costumes
Sounds: Adding sound

Sprite Pane

- Adding / Changing Sprite
- Adding / Changing Backdrop

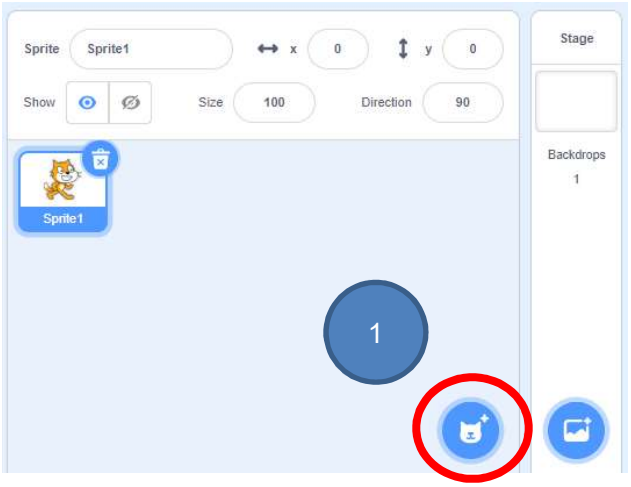
Introducing Scratch Programming

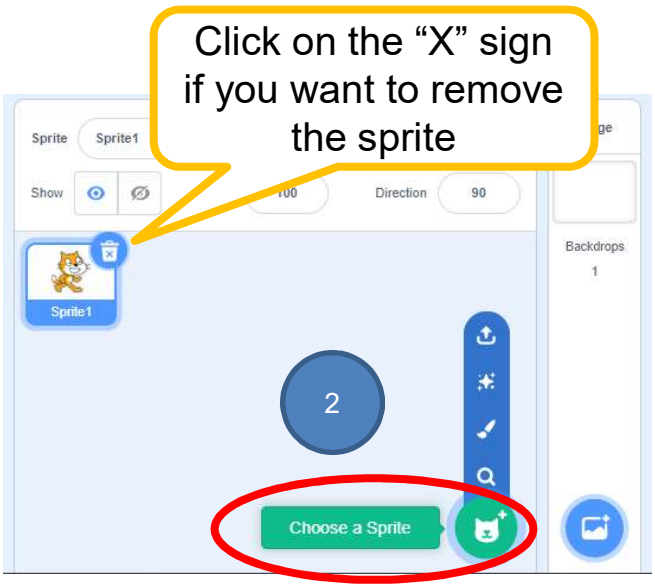
To Code

Let's try the simple tasks (1) – (6) to explore what can be done with Scratch!

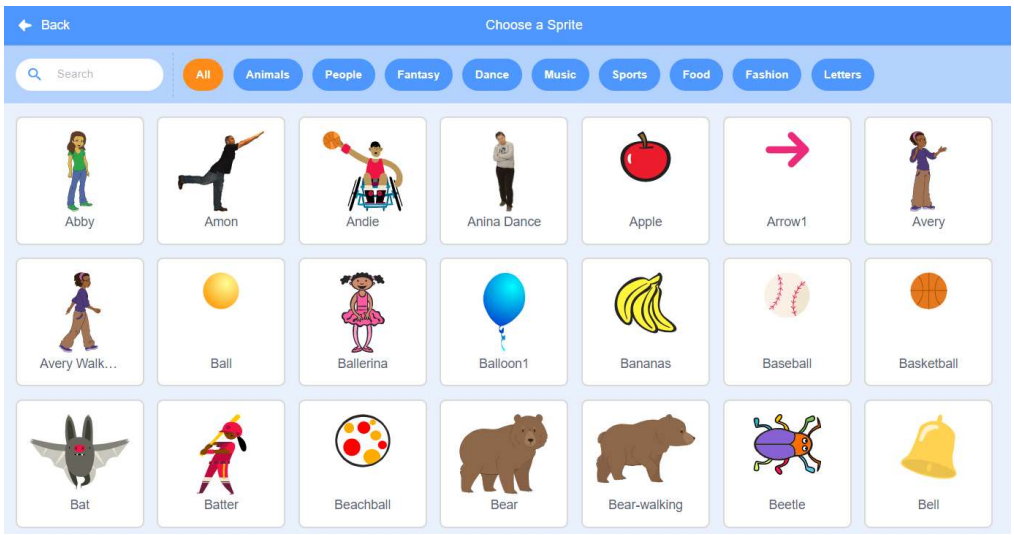
(1) Adding / Changing Sprite

- ❑ In the Sprite Pane, click "Choose a sprite".
- ❑ A set of sprites will be shown in different categories. Choose your favourite one!





3

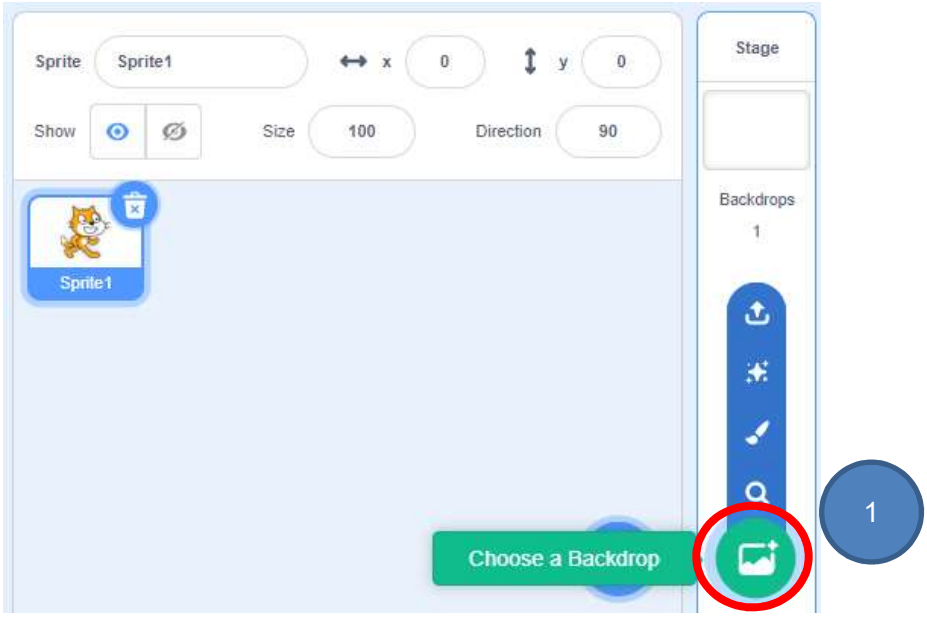


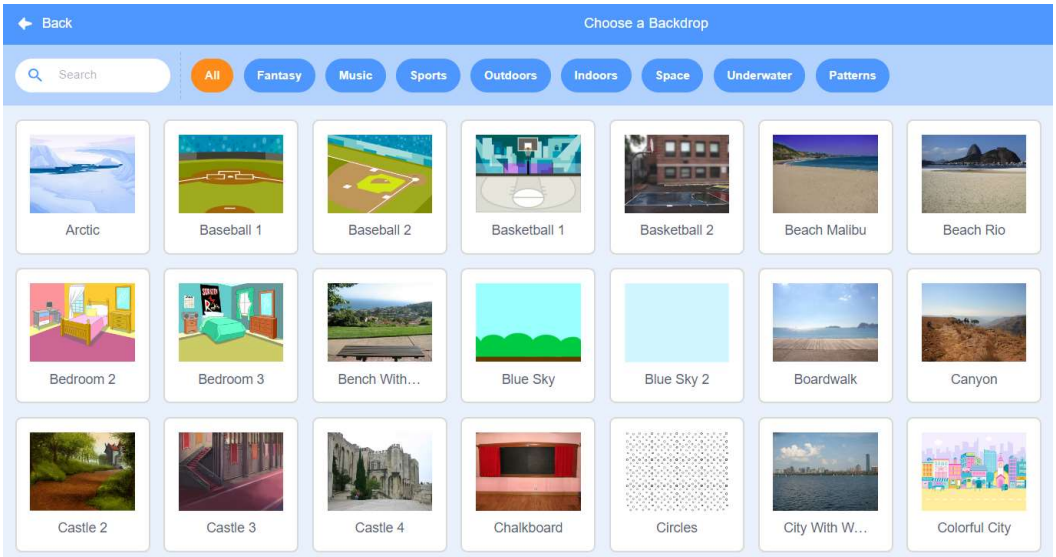
Introducing Scratch Programming

To Code

(2) Adding / Changing Background

- ❑ In the Sprite Pane, click "Choose a background".
- ❑ A set of backgrounds are now shown. Choose a good one for your project!



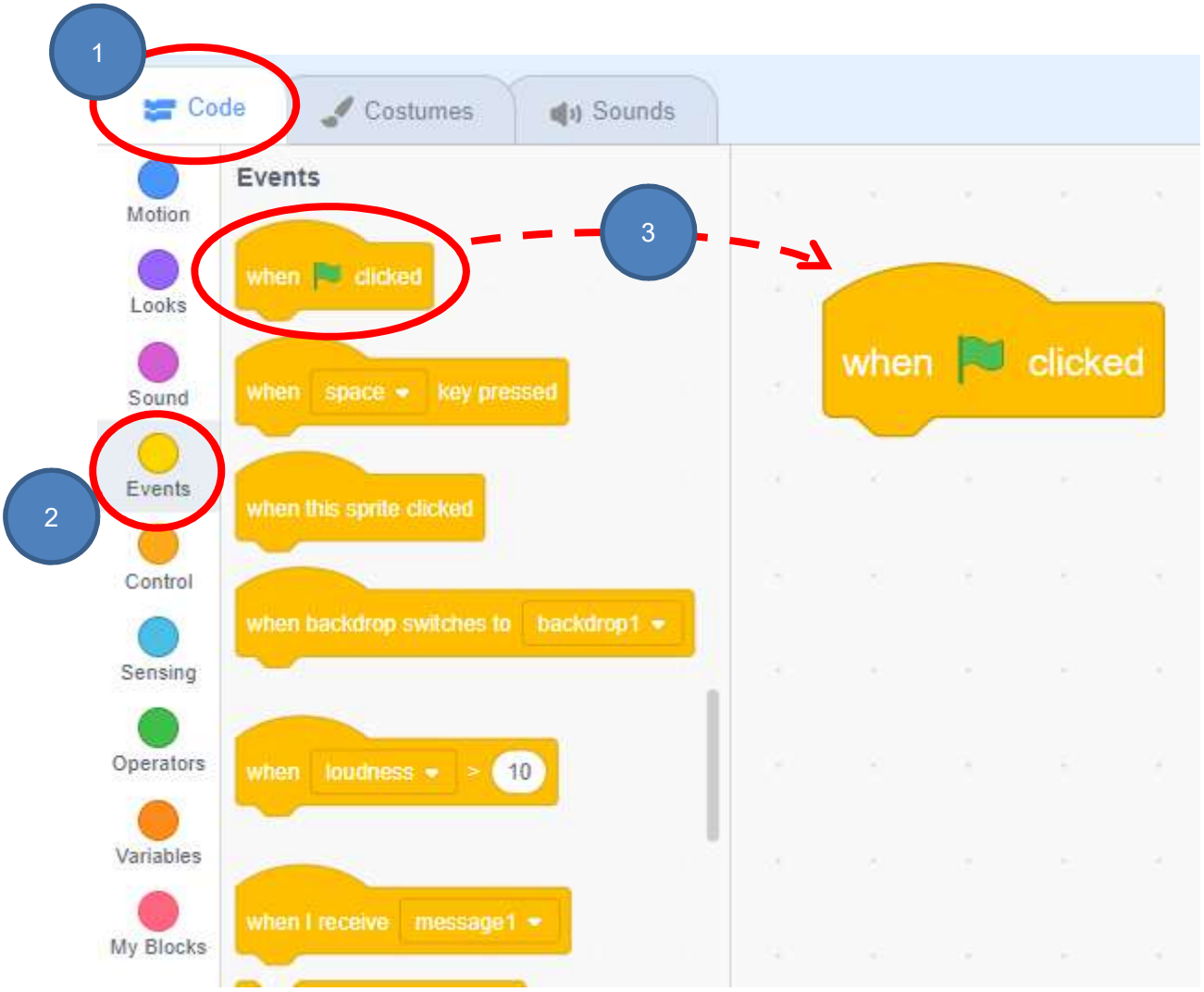


Introducing Scratch Programming

To Code

(3) Event – When Green Flag Click that trigger actions

- ❑ Let's start to code!
- ❑ In the Block Palette, click on "Code" at the top menu.
- ❑ Choose "Events" and drag the "When clicked" block to the Code area.

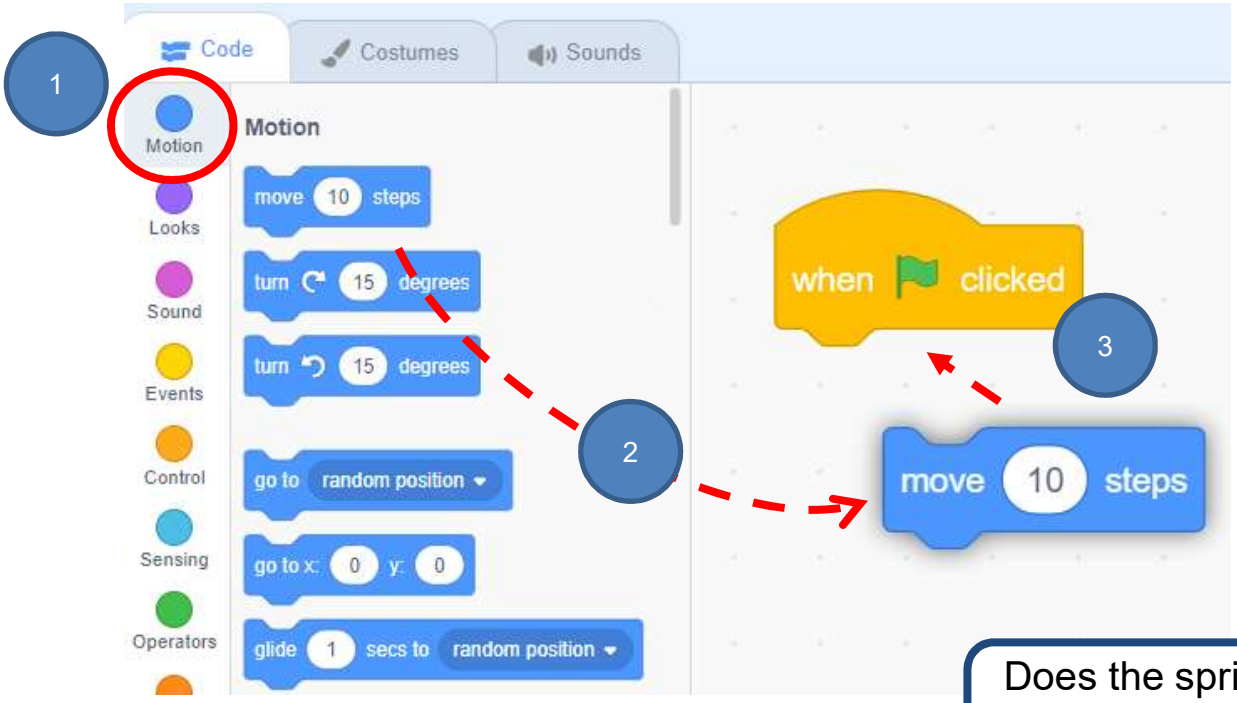


Introducing Scratch Programming

To Code

(4a) Make the Sprite Move

❑ Time to make your sprite move! Choose "Motion" and drag the "move 10 steps" block to the Code area. Snap it with "When  clicked" block.

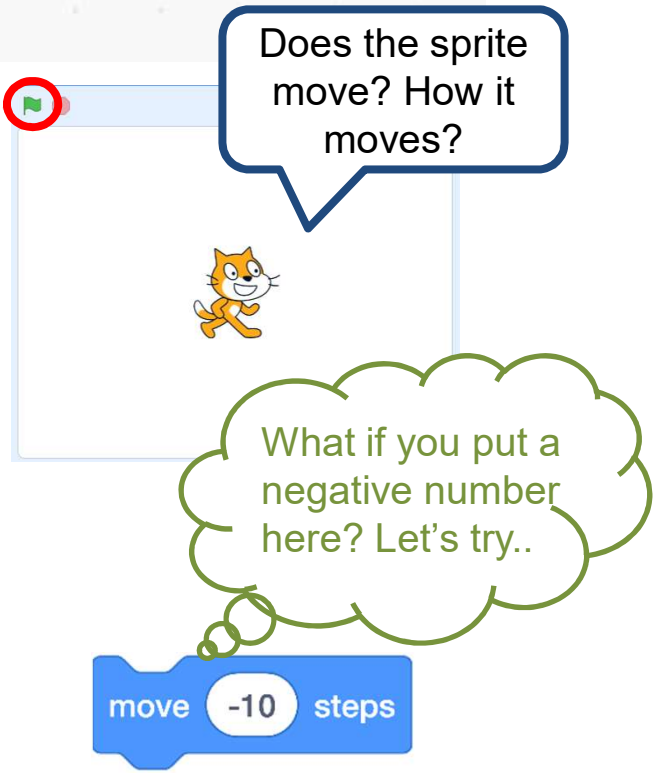


Testing and Debugging

❑ Let's test it! Click the  to test and see if it works!

Testing a computer program is the process of checking if it can produce results as designed.

Debugging a computer program is the process of finding out ways to revise the program so that the bugs can be removed.

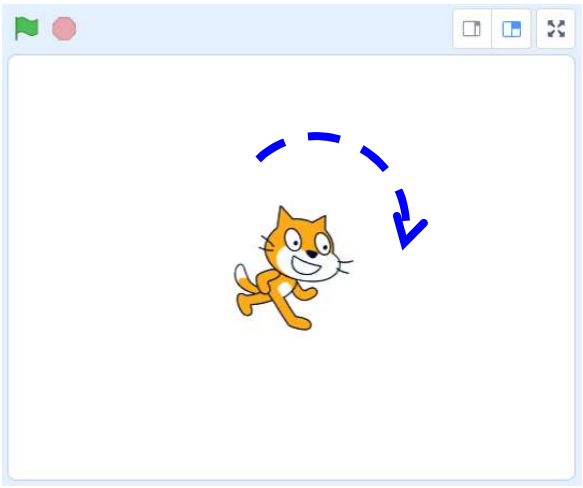


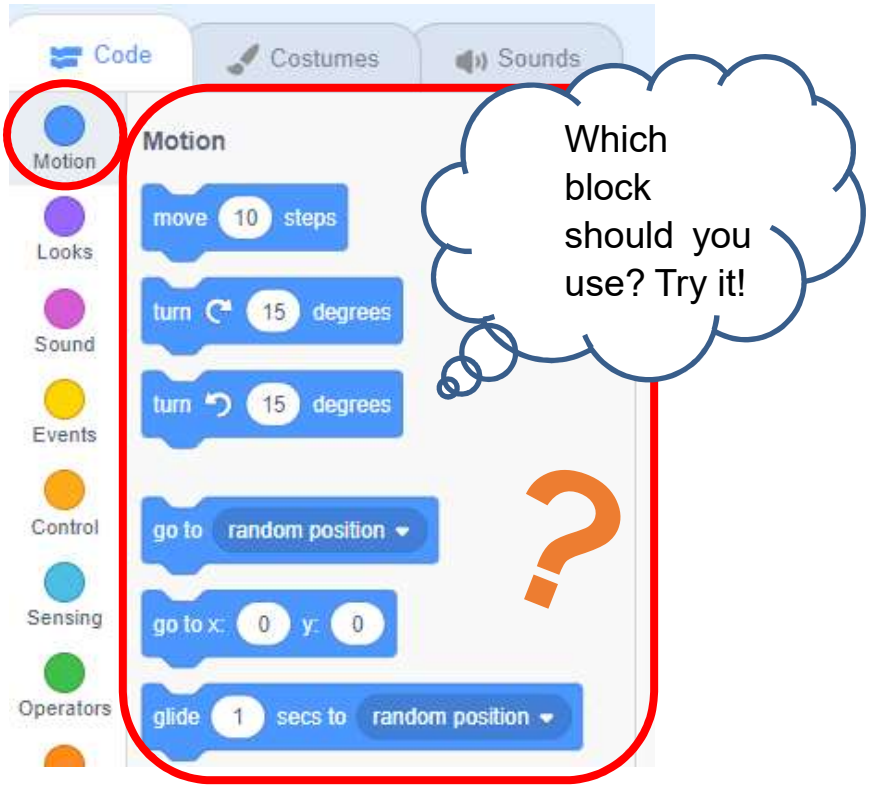
Introducing Scratch Programming

To Code

(4b) Make the Sprite Change Degree

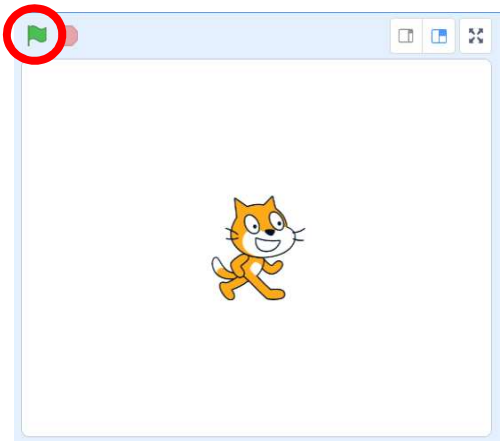
- Can you try to make the sprite turn around like this?





Testing and Debugging

- Remember, you can always click the  and see if it works!



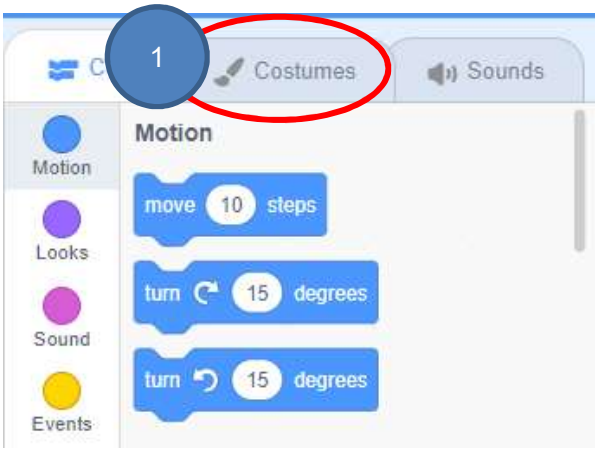
Introducing Scratch Programming

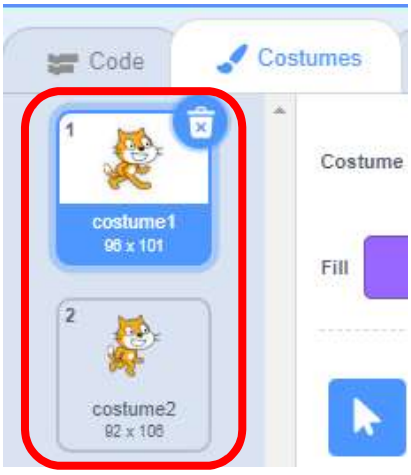
To Code

(5) Change the Sprite Costume

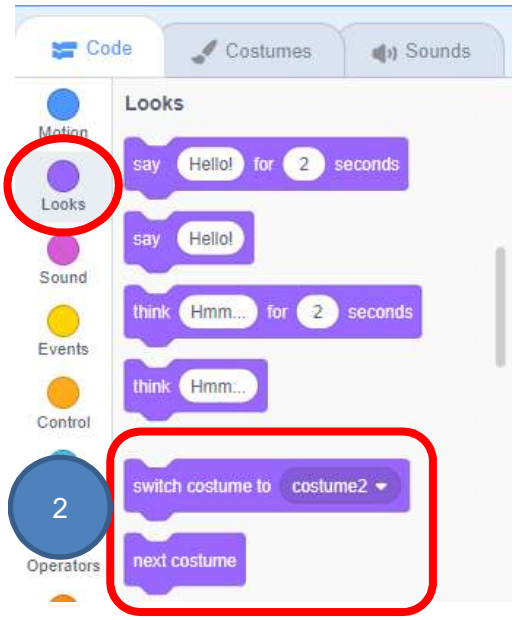
- Now let's try to change the sprite costume to make our project more interesting!

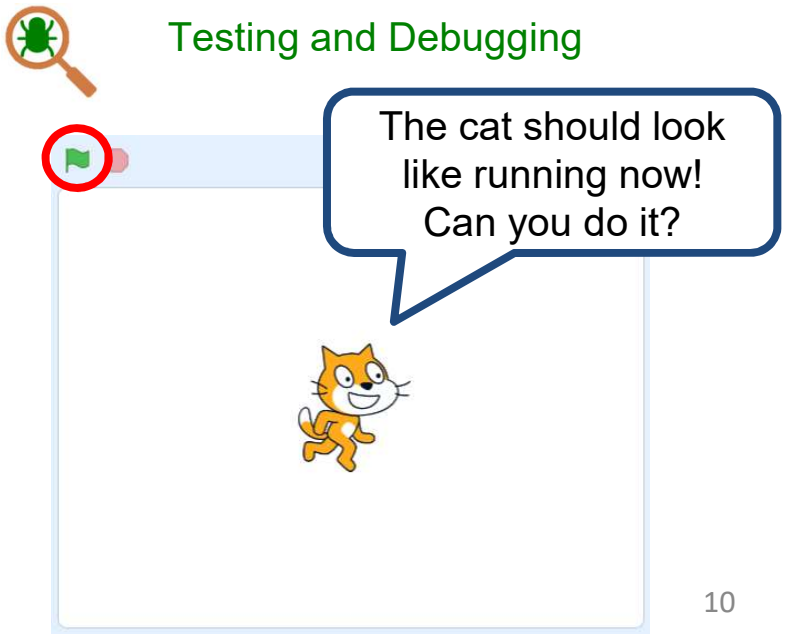
- Click on "Costumes", you will see a set of costumes of the sprite you added.





- Go back to "Code", choose "Looks" and change the sprite's costumes!
- Drag "switch costume to ___" and "next costume", snap with "when clicked" block. Click the and see what happens.



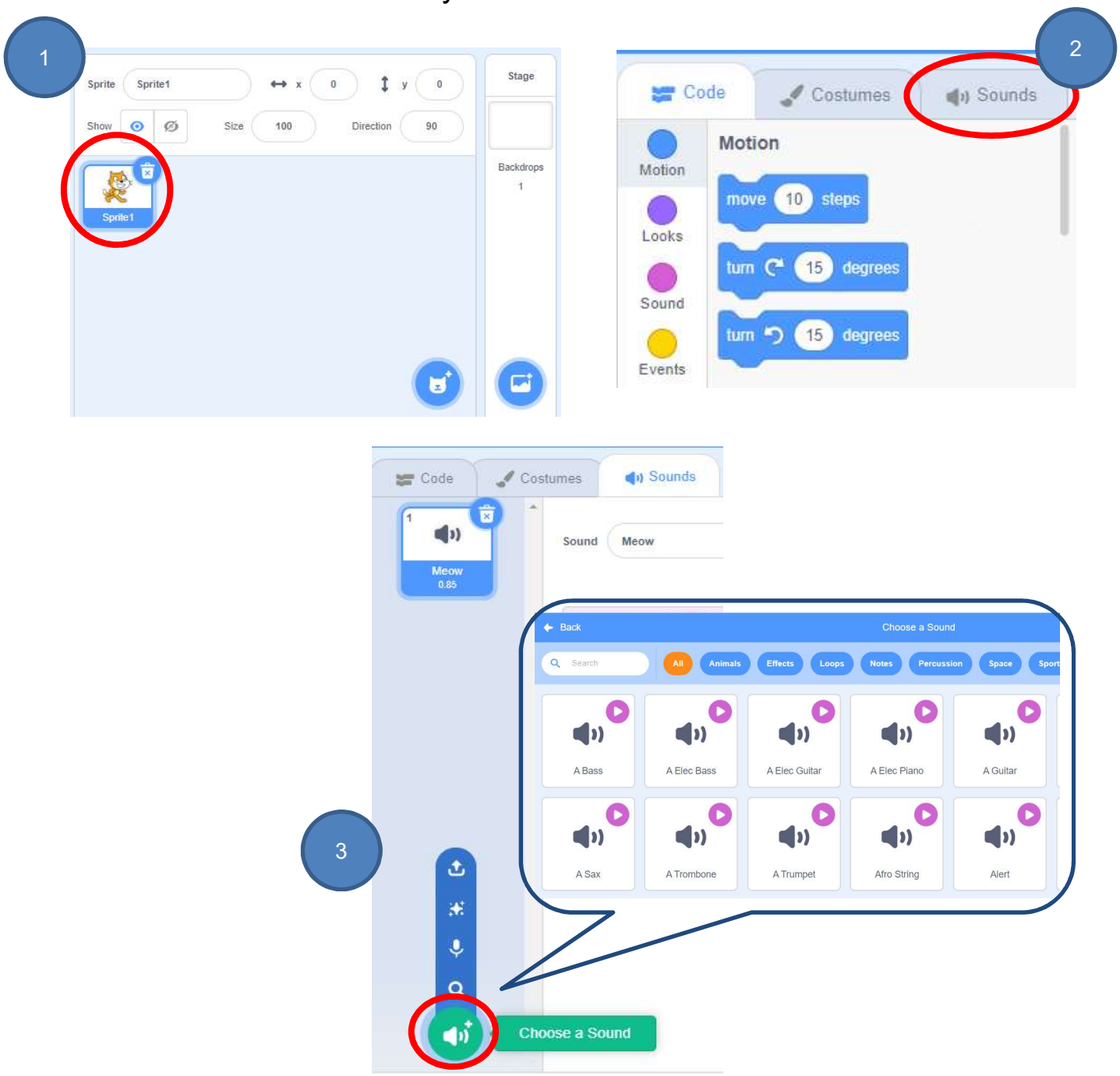


Introducing Scratch Programming

To Code

(6) Adding Sound to the Sprite

- Click on your Sprite, go to the "Sound" page, click "Choose a Sound", you will see a lot of sound that you can choose.



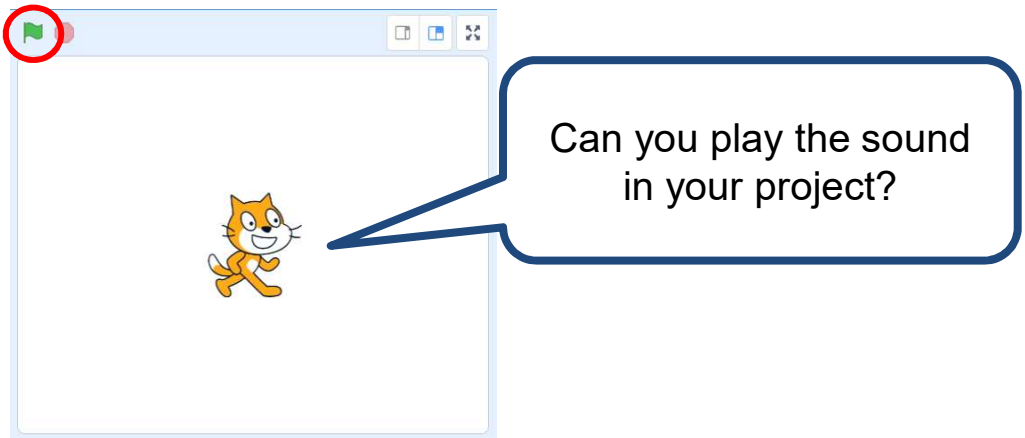
Introducing Scratch Programming

To Code

(6) Adding Sound to the Sprite

- In the "Sound" drawer, drag "play sound__until down" or "start sound ___", snap with "when clicked" block.

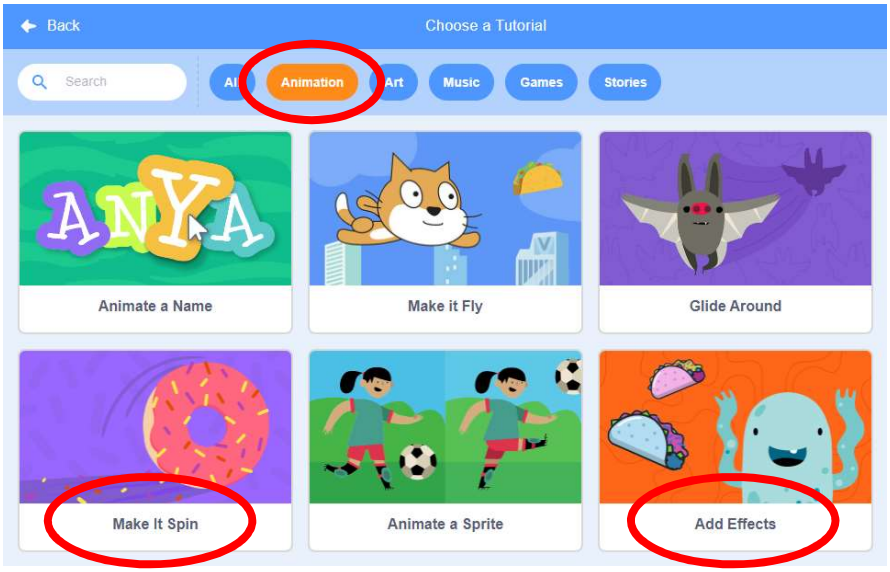
Testing and Debugging

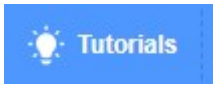


Introducing Scratch Programming

To Code

- ❑ Now you have got some basic skills for Scratch! Let's explore more.
- ❑ Click on "Tutorials" on the menu bar again and then choose the "Animation" category on the "Tutorials" page.
- ❑ Complete the "Make It Spin" and "Add Effects" tutorials.





- ❑ Change the value of turn degrees or the number of repeat times and see what happen?

Make It Spin

Add Effects

When I change the value of turn degrees, I can see…

When I change or the number of repeat times, I can see…

Introducing Scratch Programming

CAN YOU MAKE THE SCRATCH CAT DO SOMETHING INTERESTING?

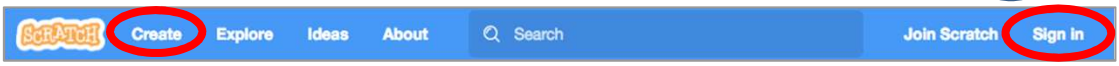
In this activity, you will create a new project with Scratch and make the cat do different actions with various costumes! What will you create?

To Create

❑ Go to the Scratch website: http://scratch.mit.edu

1

2



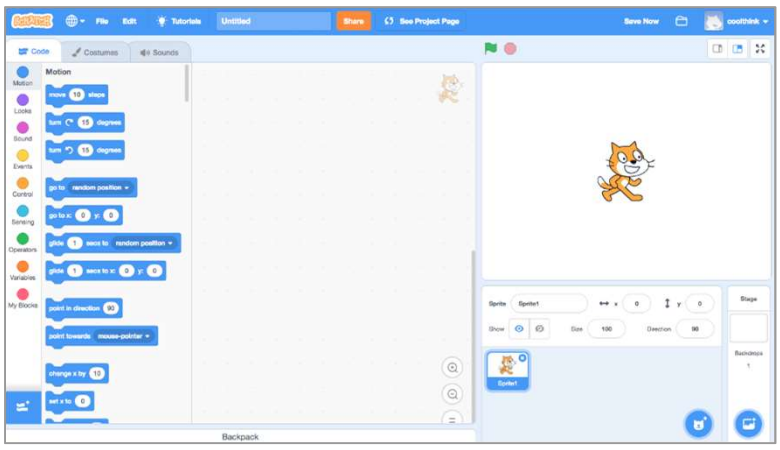
❑ Sign into your account.

3

❑ Click on the **Create** tab located at the top left of the browser to start a new project.

❑ See if you can use the right blocks to make the cat do what you want.

4



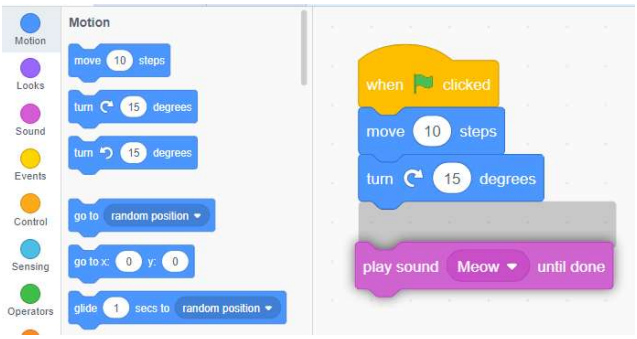
Introducing Scratch Programming

To Create

Task 1:

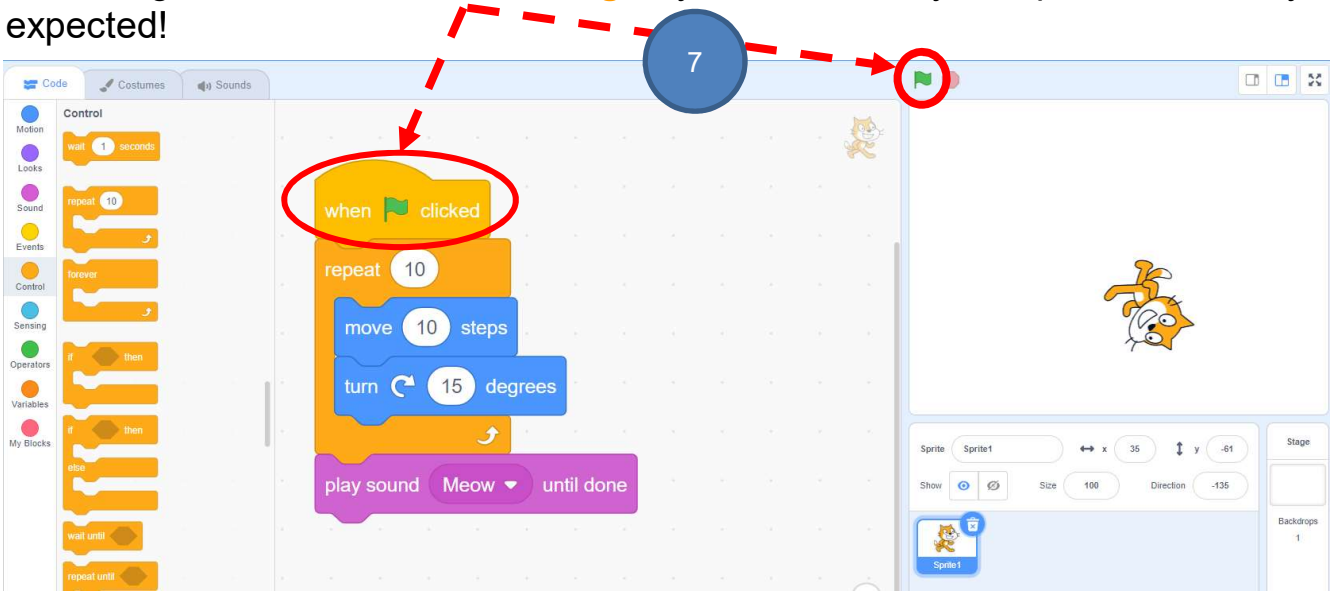
❑ Drag and drop Scratch blocks into the Code window. 5

❑ Experiment by clicking on each block to see what it does or try snapping blocks together.



6

❑ Don't forget to **Click the Green flag** anytime to test if your sprite works as you expected!



When the Green flag is clicked, does your sprite:
- ✓ Move?
- ✓ Turn around?
- ✓ Make sound?

To Think:
Do you think there is any **sequences** in the tasks (move, turn and play sound)?

Knowledge builds up: Sequences
It is a key concept in programming. It is the order in which the programming statements are executed. A wrong order would lead to incorrect programming results.

Introducing Scratch Programming

Sharing in Scratch Studio

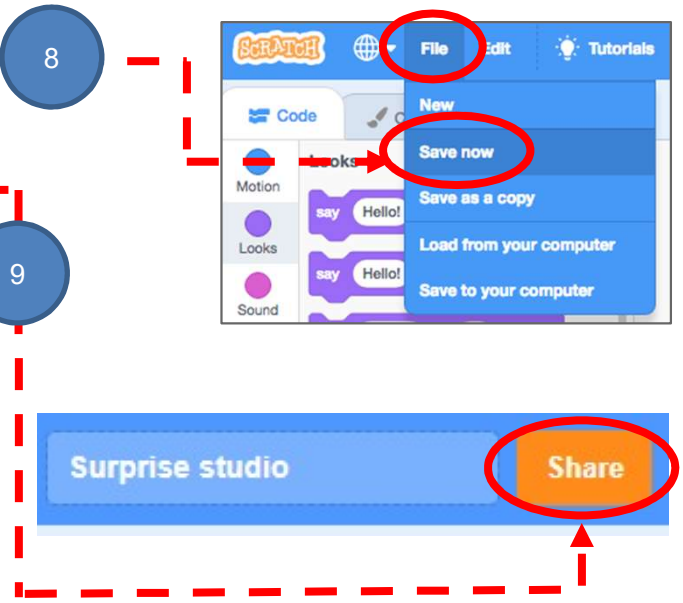
LEARN HOW TO ADD YOUR PROJECT TO AN ONLINE SCRATCH STUDIO!

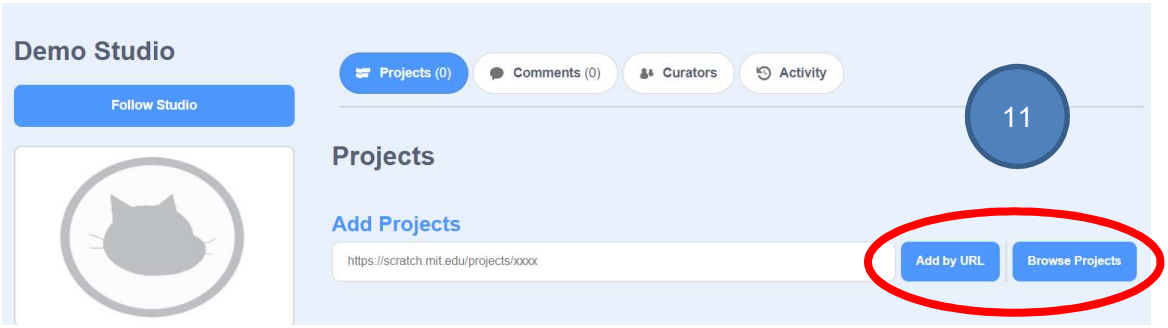
Studios are collections of Scratch projects. Follow the steps below to add your Scratch Surprise program to your class' Surprise Studio on the Scratch website.

Share to Studio

When you finish, you will add your project to your teacher's Studio:

- ❑ Save your project by clicking "Save now" under the File menu.
- ❑ Click the orange "Share" button.
- ❑ Go to your teacher's Studio (Your teacher will give you a URL).
- ❑ In the "Add projects" column, you can Add by URL or Browse Projects.





Introducing Scratch Programming

To Reflect: Two Stars and a Wish Worksheet

Name of Project: _____ Name of Creator: _____

Please write down two things that you like about this project.

1

_____

2

_____

What is one thing you would like to add or change to make this project better?

_____

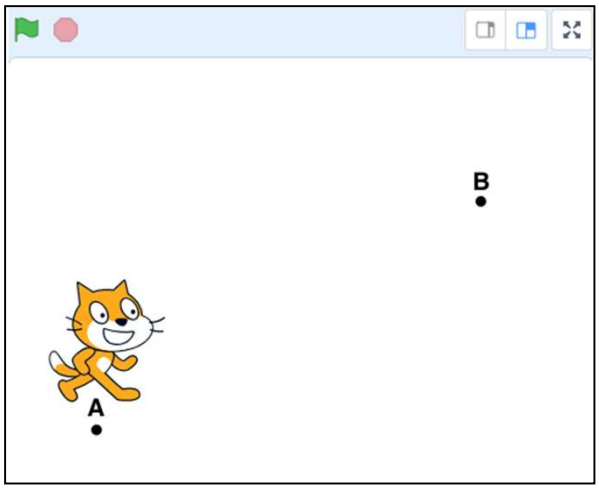
_____

_____

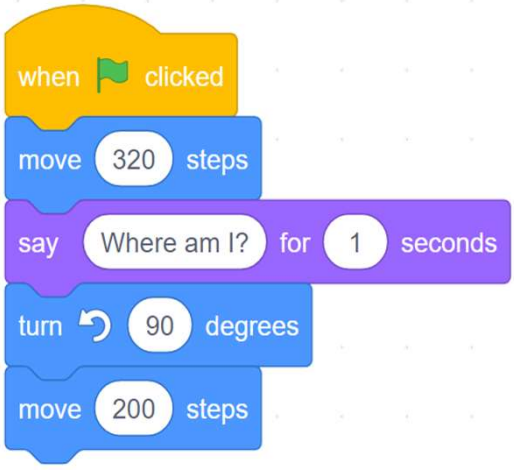
Introducing Scratch Programming

Review Questions

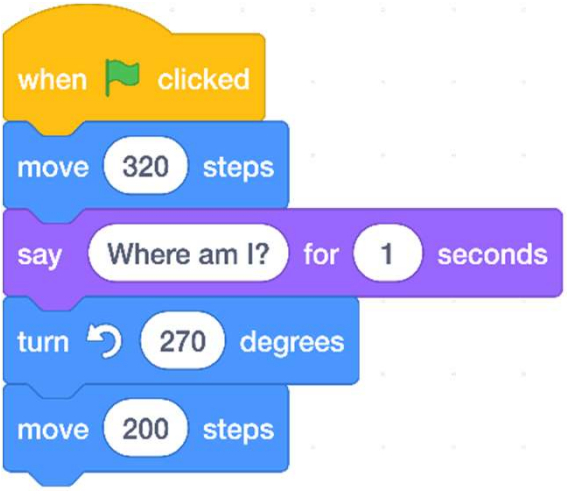
1. Which of the following sequences of commands (sets of blocks) will make the Scratch cat move from point A to point B on the stage?



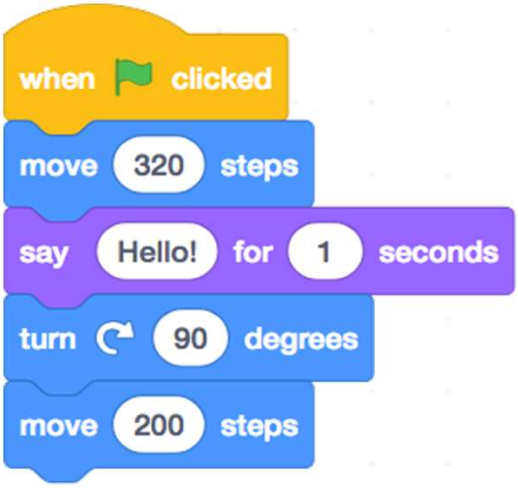
A.



B.



C.



D.

Introducing Scratch Programming

Review Questions

2. What happens when you put a negative number in the move block?

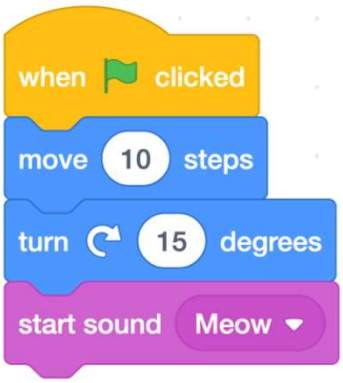
A. The cat moves down on the screen.
B. The cat moves forward (to the right) and backward (to the left).
C. The cat turns around.
D. The cat moves backward (to the left) on the screen.

Introducing Scratch Programming

Revision on Key Concepts & Practices

Events: We use event blocks to trigger Scratch to take actions.

Sequences: It is a key concept in programming. It is the order in which the programming statements are executed. A wrong order would lead to incorrect programming results.

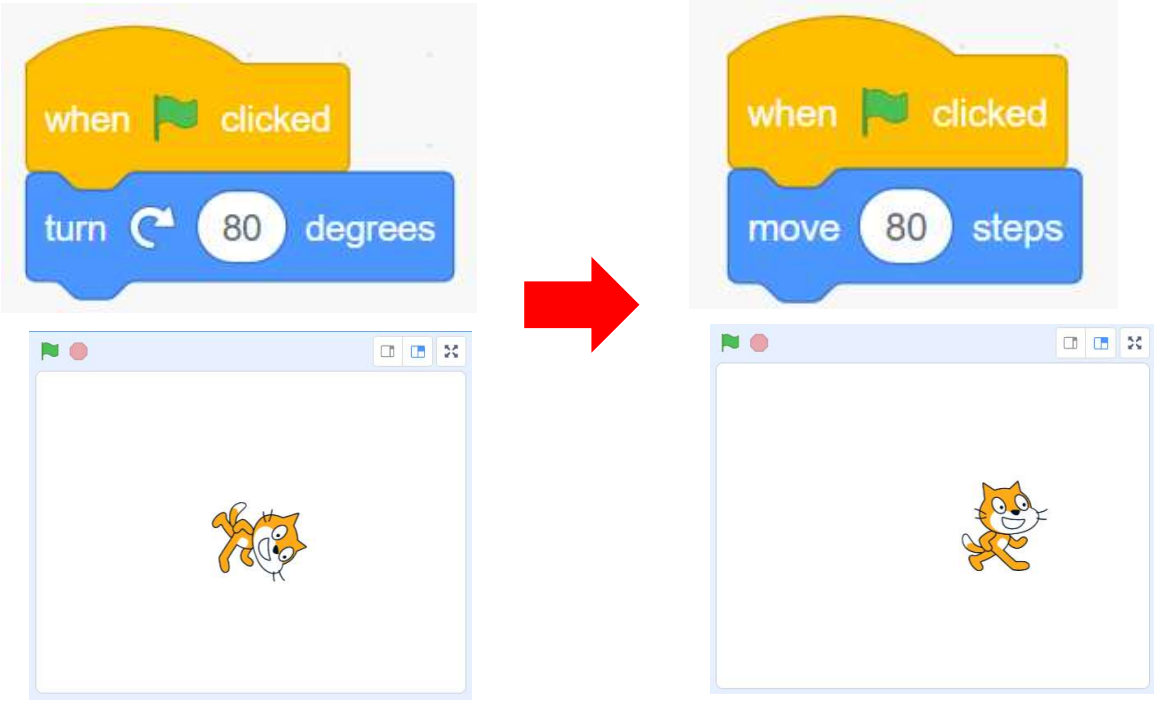


Introducing Scratch Programming

Revision on Key Concepts & Practices

Testing and debugging: Testing a computer program is the process of checking if it can produce outcomes as designed. Debugging a computer program is the process of finding out ways to revise the program so that the bugs can be removed.

E.g. I want to control the cat moving forward to the right. Test if the program works as designed, debug and remove the bugs.



Unit 2: Exploring Under the Sea Student Guide

Content

Exploring Under the Sea

- ❑ Let's review the coding environment in Scratch by putting the letters A, B, C, D and E in the correct boxes.

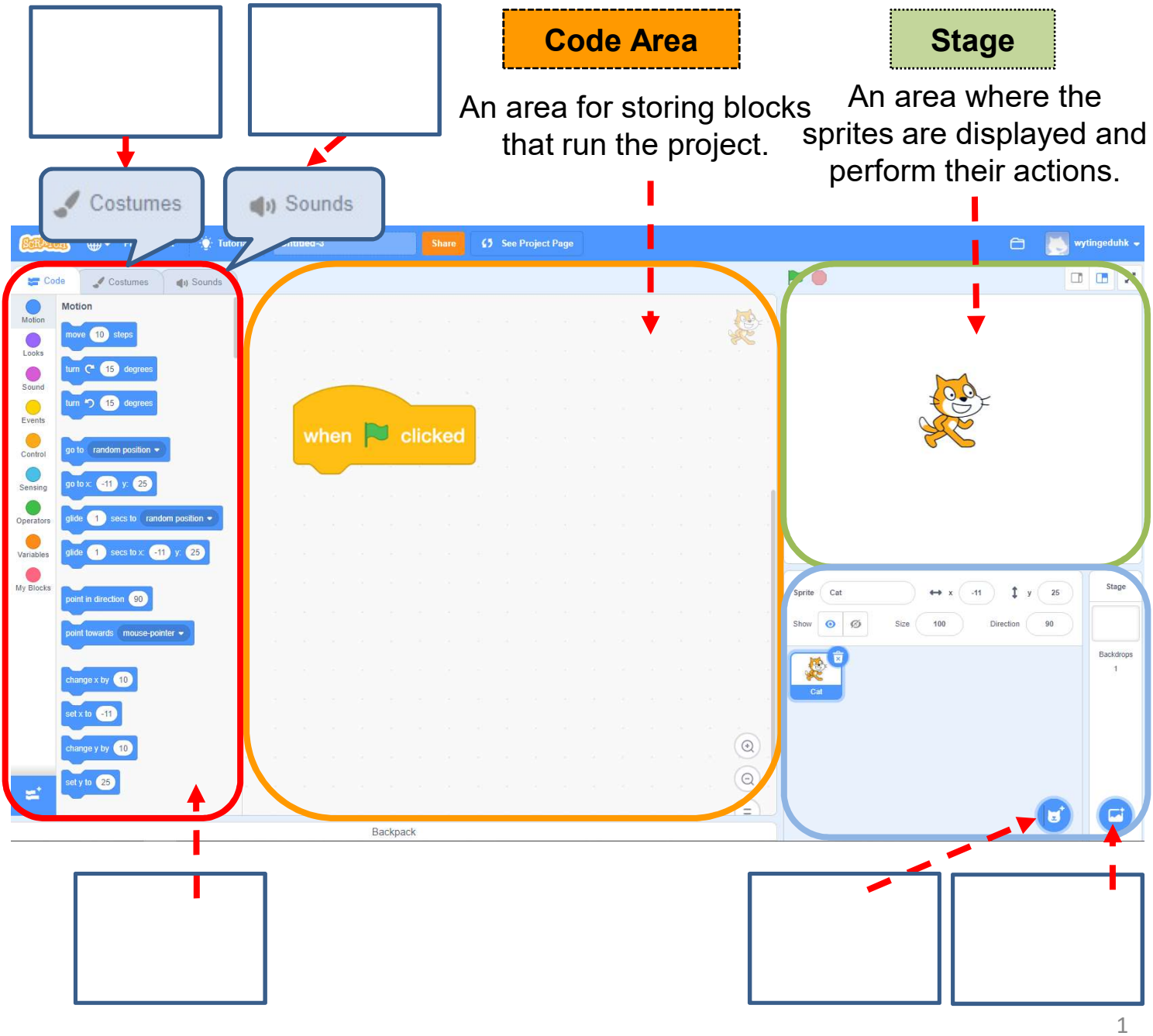
A. Block Palette		**B.** Adding / Changing costumes
C. Adding sound	**D.** Adding / Changing Sprite	**E.** Adding / Changing Backdrop

Code Area

An area for storing blocks that run the project.

Stage

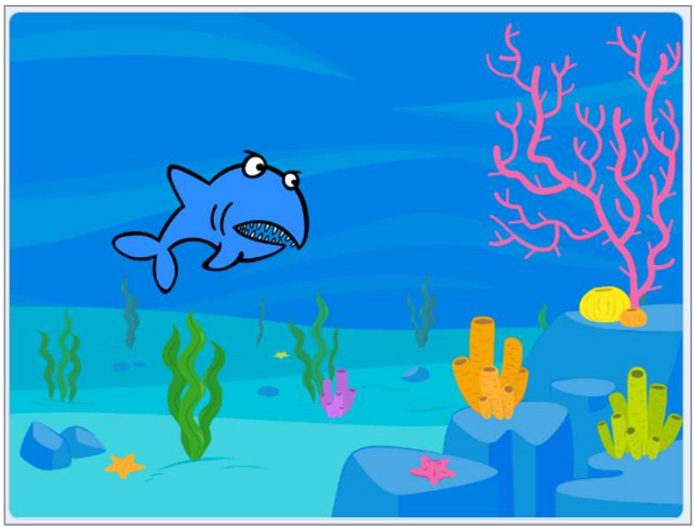
An area where the sprites are displayed and perform their actions.



Exploring Under the Sea

Let's explore under the sea!

Let's take what we've learnt so far with Scratch and make our sprites move! You will explore under the sea where you will add music, sound and movements.

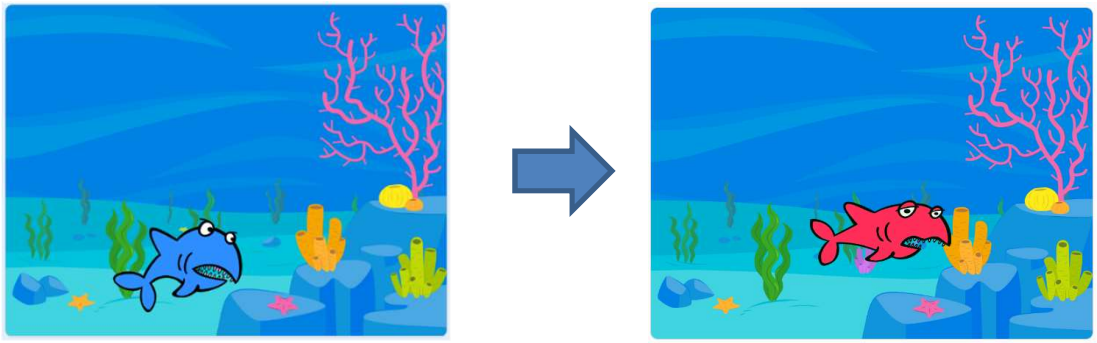


To Play

❑ Play the Scratch project, Under the Sea: https://scratch.mit.edu/projects/722781437

❑ Click the green flag and see what will happen.

- How many sprites do you see?
- How many looks of the sprite?
- Where does the shark swim to?
- Do you hear any sound effect? Is it from the shark or from the sea?



Exploring Under the Sea

- ❑ What did you observe when playing? Try to tick (✓) the correct box as follows: (You may choose more than one answer.)

1. How many sprites appear in the game?
   - ❑ 1
   - ❑ 2
   - ❑ 3

2. Which color(s) did the shark's costume change?
   - ❑ Red
   - ❑ Yellow
   - ❑ Black
   - ❑ White
   - ❑ Purple

3. Any sound effects in the game?
   - ❑ Ripples sound
   - ❑ Rain sound
   - ❑ Bite sound of shark sprite
   - ❑ Cheers
   - ❑ No sound effect

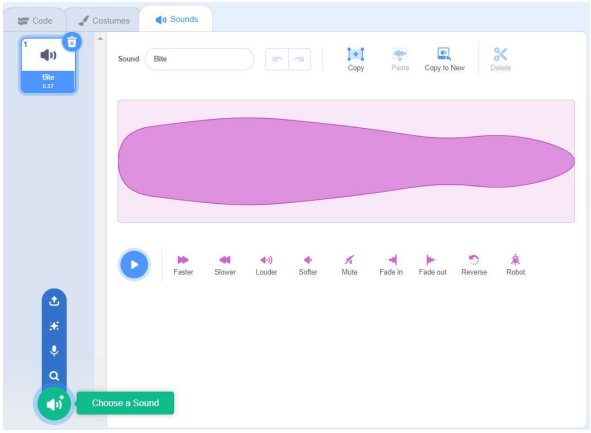
4. Which background(s) appeared in the project?
   - ❑ Forest
   - ❑ Under the Sea
   - ❑ Swimming Pool
   - ❑ Playground

Exploring Under the Sea

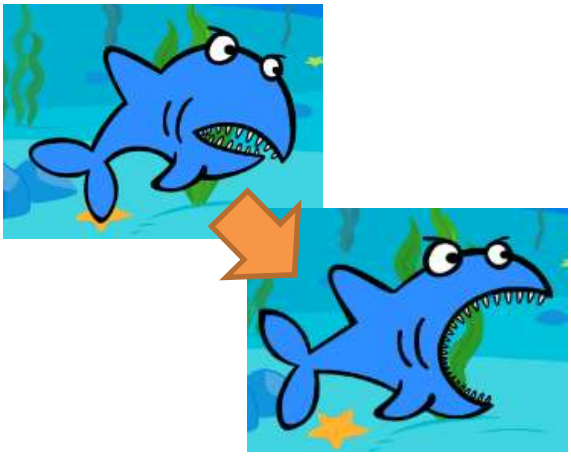
To Think

- After playing the Scratch project, we need to think how to code.
- Let's think about which action you should take first? Write down your sequence in the boxes:

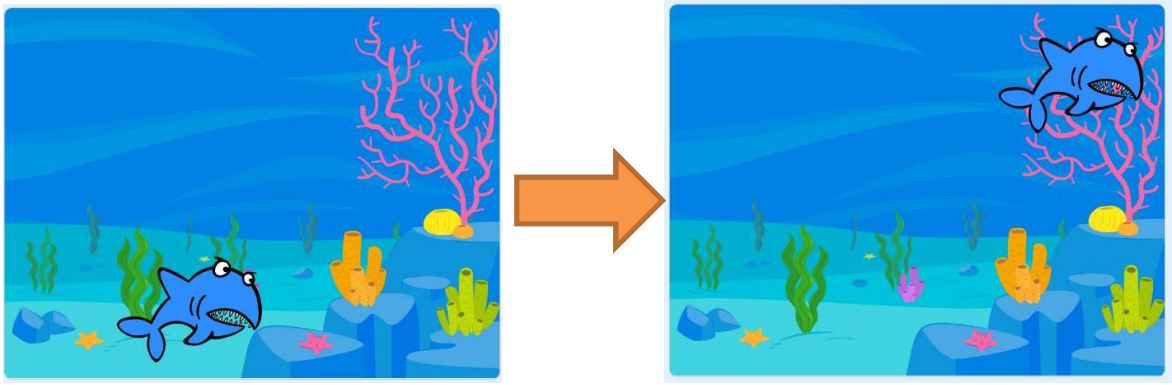
A. Add Music to Backdrop

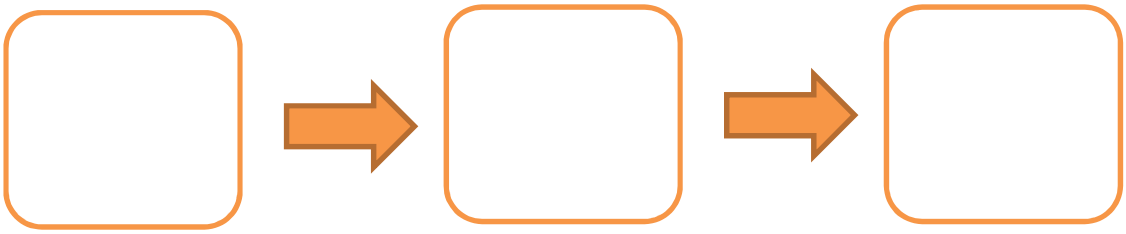


B. Change Looks of the Sprite



C. Make Sprite Swim



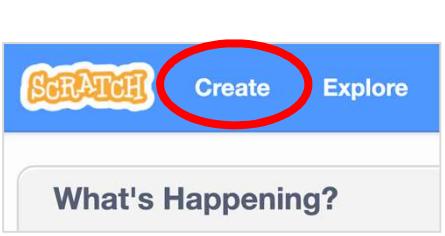


Exploring Under the Sea

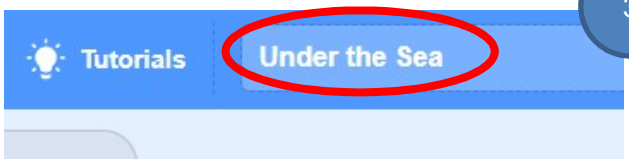
Start Here

1. Sign into your account at https://scratch.mit.edu/.

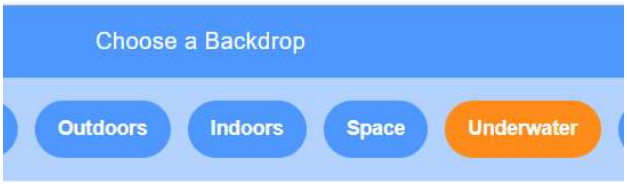
2. Go to “Create” to start a new project.

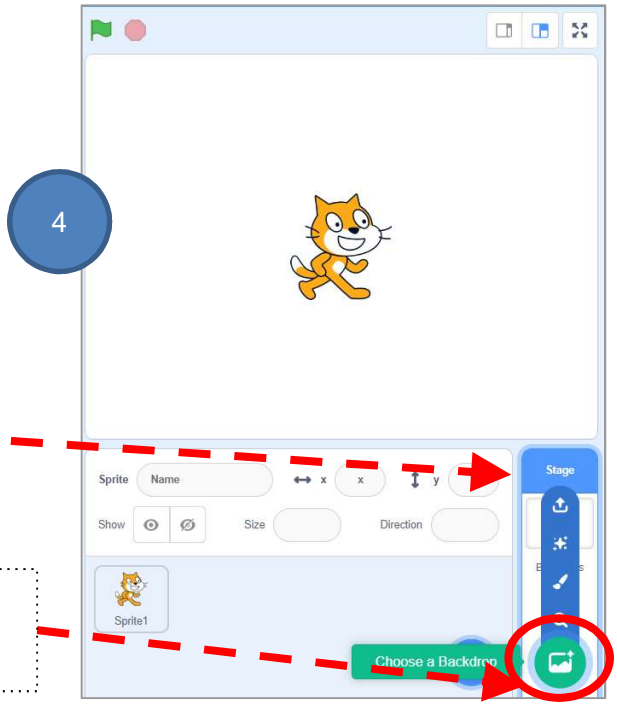


3. Name it Under the Sea.



4. Choose an underwater backdrop for your stage.





Click on the Stage

Click on the “Choose a Backdrop” icon

Exploring Under the Sea

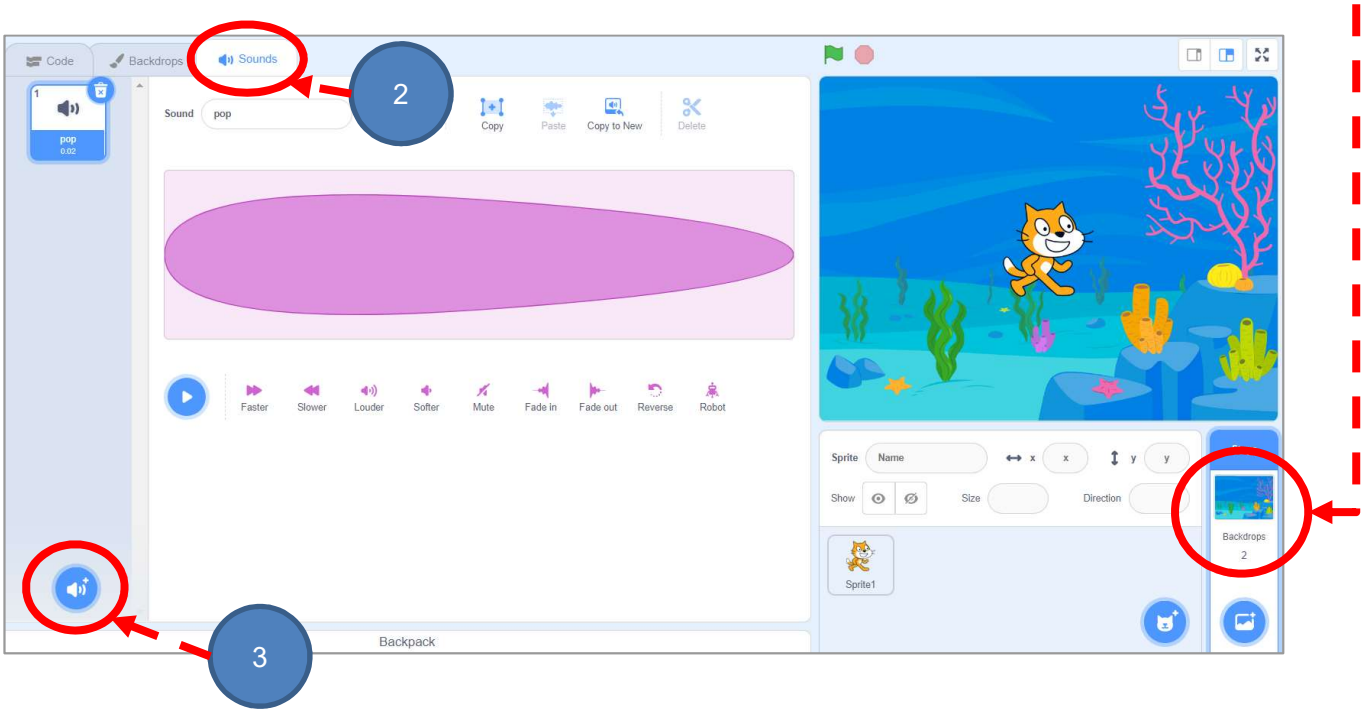
To Code: (A) Add Music to the Backdrop

To Think:
You have learnt to add sound to a sprite, but how to add to the backdrop instead of a sprite?
Let's try and code on your own. You may refer to below hints if needed.

❑ Choose some background music under the sea.

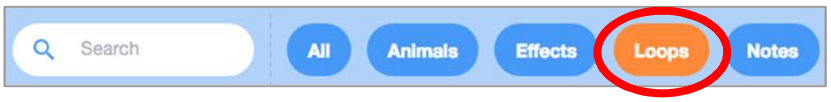
1. Click on the "Stage".

2. Click on the "Sounds" tab.



3. Click on the "Choose a Sound" icon at the bottom left to select a sound.

4. Choose a music sound (under "Loops" category is recommended) to play.

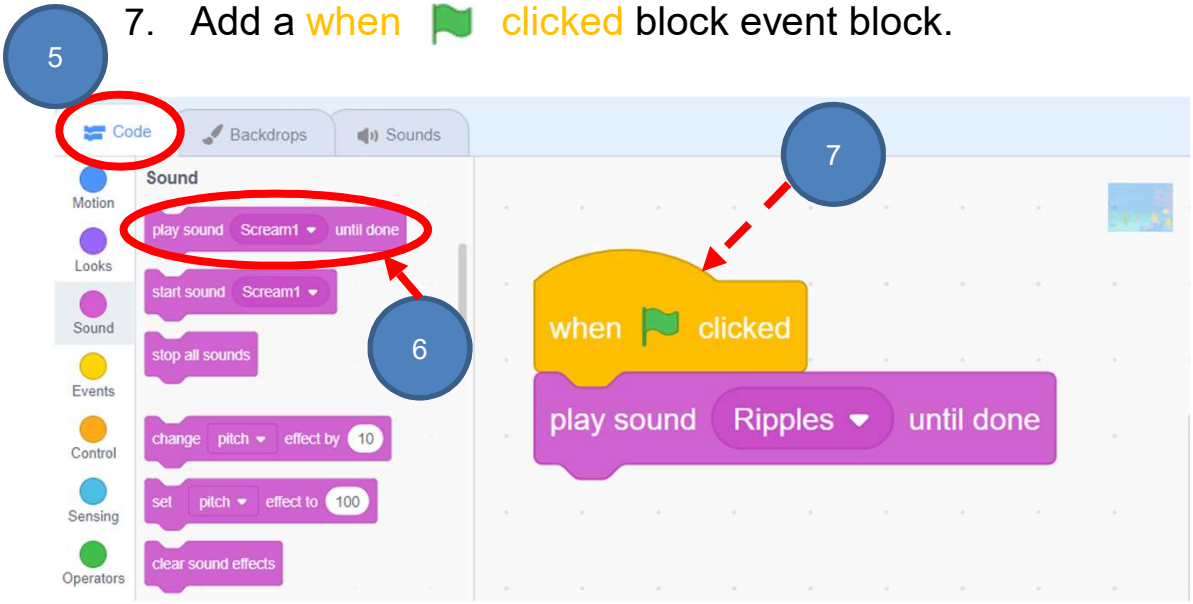


Exploring Under the Sea

To Code: (A) Add Music to the Backdrop

❑ Now add your music to the stage.

5. Click on Code tab.
6. Click Sound drawer, drag out the relevant block to play sound until done.
7. Add a when clicked block event block.



Testing and Debugging

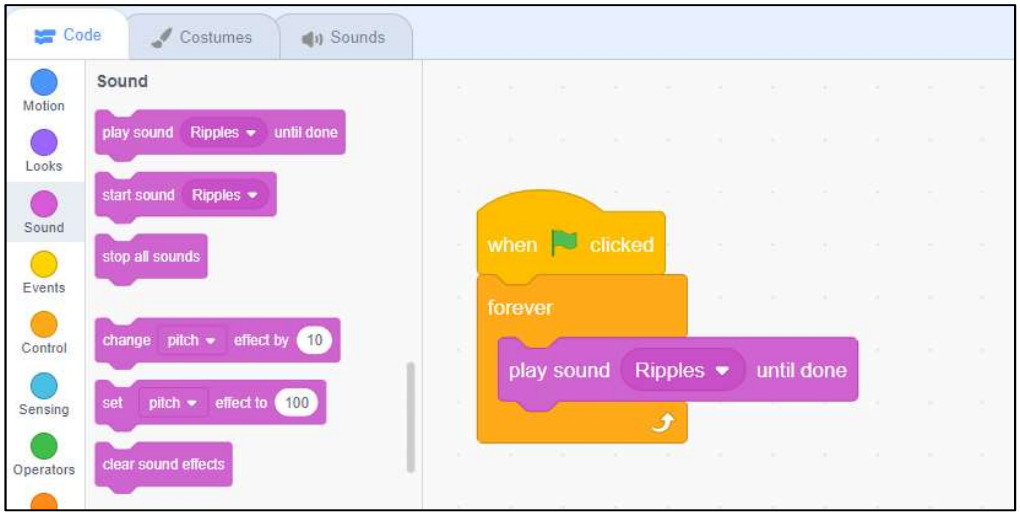
Click the above the Stage to see what happens.

Can you hear any sound? Is the music playing non-stop until the “Stop” button is clicked?

Try adding the “forever” block in “Control” to keep the music playing when the green flag is clicked.

Add Music to the Backdrop

Exploring Under the Sea



To Code: (B) Change Looks of the Sprite

1. Use what you learnt in Unit 1, now choose a good sprite for swimming under the sea! Add multiple costumes for your sprite. [*You can delete the original Scratch Cat costumes.]

2. Switch costume to certain look or next costume. You can also switch costumes to make the fish looks like opening and closing its mouth.

3. Remember which block should we add to make sure we can see each costume when the green flag is clicked?

4. How many times is the costume changed? Do you observe any pattern?

Let's try and code.

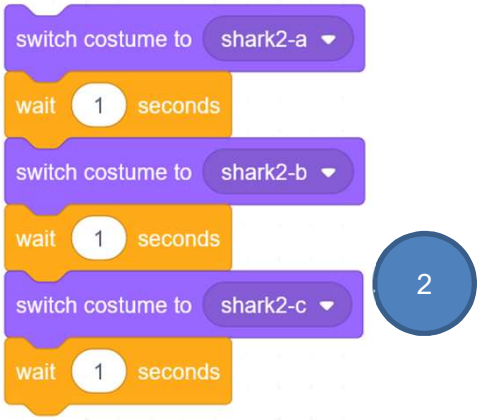
See Appendix P.24

Exploring Under the Sea

To Code: (B) Change Looks of the Sprite

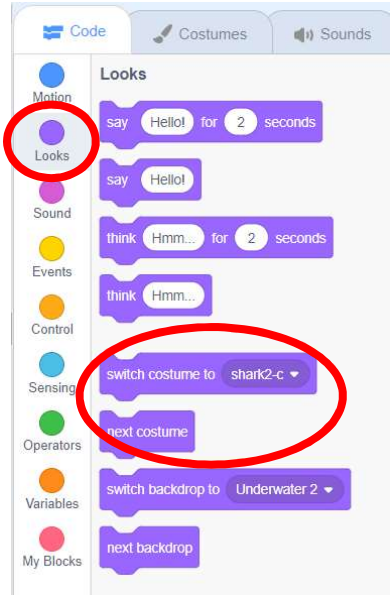
Remember how to switch costumes?

1. Click Looks Drawer, you can switch costume to certain look or next costume.



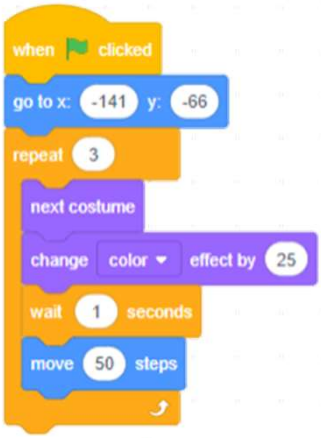
1

2



2. Time how long the sprite waits between switching costumes with the "wait" block.

Which codes are repeated? Can you identify the pattern? Can you use Repeat block to switch costumes?



Testing and Debugging

Time to test! Simply click the green flag to see the changes you made.

Add Music to the Backdrop → Change Looks of the Sprite →

Knowledge builds up: Iteration – repeat

Iteration is repeating a process in order to produce a sequence of outcomes. Other than "forever" block, "repeat" block can also trigger iteration in Scratch.

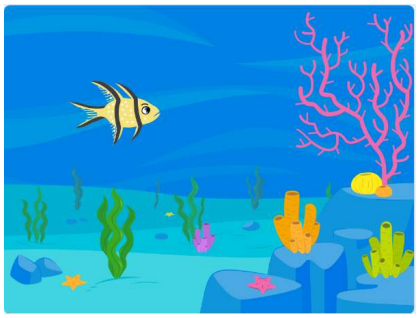
Exploring Under the Sea

See Appendix P.25

To Code: (C) Make Sprite Swim

How does the sprite move? Now add blocks to make the sprite move (swim). You can explore move, turn, go to and glide blocks.

Let's try and code.



Testing and Debugging

When you complete, click the green flag to test it and see if everything works fine.

Knowledge builds up: Sequence

Put the following morning routines into a reasonable sequence by filling 1, 2, 3...

( ) Put on your pants / skirt.
( ) Get out of bed.
( ) Eat breakfast.
( ) Get on bus/get in car to drive to school.
( ) Grab your backpack.
( ) Put on your shoes.
( ) Brush your teeth.
( ) Put on your shirt.

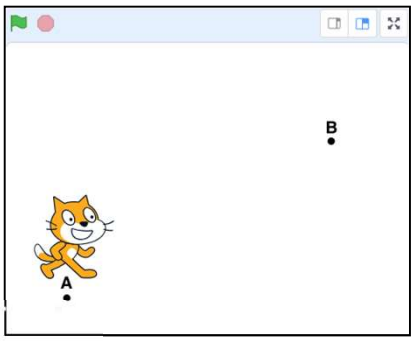
Exploring Under the Sea

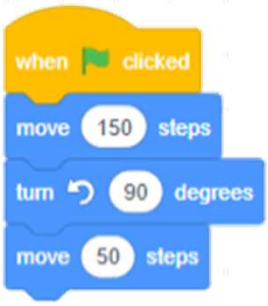
To Code: Understanding Sequence

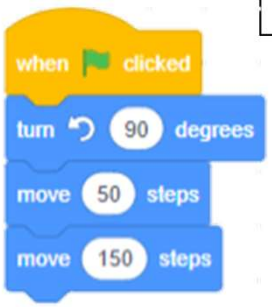
Do you think sequence is important?

Let's try and explore.

Which of the following sequences of commands (sets of blocks) will make the Scratch cat move from point A to point B on the stage? Please tick.







Knowledge builds up: Sequence

Sequence is a key concept in programming. It is the order in which the programming statements are executed. A wrong order would lead to incorrect programming results. For example, there is a specific sequence of dialogues for a meaningful conversation to go on.

Being incremental and iterative

Add Music to the Backdrop → Change Looks of the Sprite → Make Sprite Swim

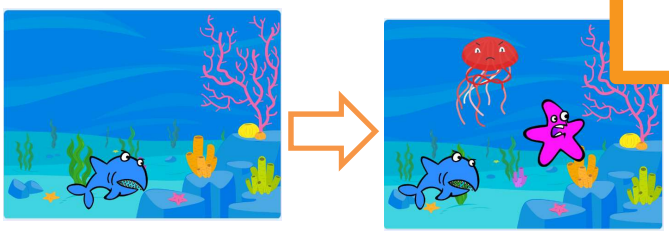
Knowledge builds up: Being incremental and iterative

It is a programming approach. It is iterative because it plans for the work of one iteration (Add music) and then another iteration (Change looks). It is incremental because each iteration will be built on the previous one with improvement (completed the first two steps, then enhance the program to make sprite swim) until the programming task is completed. (Examples can be found in Revision on Key Concepts & Practices on P.22)

Exploring Under the Sea

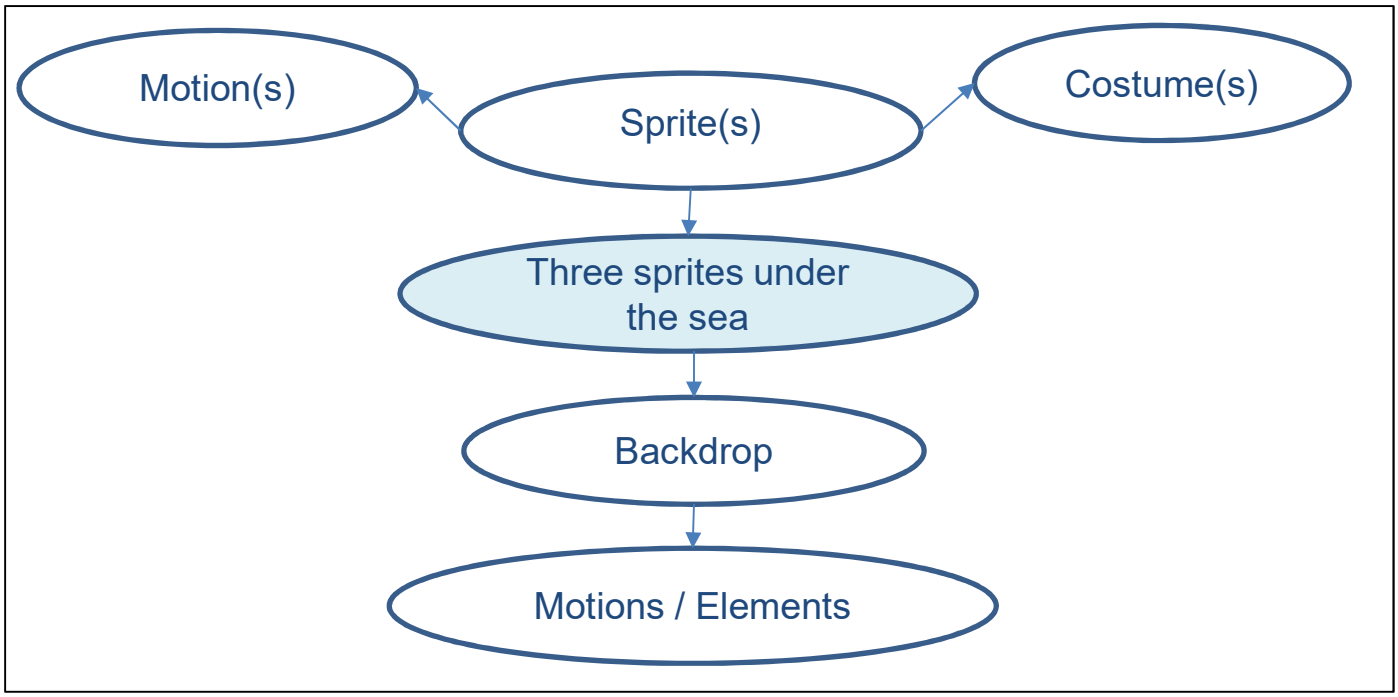
To Create

Let's make more friends under the sea!



See Appendix P.26-27

- ❑ Task 2: Make your design, e.g. add two more sprites, and more different effects! Take a look on the following mindmap, think about what you want to add (e.g. motions and costumes of sprites etc.)



- ❑ How will you plan the programming sequences for your own design? Please fill in the boxes with A, B or C.

A. Add Sprites	B. Change Costumes	C. Add Sound

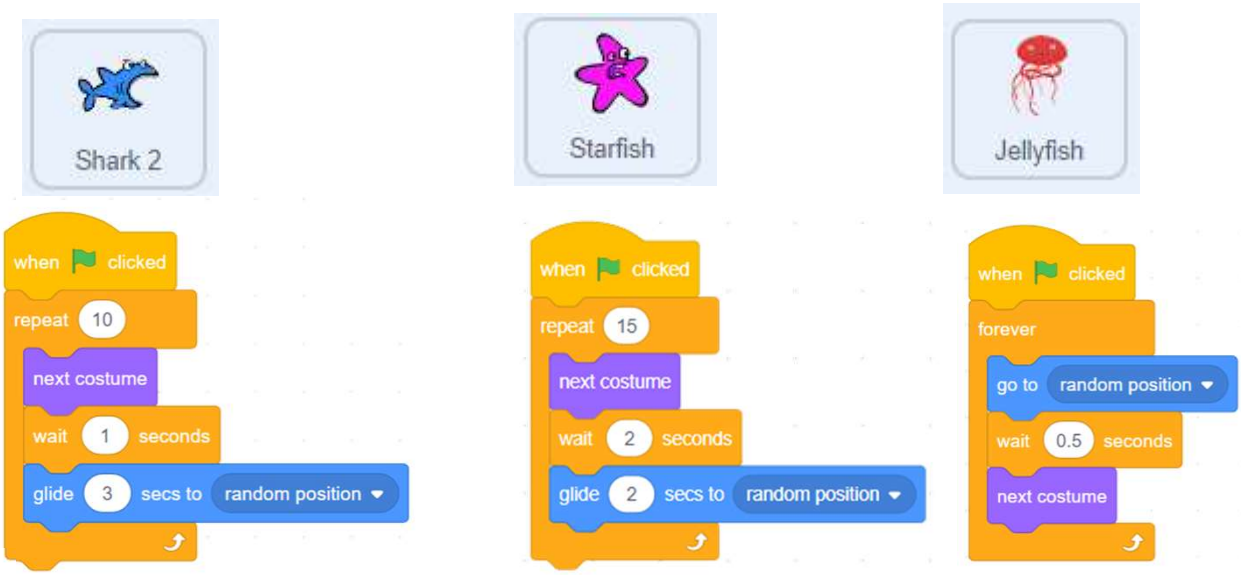
Exploring Under the Sea

See Appendix P.28-29

To Create

In Task 2, we have three sprites, swimming and having fun under the sea.

When we click the green fag, all the sprites start to move. This is called **parallelism** in programming.



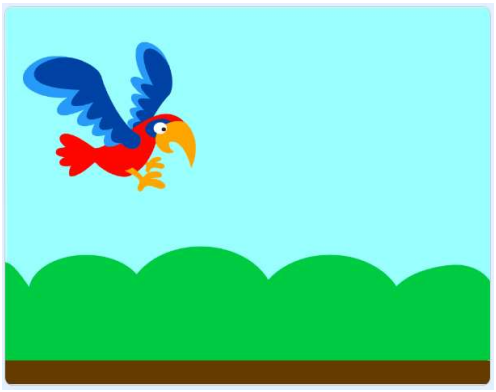
Knowledge builds up: Parallelism

Scratch uses parallelism to allow more than one events to take place at the same time. For example, Scratch allows more than one character to perform an action at the same time in a Scratch program.

(Based on the above code blocks, when green flag clicked, the above three sprites will perform actions together.)

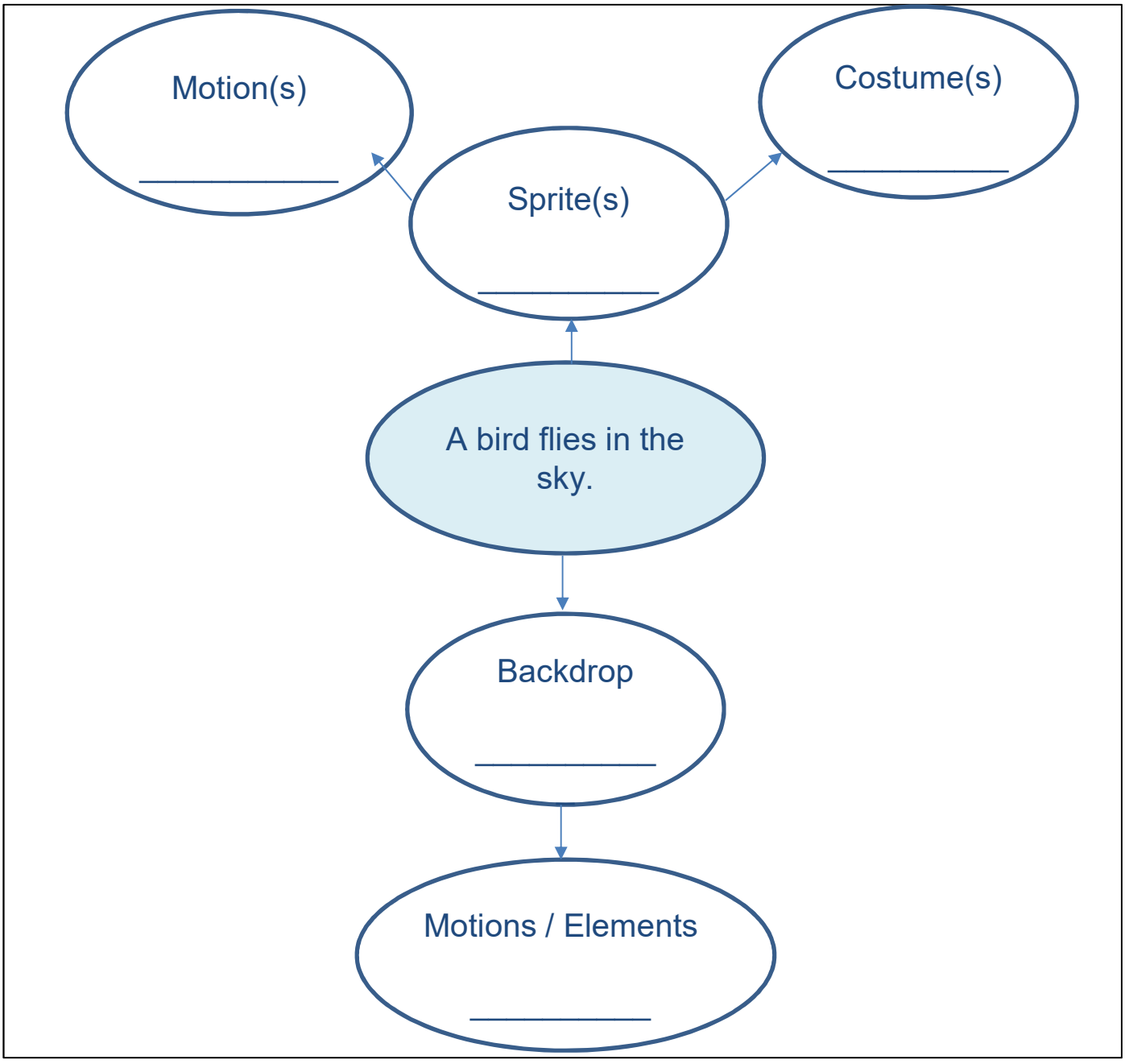
Exploring Under the Sea

To Create



- ❑ Can you make use of what you learnt in this unit?
- ❑ Let's draw something new.
  e.g. A bird flies in the sky.

- ❑ You can use a mind map to plan your design.



Exploring Under the Sea

To Reflect: Two Stars and a Wish Worksheet

Name of Project: _____ Name of Creator: _____

Please write down two things that you like about this project.

1

_____

2

_____

What is one thing you would like to add or change to make this project better?

_____

_____

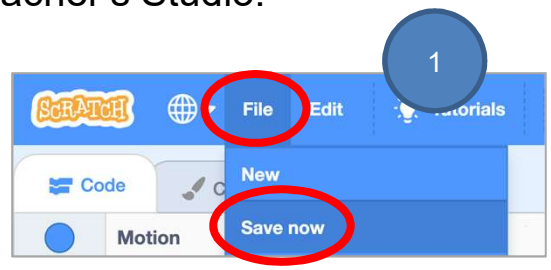
_____

Exploring Under the Sea

Sharing to Studio and Provide Constructive Feedback on Program Design

When you finish, you will add your project to your teacher's Studio.

1. Save your project by clicking "Save now" under the File menu.

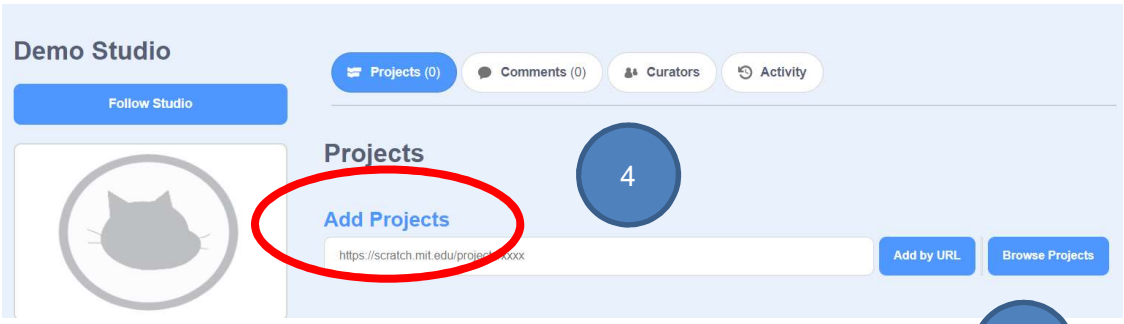


2. Click the orange "Share" button.

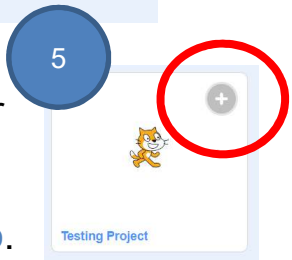
3. Go to your teacher's Studio (they will give you a URL).

3

4. In the "Add projects" column, you can Add by URL or Browse Projects.



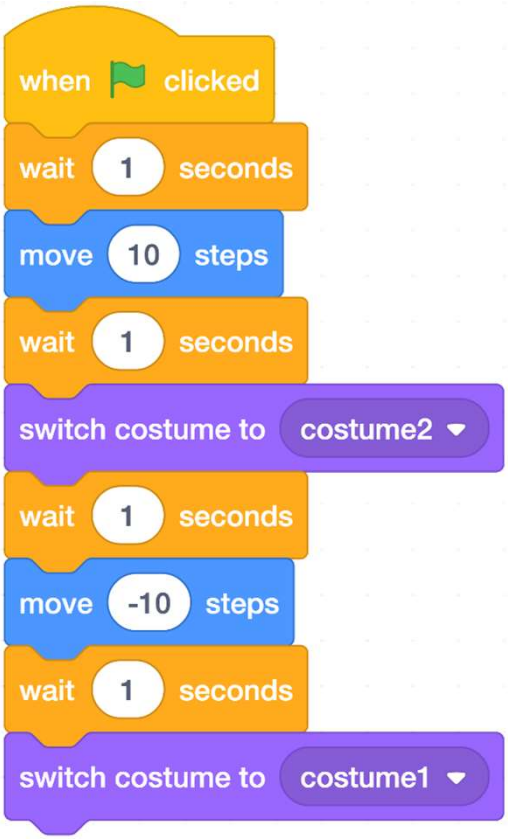
5. If you choose Browse Project, then you will see all your shared projects.
   Find the right project and click the "+" to add it to studio.



Exploring Under the Sea

Review Questions

1. Assuming that the sprite is initially wearing costume1 and facing right, what will its state be 3 seconds after the green flag is clicked?

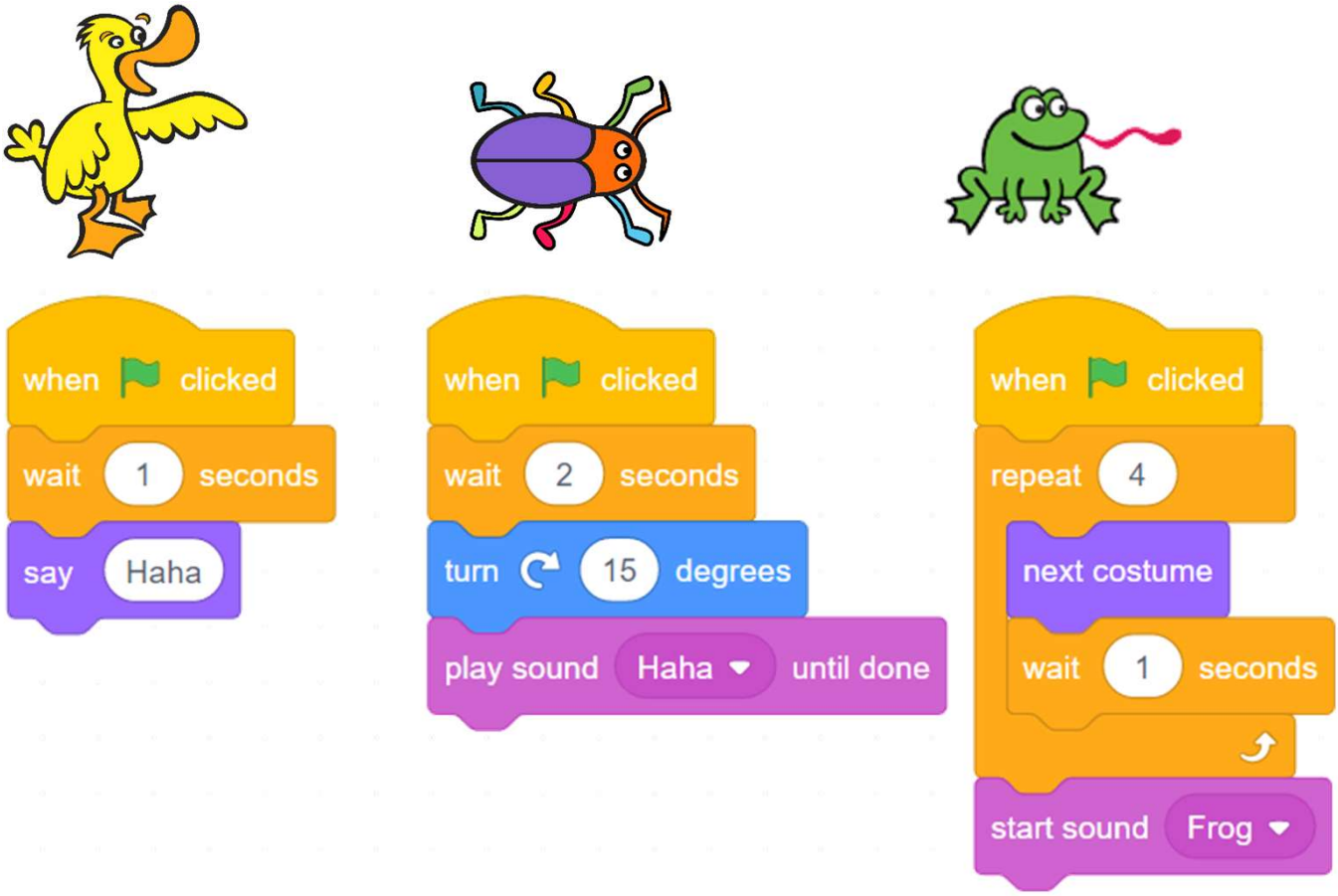


A. The sprite will wear costume1 and be in its original position.
B. The sprite will wear costume1 and has moved 10 steps to the right.
C. The sprite will wear costume2 and be in its original position.
D. The sprite will wear costume2 and has moved 10 steps to the right.

Exploring Under the Sea

Review Questions

2. When green flag clicked, who will make sound first?

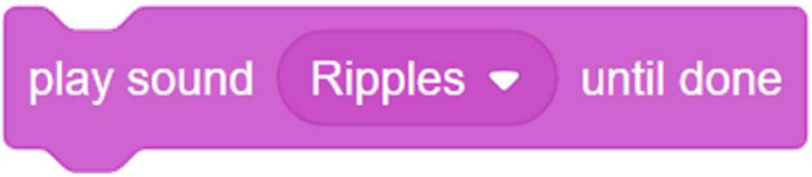


A. Duck
B. Beetle
C. Frog
D. All sprites will make sound at the same time.

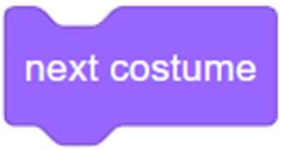
Exploring Under the Sea

Revision on Key Features

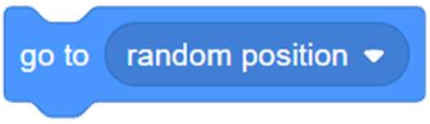
Music:

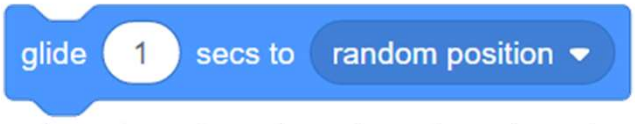


Looks:



Motion:

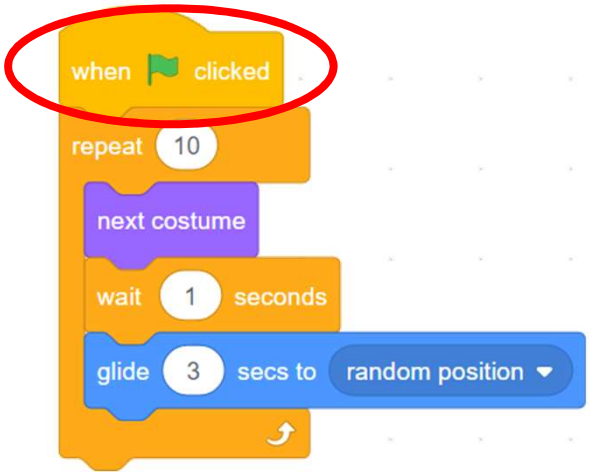




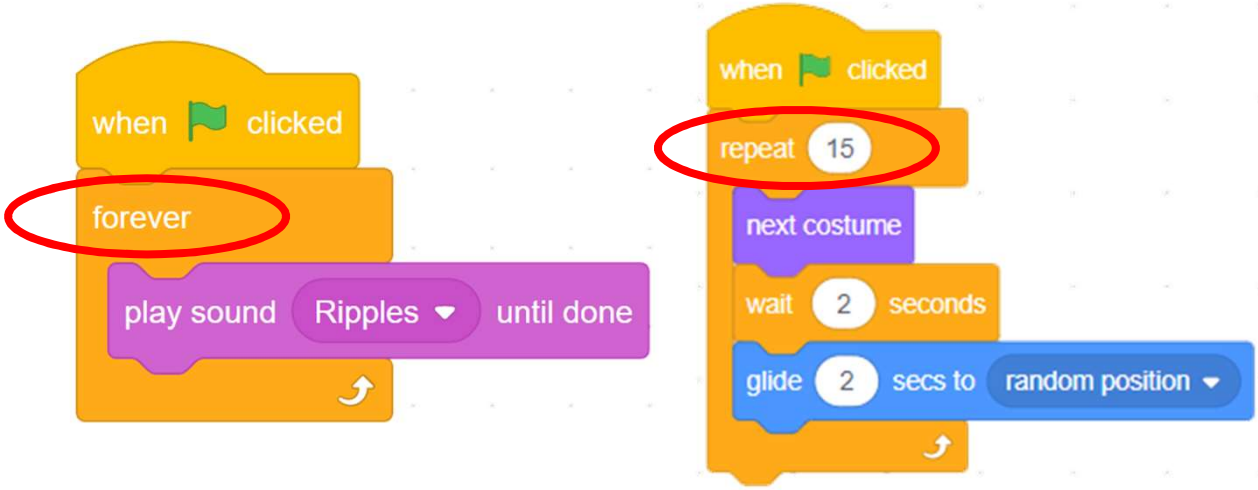
Exploring Under the Sea

Revision on Key Concepts & Practices

Events: We use event blocks to trigger Scratch to take actions. For example, when green flag clicked is a typical event.



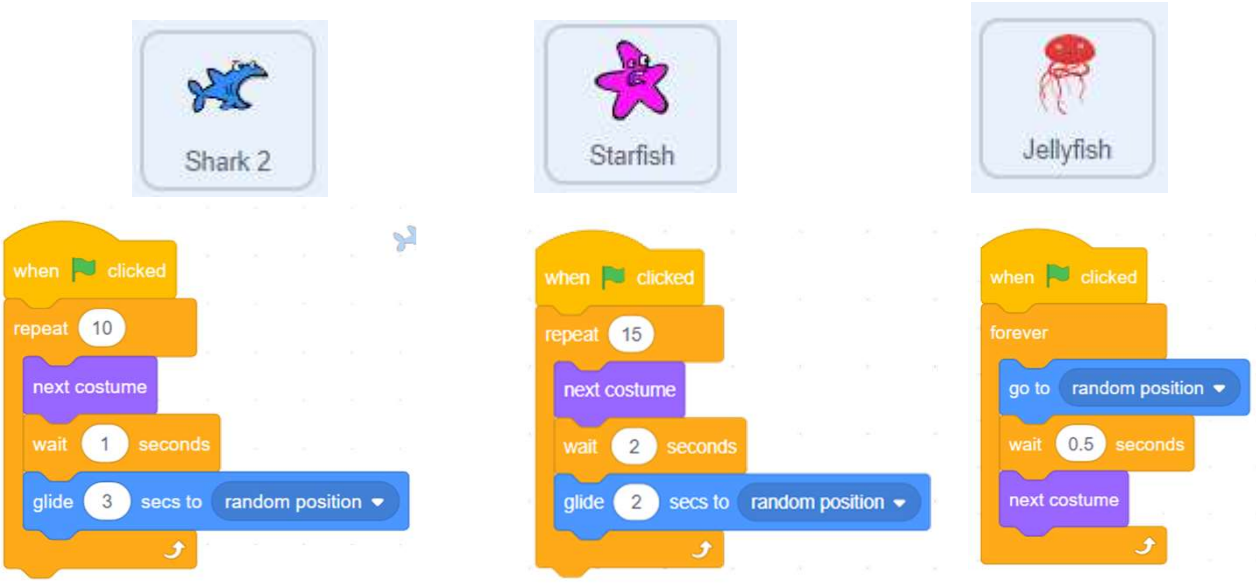
Iteration - repeat: Iteration is repeating a process in order to produce a sequence of outcomes. “Forever” and “repeat” blocks can trigger iteration in Scratch.



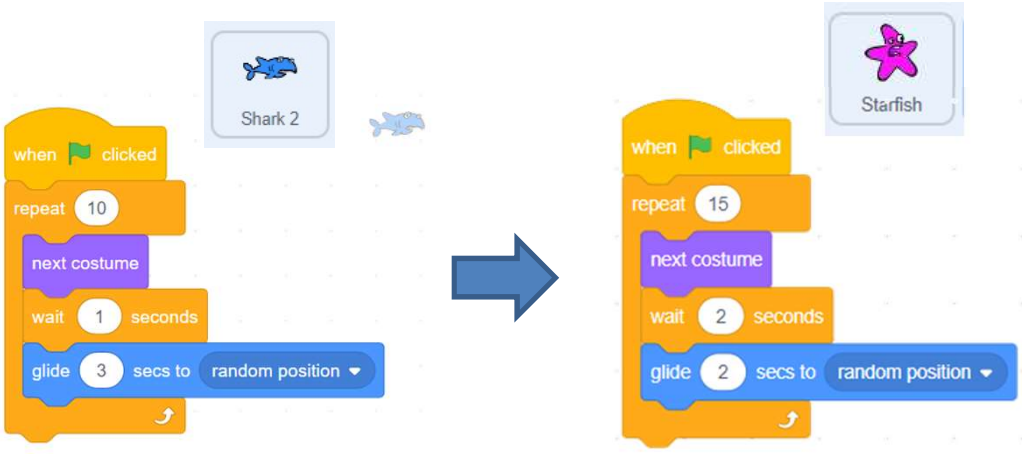
Exploring Under the Sea

Revision on Key Concepts & Practices

Parallelism: Parallelism is a series of codes running at the same time. Scratch supports parallelism across multiple sprites.



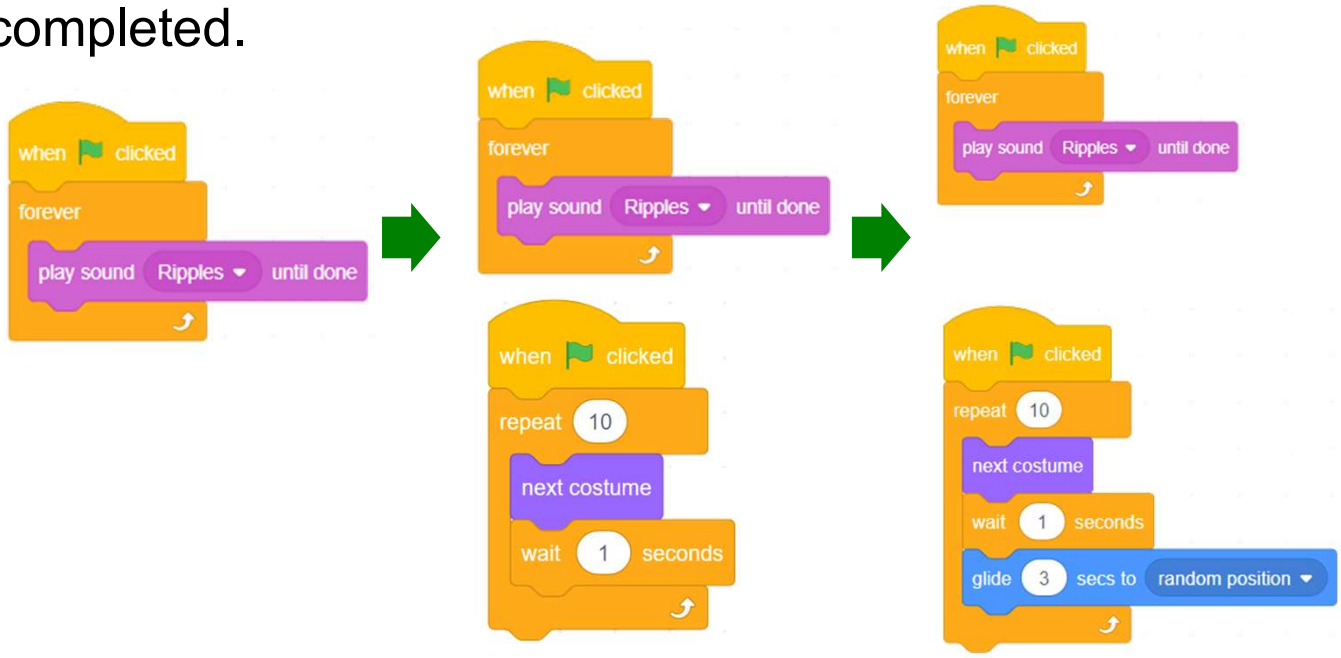
Reuse and remix programs / codes: In Task 2, we reuse and remix the codes of the shark sprite and use them for the second and third sprites.



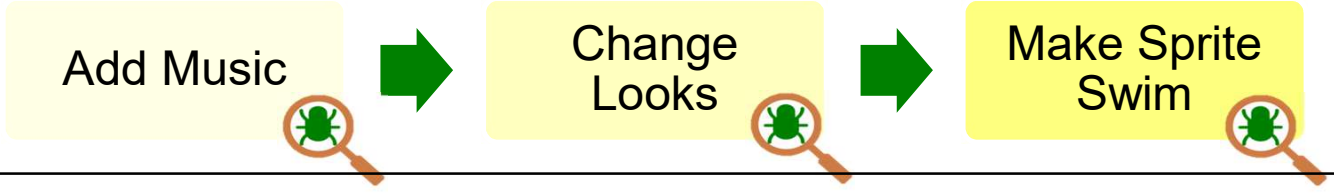
Exploring Under the Sea

Revision on Key Concepts & Practices

Being incremental and iterative: To work out a sub-task as an iteration, try it out, then work out another sub-task in another iteration until the whole programming task is completed.



Testing and debugging: Testing a computer program is the process of checking if it can produce outcomes as designed. Debugging a computer program is the process of finding out ways to revise the program so that the bugs can be removed.



Appendix

Operation Manual

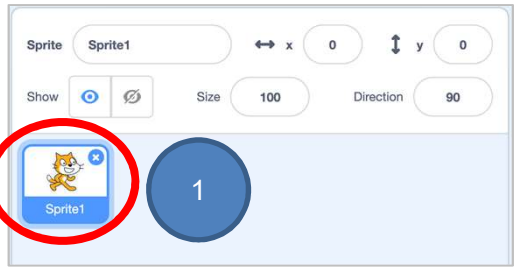
Exploring Under the Sea

See Student Guide P.8

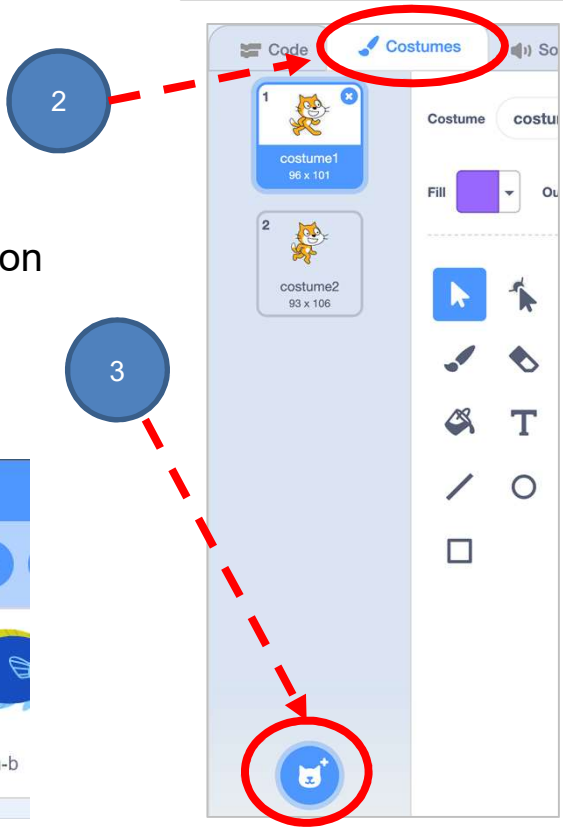
To Code: (B) Change Looks of the Sprite

Choose a good sprite for swimming under the sea!

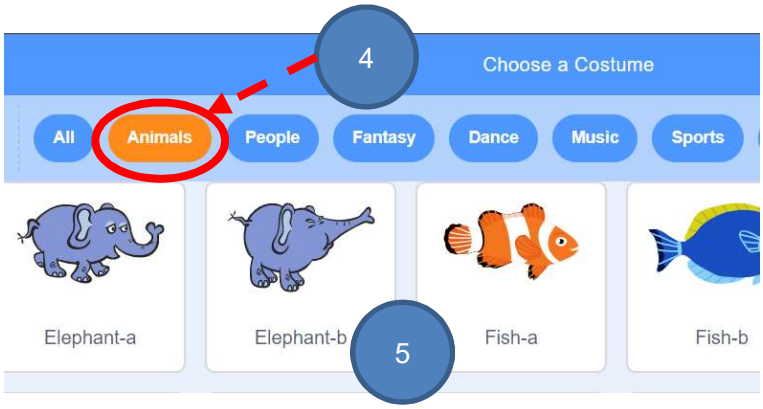
1. Click on the Scratch cat.



2. Click on the **Costumes** tab.



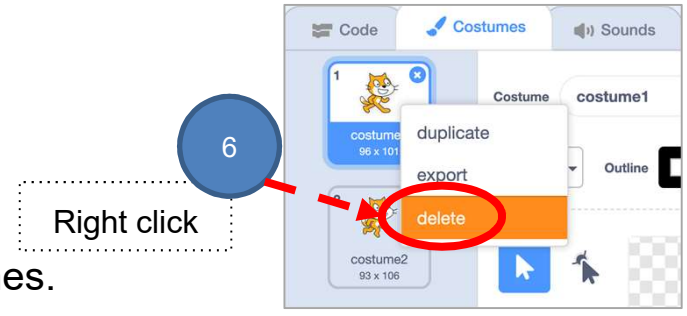
3. Click on the **"Choose a Costume"** button at the bottom left to select a costume.



4. Click on the **"Animals"** category on the top to see a large set of animal costumes.

5. Add *multiple* costumes for your sprite.

6. Delete the original Scratch cat costumes.



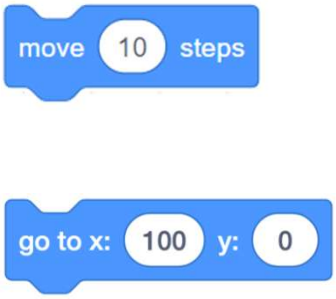
Exploring Under the Sea

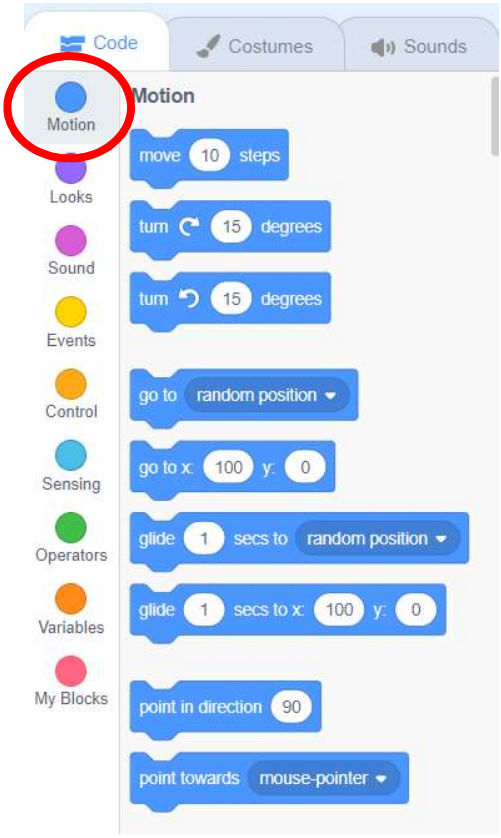
To Code: (C) Make Sprite Swim

See Student Guide P.10

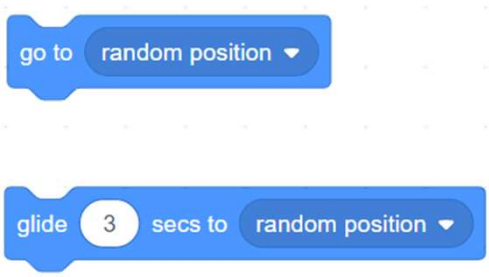
Now, let's make your sprite swim under the sea.

1. Click **Motion** drawer, you can move the sprite to certain point.





2. Or let it swim and explore to random position with different motion.



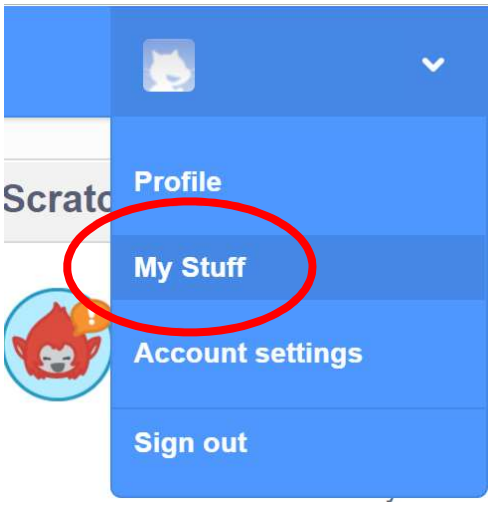
Test to see the difference between **go** and **glide**.

Exploring Under the Sea

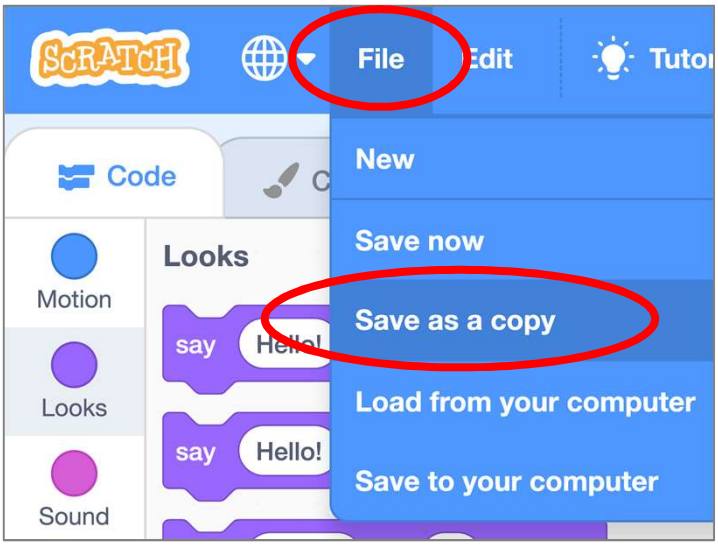
See Student Guide P.12

Save Project as Copy

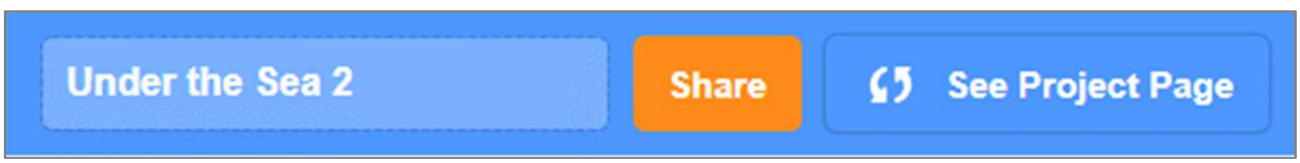
1. Sign into your account at https://scratch.mit.edu/ Click on "My Stuff" then open your "Under the Sea" Project.



2. Save as a copy.



3. Name it **Under the Sea 2**.

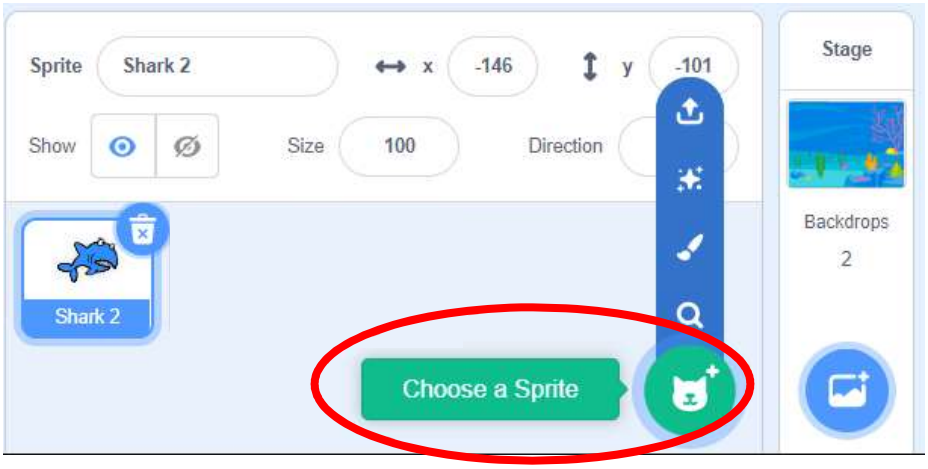


Exploring Under the Sea

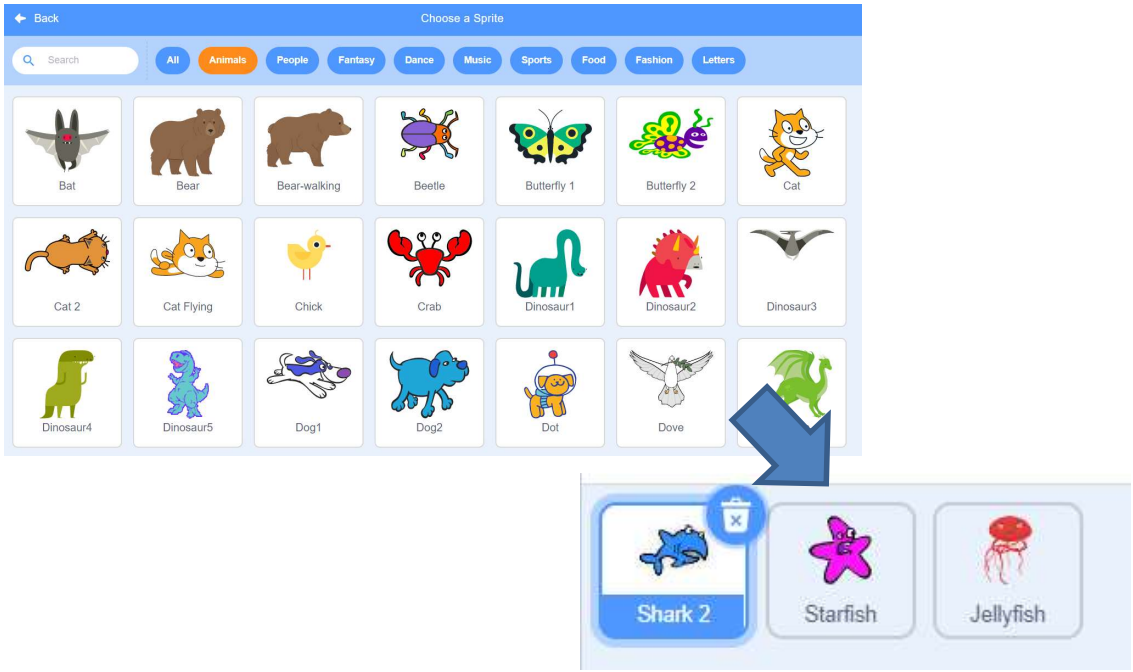
See Student Guide P.12

To Code: Add Sprites

1. Let's make friends under the sea. Click **Choose a Sprite.**



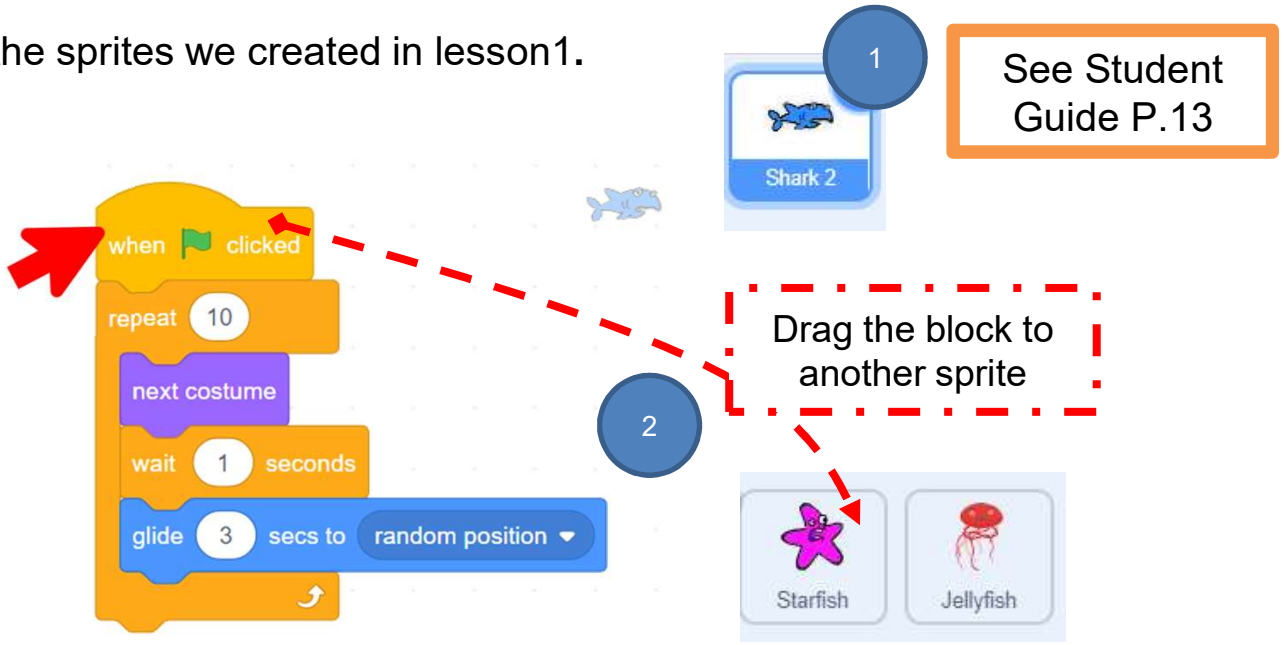
2. Add another sprite (and even more) to the sea.



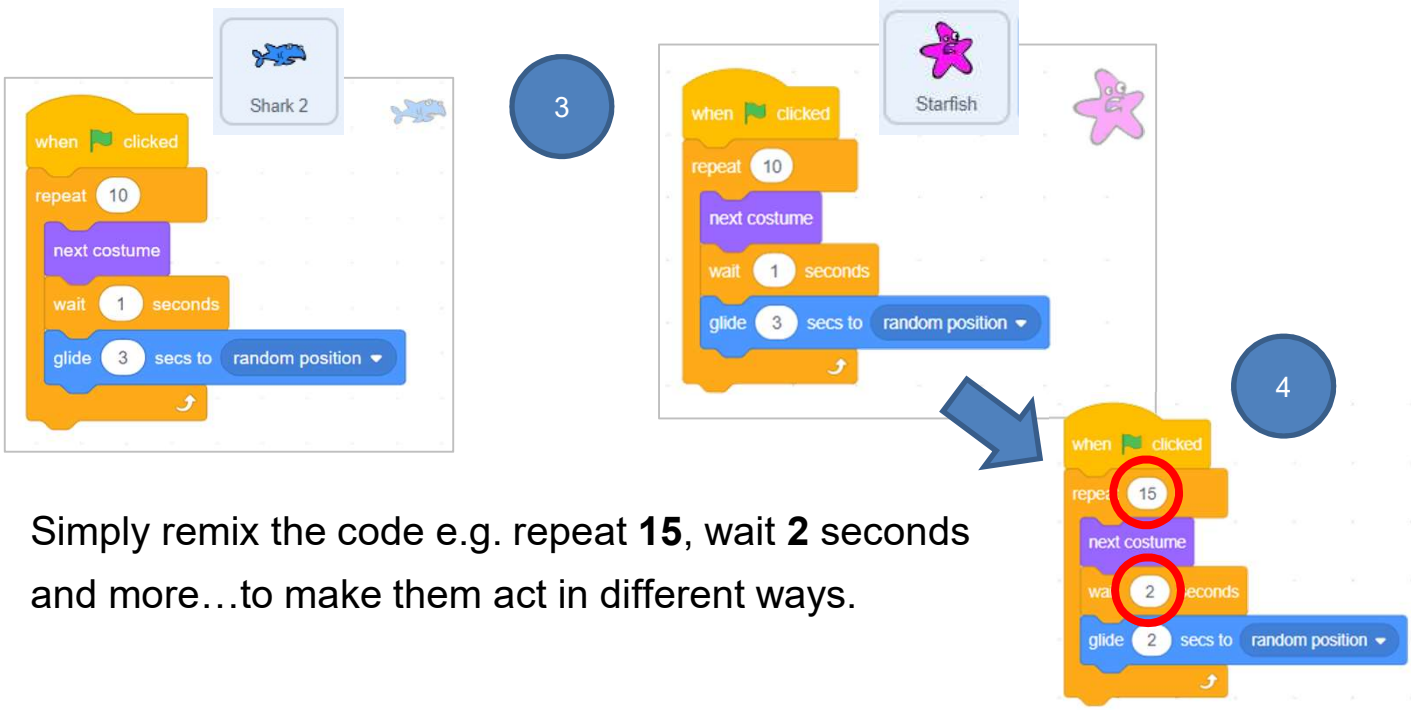
Exploring Under the Sea

Reuse and Remix Code Blocks of Sprites

1. Click the sprites we created in lesson1.



2. Drag the whole set of code blocks to other sprites you want.
3. Now you will see two sprites have the same code blocks.



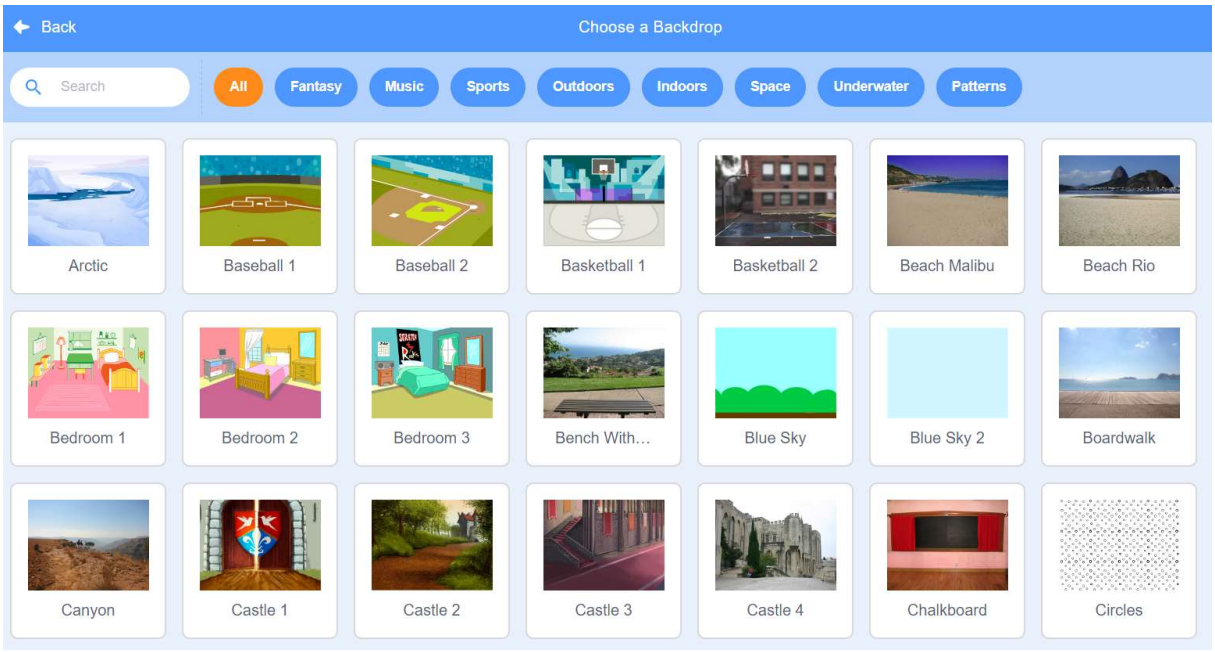
4. Simply remix the code e.g. repeat **15**, wait **2** seconds and more…to make them act in different ways.

Exploring Under the Sea

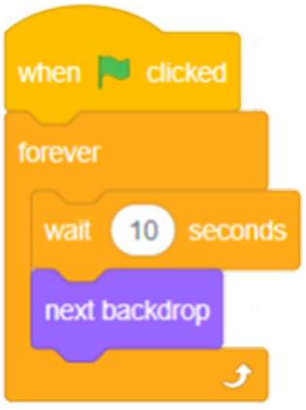
More Effects: Add Backdrop and Sound

See Student Guide P.13

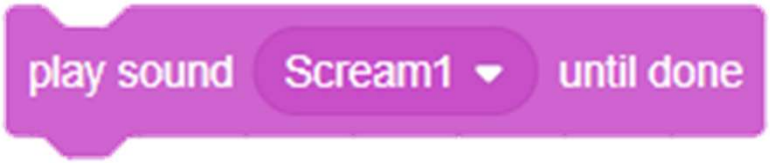
1. You can add more backdrops (e.g. underwater, ocean or sea etc.)!



2. Here it's an example to keep changing the backdrop.



3. You may also add sound effect:



Unit 3: Storytelling
Student Guide

Content

Storytelling

To Play

- ❑ Play the story (Demo) https://scratch.mit.edu/projects/753681874/
- ❑ How many sprites are there?
- ❑ What do they say?
- ❑ Why can they speak like having a conversation? How to do that?
- ❑ Does the conservation run too slow or too fast? Can you adjust the speed?

Scene1

Scene2

Storytelling

❑ Complete the mind map below when playing to observe.

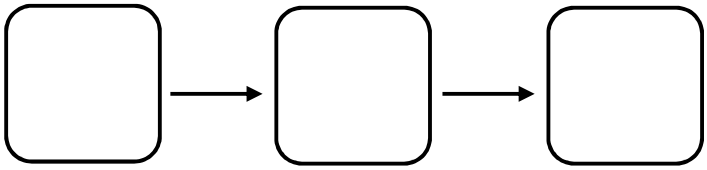
Storytelling

To Think

❑ Check that you understand the sequence of steps of the storyboard in Scene 1 at Metro.

❑ Fill the empty boxes with the correct letters.

Scene 1 at Metro



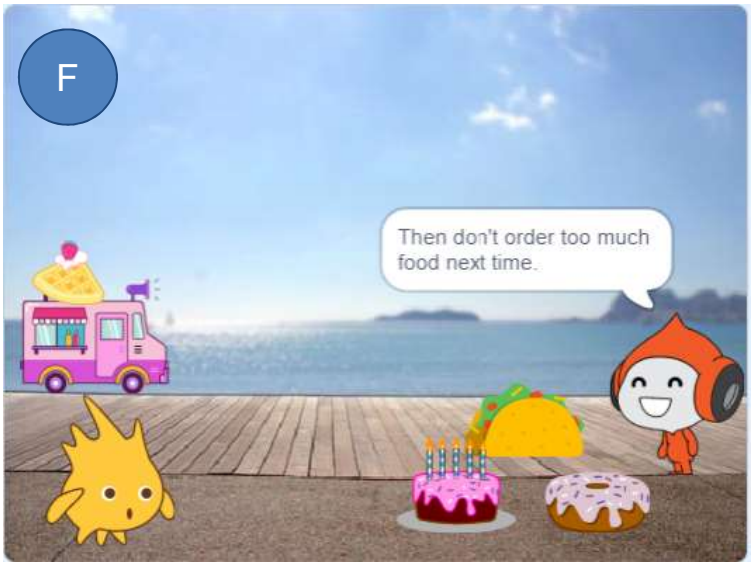
Storytelling

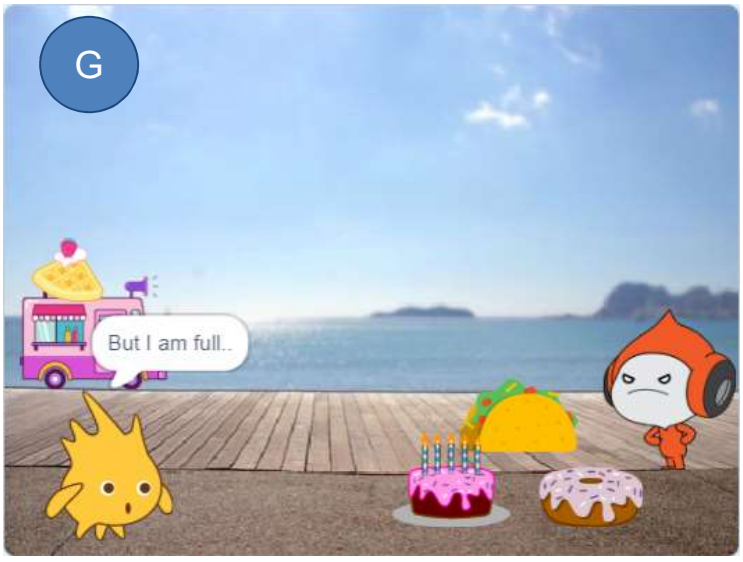
To Think

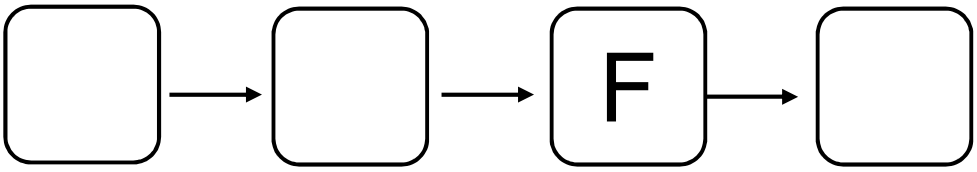
❑ How about the sequence of steps of the storyboard in Scene 2 at the Boardwalk?

❑ Fill the empty boxes with the correct letters “D” to “G”.

Scene 2 at Boardwalk







Storytelling

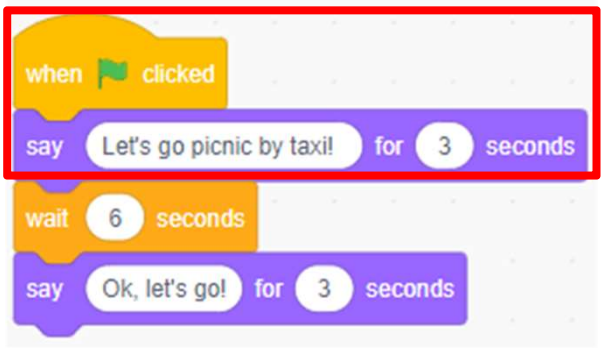
To Think

❑ Review the sequences on Scene 1 at Metro to follow the current blocks that tell the story.

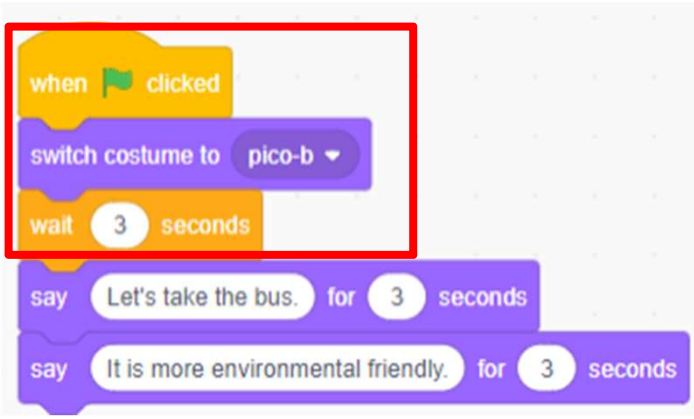
❑ By using the "wait" block, it seems that they are talking.

1

Gobo:



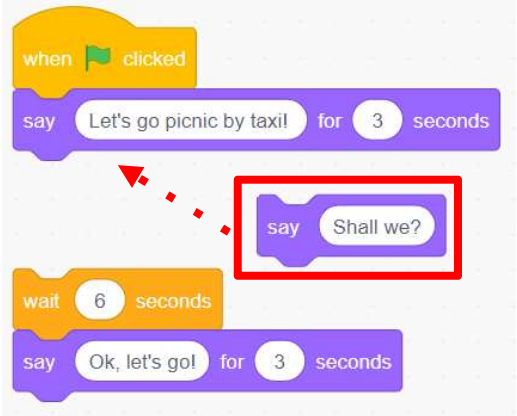
Pico:



❑ Now, you try to add "Shall we?" after Gobo's "Let's go picnic by taxi".

❑ Then click the  and see what happens!

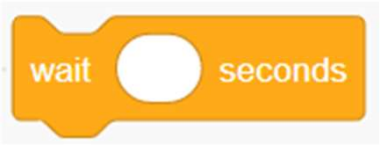
Gobo:



After updating the script, did they speak to each other in order?
Did they speak at the same time?
Why would this happen?

Storytelling

To Think:



❑ What is the problem with the “wait” block?

_____

❑ Why would that happen?

_____

❑ What did you do to solve this problem?

_____

Storytelling

Unplugged Activity: Role Play on Broadcast

Student 1 (Gobo) 	Student 2 (Pico) 
When  **is clicked** Say: Let's go picnic by taxi! **(Raise Card No.1)**	
	WHEN I RECEIVE Message 1 Say: Let's take the bus instead. It is more environmental friendly. **(Raise Card No.2)**
WHEN I RECEIVE Message 2 Say: Ok, let's go! **(Raise Card No.3)**	
	WHEN I RECEIVE Message 3 Say: Why are there so much leftover food? It is not environmental friendly. **(Raise Card No.4)**
WHEN I RECEIVE Message 4 Say: But I am full… **(Raise Card No.5)**	
	WHEN I RECEIVE Message 5 Say: Then don't order too much food next time. **(Raise Card No.6)**
WHEN I RECEIVE Message 6 Say: We are all eco-warrior!	**WHEN I RECEIVE Message 6** Say: We are all eco-warrior!

Storytelling

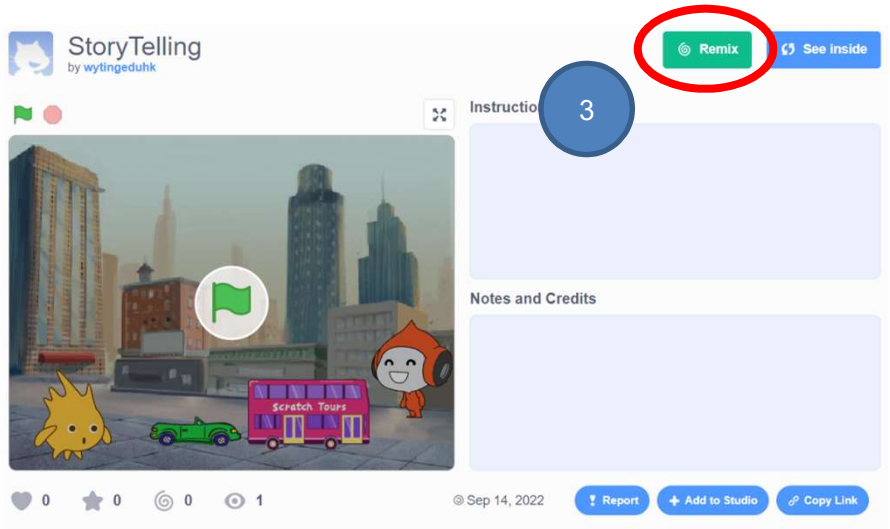
In this activity, you will modify a Scratch project to use "broadcast" and "when I receive" blocks instead of the "wait" block.

To Code

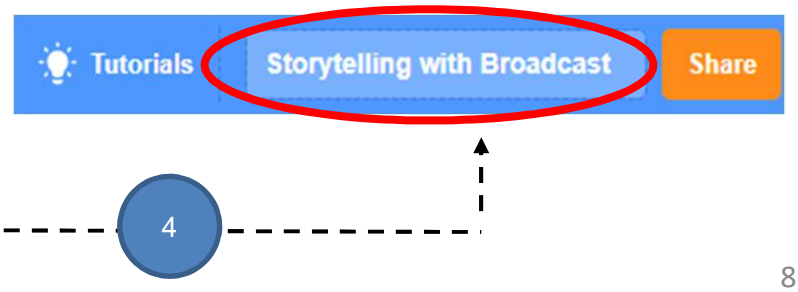
1. Sign into your account at scratch.mit.edu.

2. Go to the Storytelling project at: https://scratch.mit.edu/projects/753681874/

3. We are going to use the code from this project, so we need to "remix" the project first. Click the "Remix" button.



4. You can now use this original code and save it as your own project! Rename the project to "Storytelling with Broadcast" and save it.

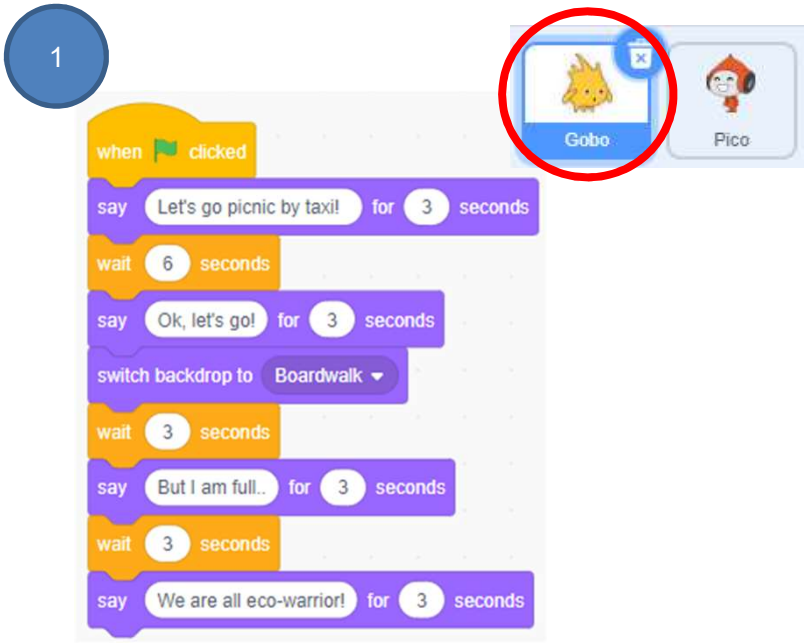


Storytelling

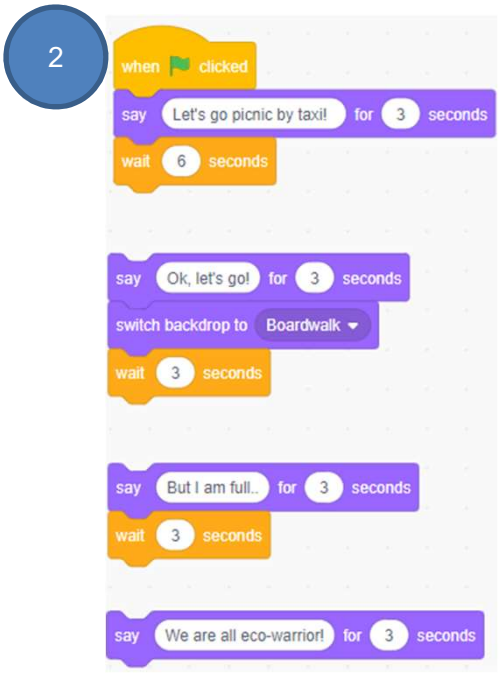
To Code: Remove “Wait” Block

Now we are going to use “Broadcast” and “When I receive” to replace “wait” block. Break the code out for Gobo and Pico by each part of the storyboard, and then remove the “wait” blocks.

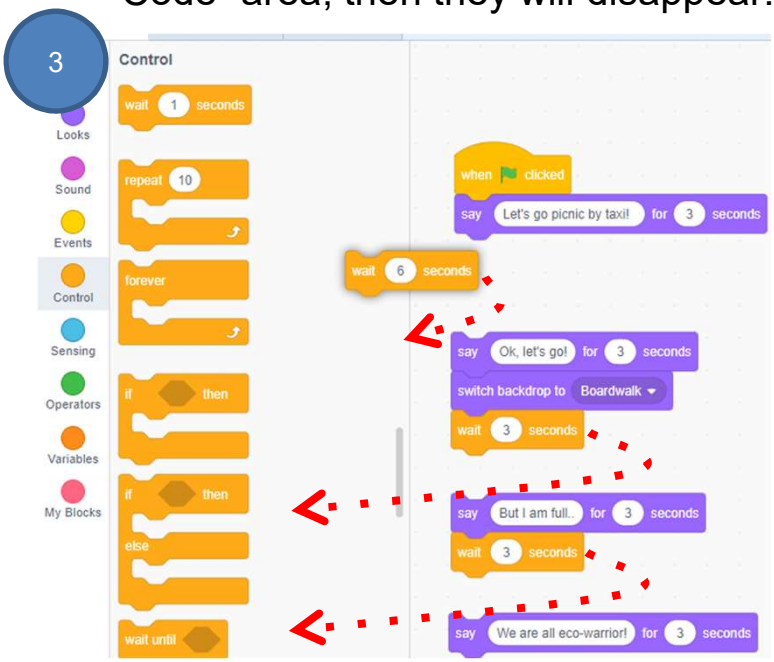
1. Click on Gobo under “Sprite” to open its script.



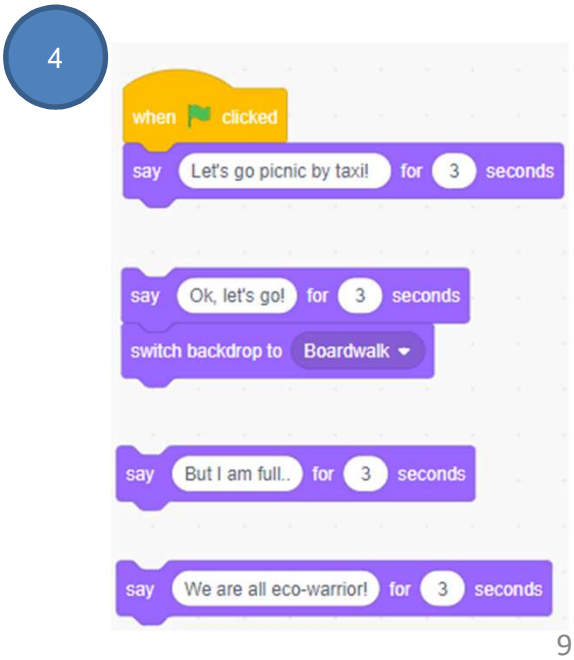
2. Break the code out.



3. Drag the “wait” blocks out of the “Code” area, then they will disappear.



4. Done!

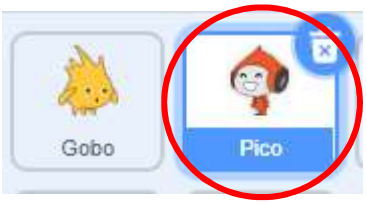


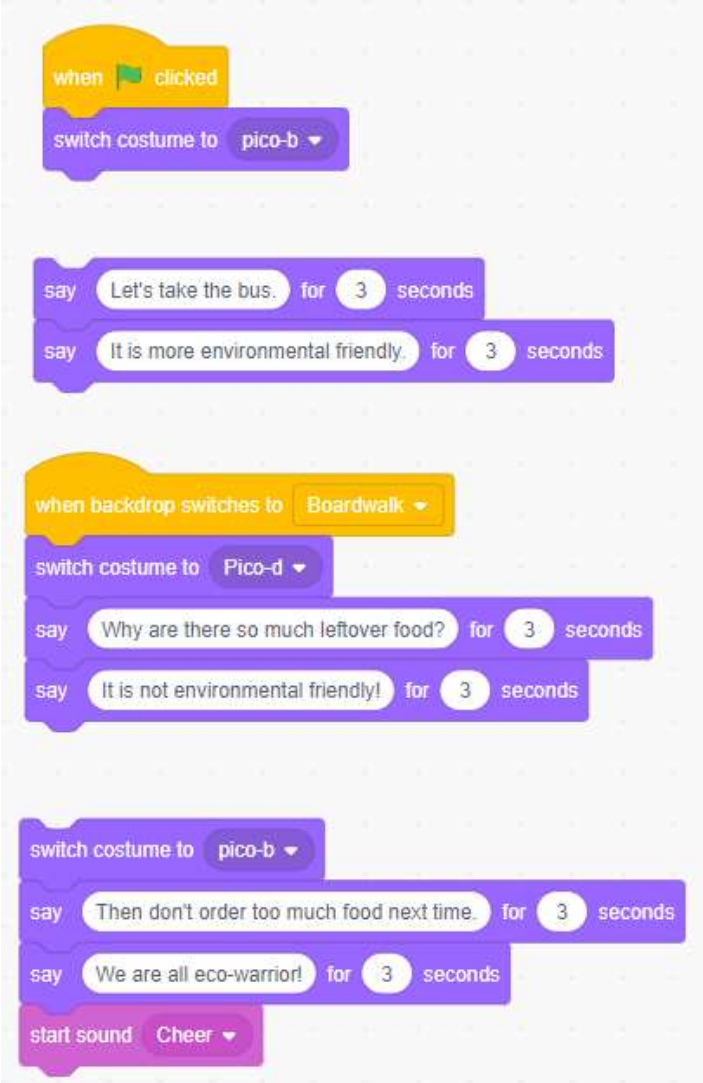
Storytelling

To Code: Remove “Wait” Block

- ❑ Repeat the steps we just completed for Gobo, remove the “wait” block for Pico as below.

Pico’s code:





Testing and Debugging

Let’s test it! After removing all “wait” block, what happened when you click the green flag?

Remove “Wait” Block → ____ → ____ → ____

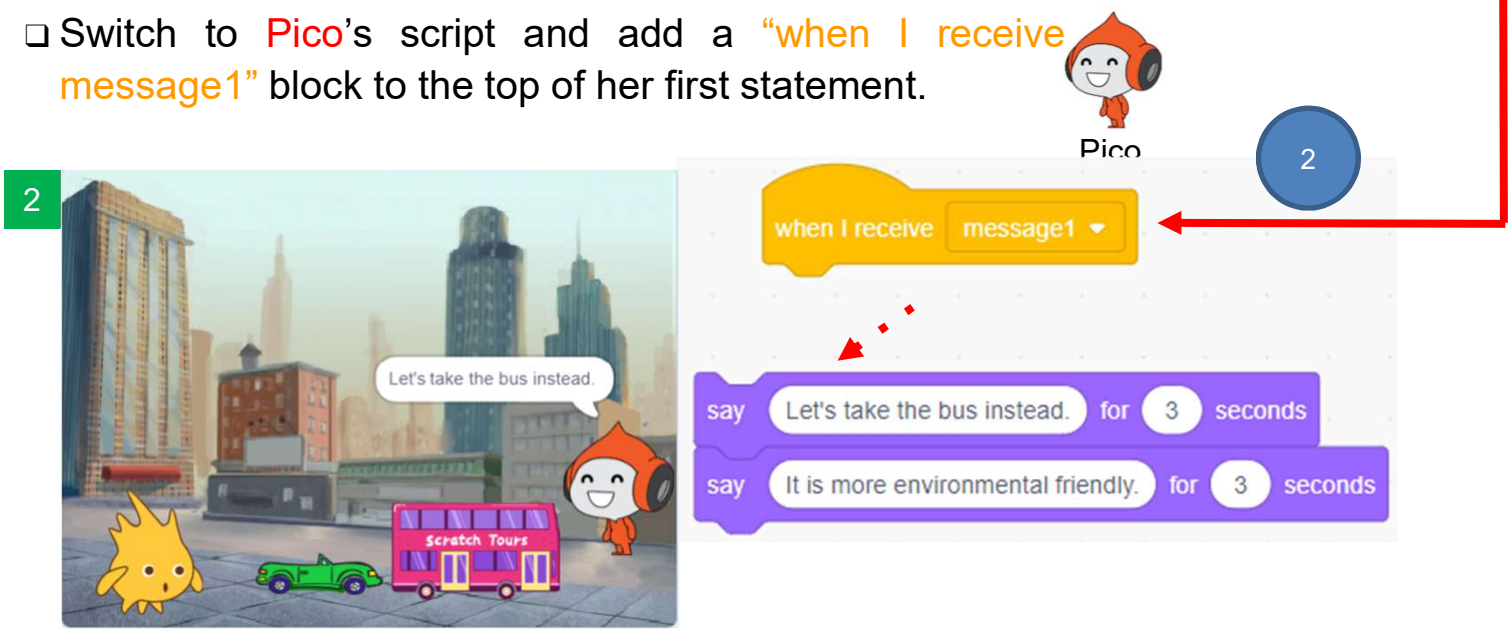
Storytelling

To Code: Replace with “broadcast” and "when I receive"

Use the “broadcast” and “when I receive” blocks to replace all of the “wait” blocks for the conversation of the sprites. Remember that both blocks are in the Events drawer.

❑ Start with Gobo’s first statement in Scene 1 at Metro. Add a “broadcast message1” block to the end. The message number follows the storyboard sequences.

❑ Switch to Pico’s script and add a “when I receive message1” block to the top of her first statement.



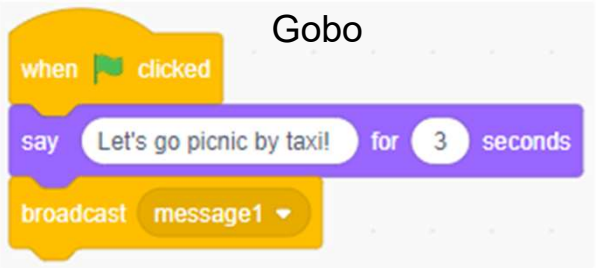
Storytelling

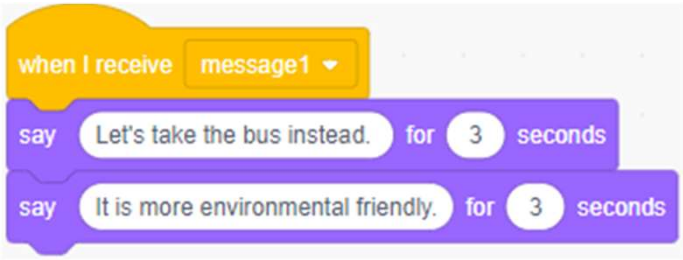
To Code: Replace with “broadcast” and "when I receive"

Now you should have the following blocks after adding “when I receive” and “broadcast” block.

Gobo

Pico





Remember to include a “when I receive” block before a sprite speaks and a “broadcast” block after the sprite speaks.

Testing and Debugging

Let’s test it! When you click the green flag button, what happen? Did the sprites talk to each other?

Remove “Wait” Block → Replace with “Broadcast” and "when I receive" → → →

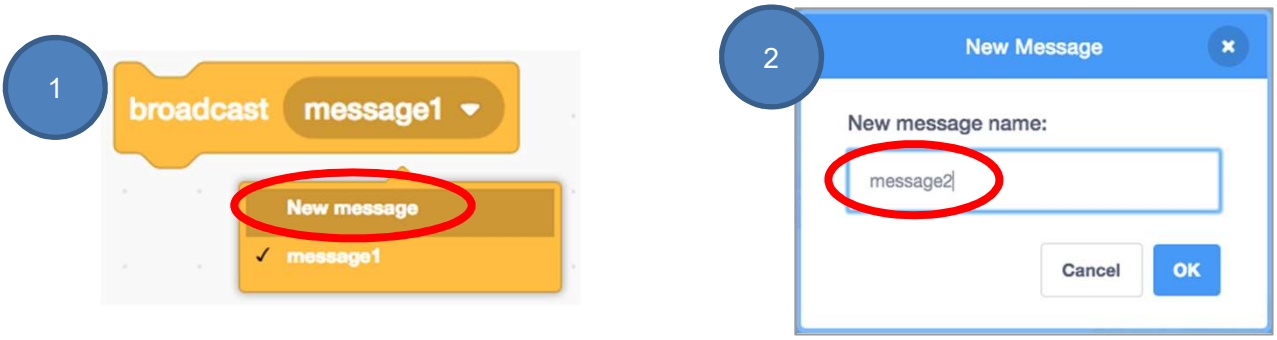
Storytelling

To Code: Complete the Second Dialogue of Scene 1

Let's add "Broadcast" Block to all statements in Scene 1. Now Gobo needs to broadcast back to Pico that she's responded, and it's Gobo's turn to speak.

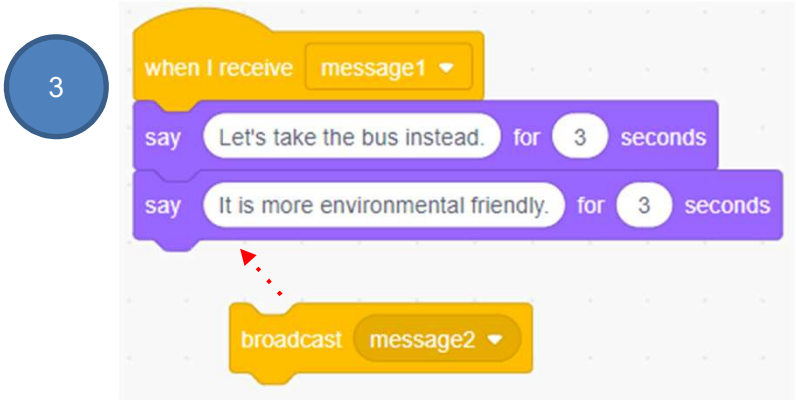
1. Drag out a new "broadcast" block, and click on message1, and select New message.

2. Name it message2.



3. Snap the broadcast block to the end of Pico's speaking block.

Pico



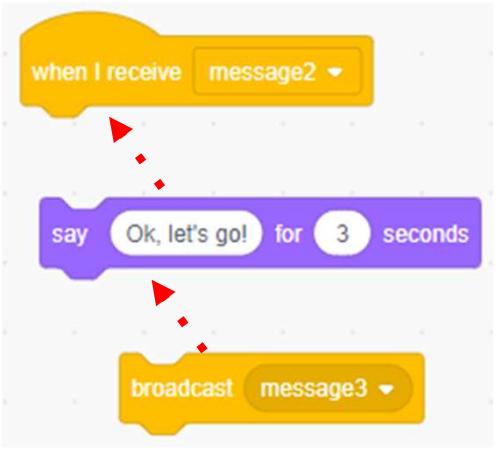
Storytelling

To Code: Complete the Second Dialogue of Scene 1

The next step (3) in the storyboard is Gobo's next statement.

❑ Repeat the step you completed earlier. Add "when I receive" and "broadcast" blocks to Gobo's script. Name the new message as message3.

Gobo



Testing and Debugging

Let's test it again! Did the sprites talk to each other same as you expected?

Remove "Wait" Block → Replace with "Broadcast" and "when I receive" → Complete Scene1 →

❑ Still remember how to add new backdrop you have learnt in the previous unit?

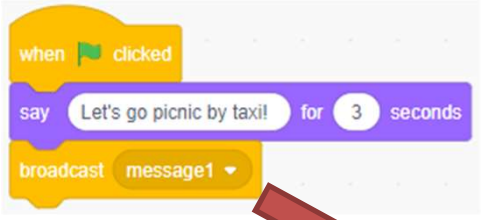
❑ We are going to switch to another backdrop (Scene 2 in Boardwalk) in the next step! We will use "When backdrop switches to ___" to trigger all actions in Scene 2.

"When backdrop switches to ___" is another Event. Event is something that triggers actions.

Storytelling

Before going to Scene 2, let's take a look at how Scene 1 should look like now.

Scene 1

Screen	Gobo	Pico
1 		
2 		
3 		

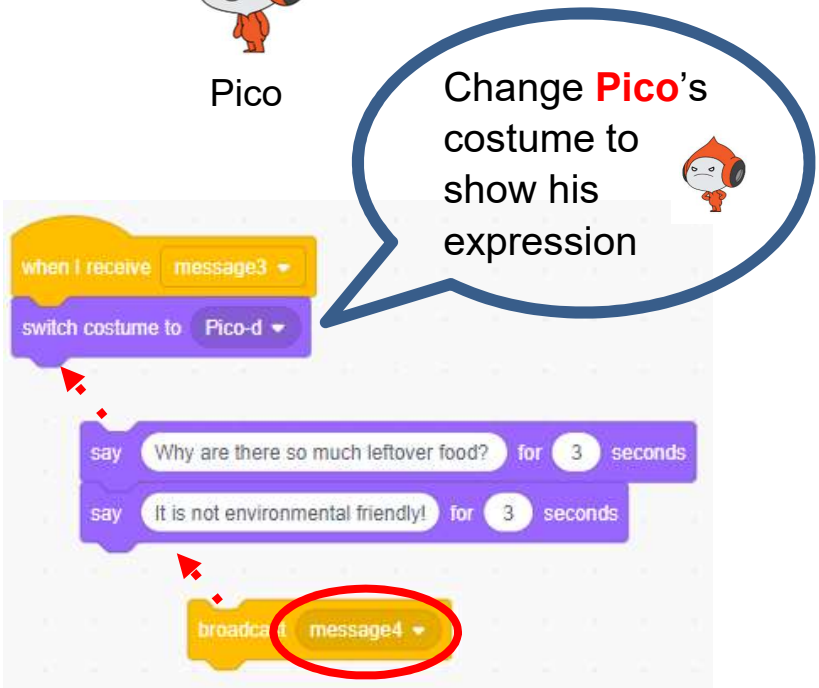
Storytelling

To Code: Complete the Dialogue of Scene 2

Now, let's move on the Scene 2 at Boardwalk to have their picnic!

- ❑ The next step (4) in the storyboard is Pico next statement.
- ❑ Add "when I receive" and "broadcast" blocks to Pico's script. Name the New message as message4.

Pico



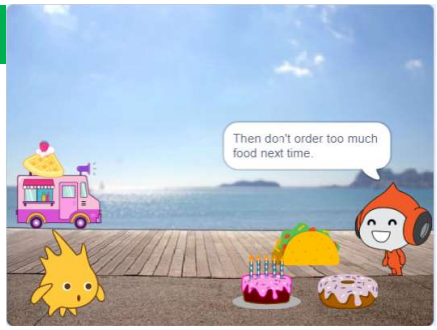
Storytelling

To Code: Complete the Dialogue of Scene 2

- ❑ The next step (5) is Gobo's next statement. Can you complete it (5) and also the rest of them (6) (7)?
- ❑ Try to add some sound at the end to make the story more interesting!

5

6



7

Testing and Debugging

Good job! Click the green flag to see if the story works.

Remove "Wait" Block → Replace with "Broadcast" and "when I receive" → Complete Scene1 → Complete Scene 2

Storytelling

Now Scene 2 should look like …

Scene 2

Screen	Gobo	Pico
4 		when I receive message3 switch costume to Pico-d say Why are there so much leftover food? for 3 seconds say It is not environmental friendly! for 3 seconds broadcast message4
5 	when I receive message4 say But I am full.. for 3 seconds broadcast message5	
6 		when I receive message5 switch costume to pico-b say Then don't order too much food next time. for 3 seconds broadcast message6
7 	when I receive message6 say We are all eco-warrior! for 3 seconds start sound Cheer	when I receive message6 say We are all eco-warrior! for 3 seconds start sound Cheer

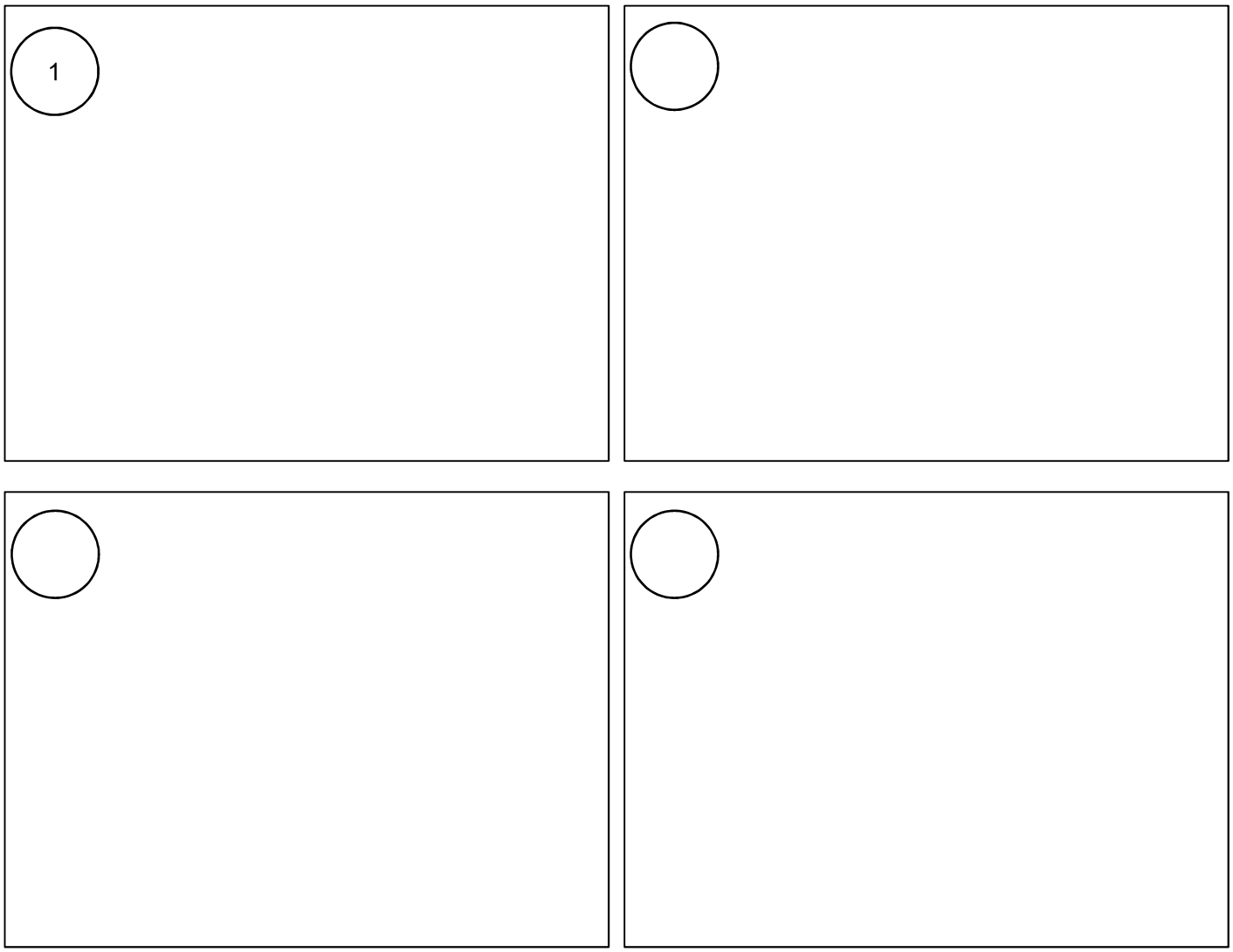
Storytelling

To Create: Storyboarding

Task 3:

Time to design your own story with your own theme! (e.g. helping to improve the living of the people in need, recycling for protecting our environment etc.)

- ✔ Number the dialogues in the order they happen (1, 2, 3, 4…)
- ✔ Try to think about the aesthetic aspects (For example, the colour and position of your sprites, how they can form the colour harmony with the background…)
- ✔ Try to think about the theme and write down your ideas below:
  1. Describe the costumes/ motion of your sprites in the story.
  2. Introduce the design of using different backdrops



Storytelling

To Create: Code and Tell Your Story

❑ Sign into your account at scratch.mit.edu.

❑ Create a new project named Storytelling.

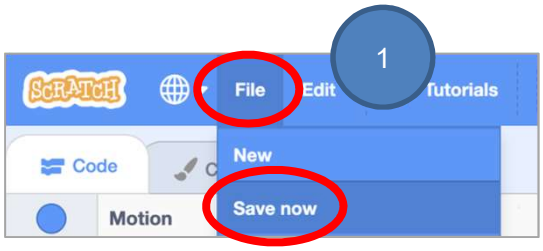
❑ Based on the storyboarding you made, start to create your own story with Scratch.

❑ Make use of "Broadcast" & "When I receive block" to complete the conversation.

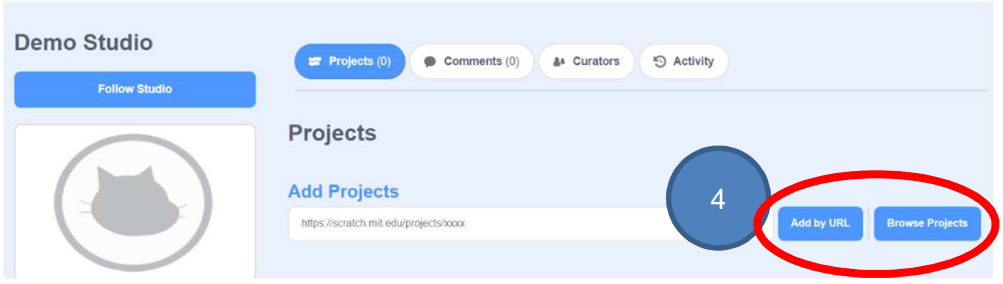
Sharing to Studio and Provide Constructive Feedback on Program Design

When you finish, you will add your project to your teacher's Studio.

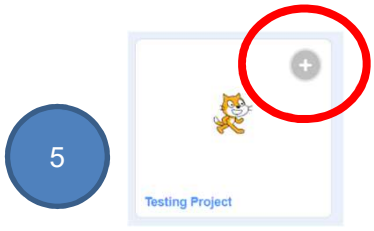
1. Save your project by clicking "Save now" under the File menu.
2. Click the orange "Share" button.
3. Go to your teacher's Studio (they will give you a URL).
4. In the "Add projects" column, you can Add by URL or Browse Projects.



3



5. If you choose Browse Project, then you will see all your shared projects. Find the right one and click the "+" to add it to studio.



Storytelling

To Reflect: Two Stars and a Wish Worksheet

Name of Project: _____ Name of Creator: _____

Please write down two things that you like about this project.

1

_____

2

_____

What is one thing you would like to add or change to make this project better?

_____

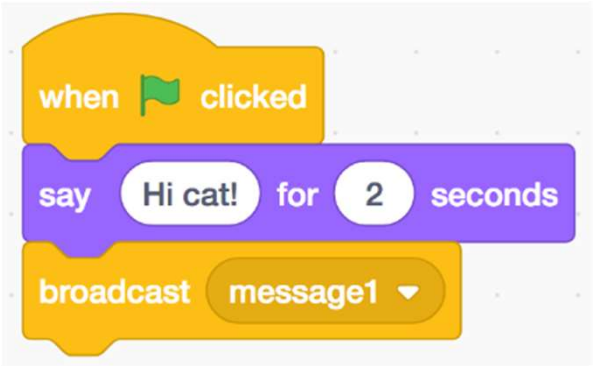
_____

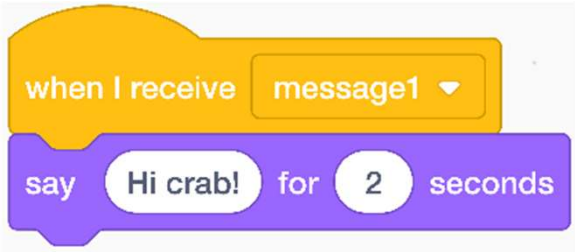
_____

Storytelling

Review Questions

1. For the blocks below, what happens when you click the green flag?



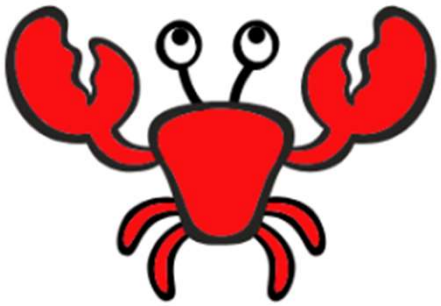


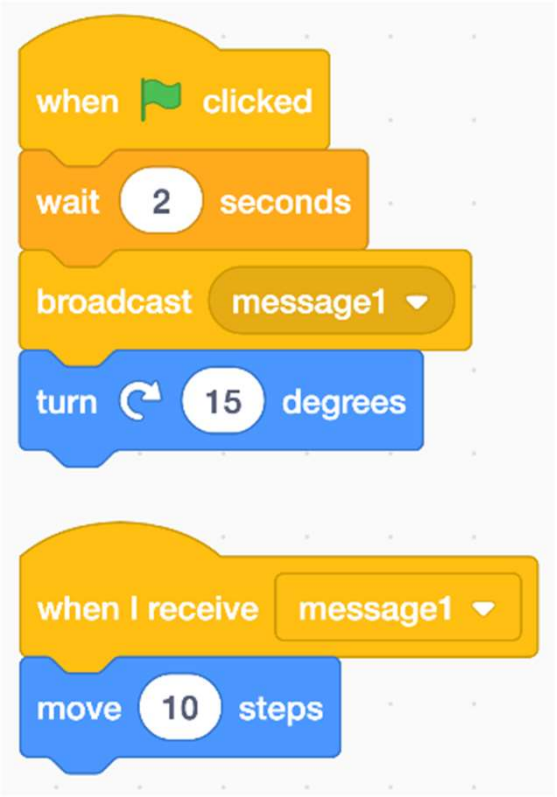
A. The crab says “Hi cat!” for 2 seconds. The cat does nothing.
B. The cat says “Hi crab!” for 2 seconds. The crab does nothing.
C. The crab says “Hi cat!” and the cat says “Hi crab!” at the same time for 2 seconds.
D. The crab says “Hi cat!” for 2 seconds, and then the cat says “Hi crab!” for 2 seconds.

Storytelling

Review Questions

2. What does the crab do when the green flag is clicked?

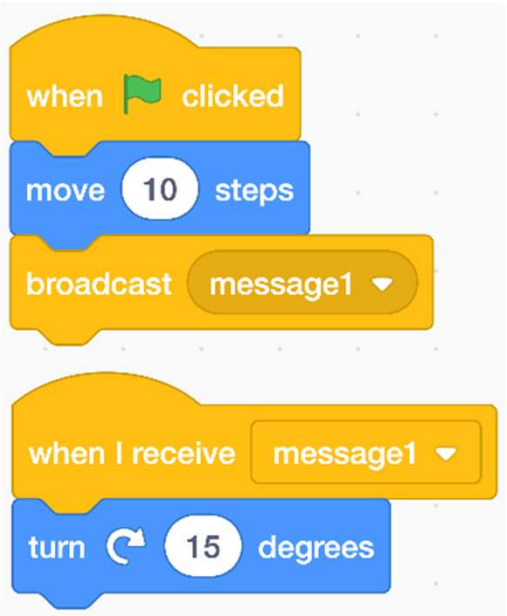




A. After a 2-second delay, the crab moves 10 steps and turns 15 degrees.
B. After a 2-second delay, the crab moves 10 steps.
C. The crab moves 10 steps, waits 2 seconds, and then turns 15 degrees and moves another 10 steps.
D. The crab moves 10 steps, waits 2 seconds, and moves another 10 steps.

Review Questions

3. What does the cat do when the green flag is clicked?

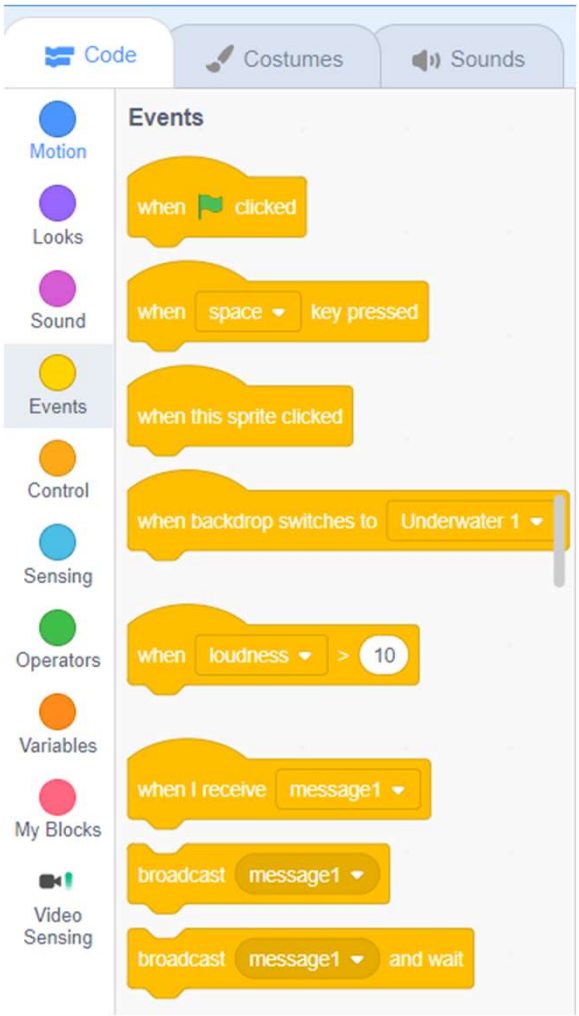


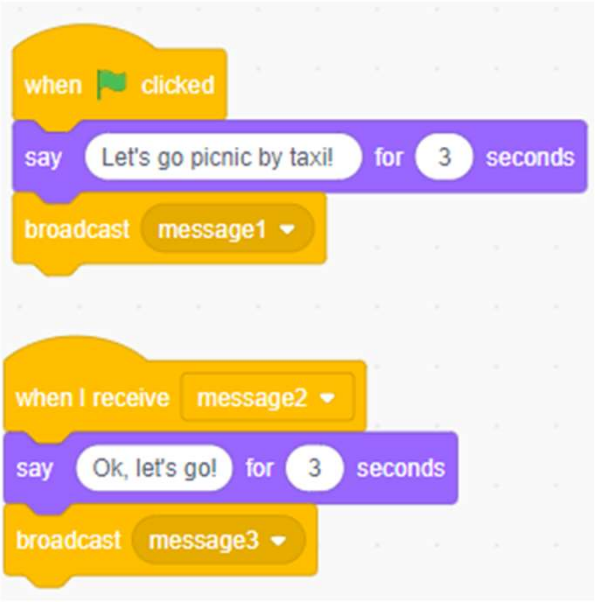
A. The cat moves 10 steps and turns 15 degrees, waits 2 seconds, and then turns 15 degrees more.
B. The cat moves 10 steps and turns 15 degrees, waits 2 seconds, and then moves another 10 steps and turns another 15 degrees.
C. The cat moves 10 steps and turns 15 degrees.
D. The cat turns 15 degrees.

Storytelling

Revision on Key Features

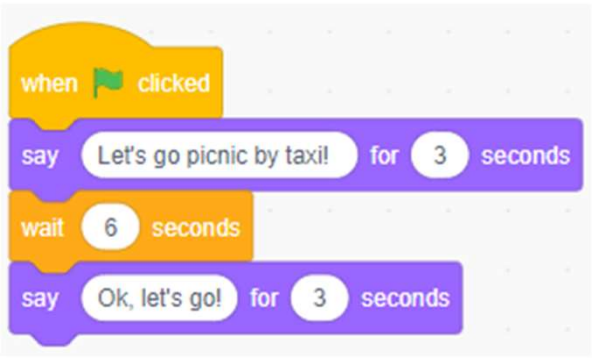
- **Broadcast (message)**
- **When I receive (message)**

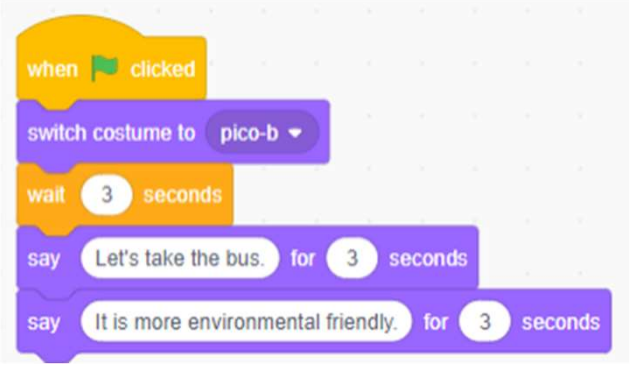




Revision on Key Concepts & Practices

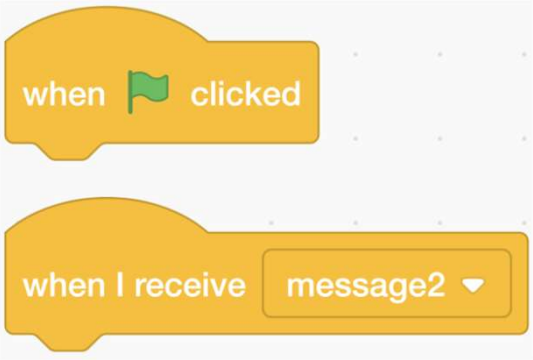
Sequences: It is a key concept in programming. It is the order in which the programming statements are executed. A wrong order would lead to incorrect programming results. For example, there is a specific sequence of dialogues for a meaningful conversation to go on.



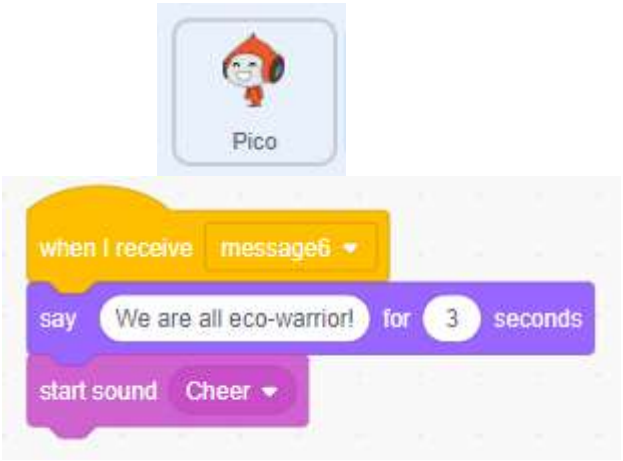


Revision on Key Concepts & Practices

Events: We use event blocks to trigger Scratch to take actions.



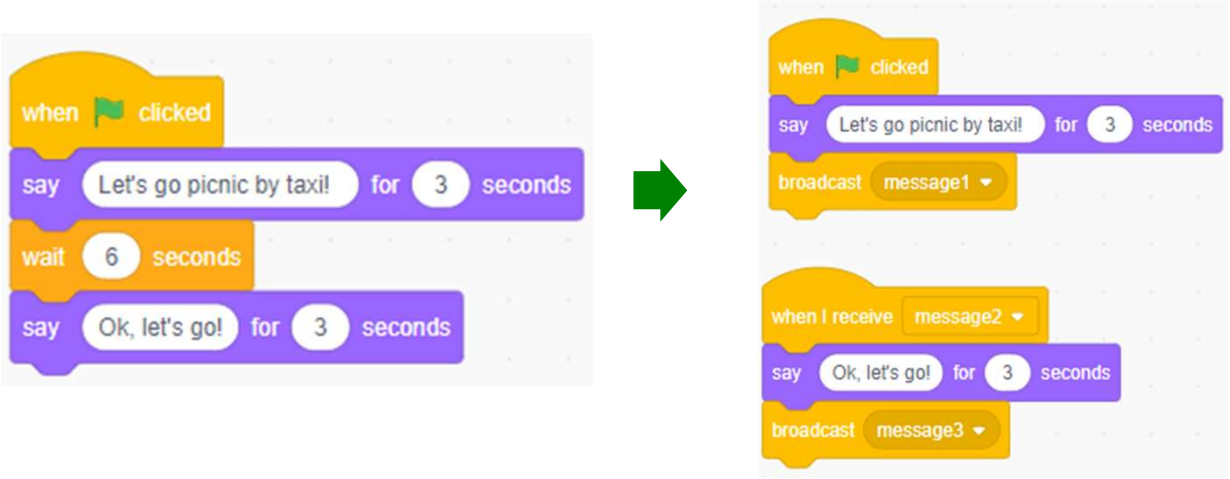
Parallelism: Scratch uses parallelism to allow more than one events to take place at the same time. For example, Scratch allows more than one character to perform an action at the same time in a Scratch program.



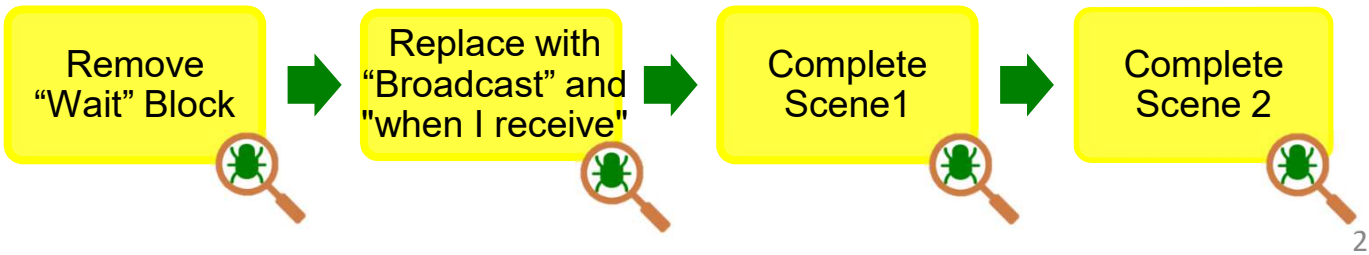
Storytelling

Revision on Key Concepts & Practices

Being incremental and iterative: to work out a sub-task as an iteration, try it out, then work out another sub-task based on the codes of the previous sub-task in one more iteration until the whole programming task is completed.

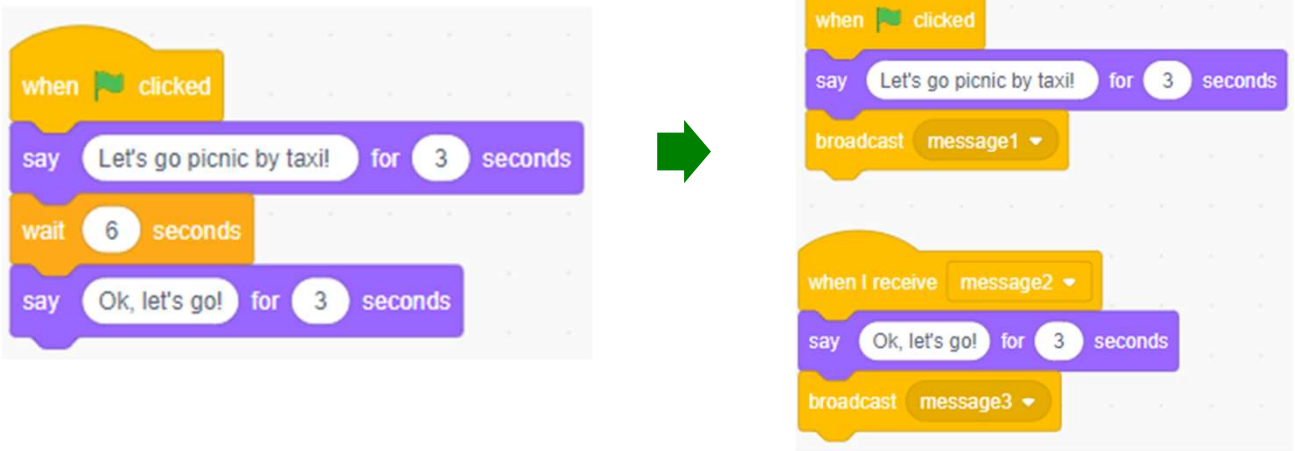


Testing and debugging: Testing a computer program is the process of checking if it can produce outcomes as designed. Debugging a computer program is the process of finding out ways to revise the program so that the bugs can be removed.



Revision on Key Concepts & Practices

Reuse and Remix programs/codes: The reuse and remix of the works of other programmers are crucial in the online communities of Scratch. For example, we can reuse and remix the codes of a sprite such as the shark in Unit 2 and use them for the second and third sprites.



Unit 4: Space Travelling
Student Guide

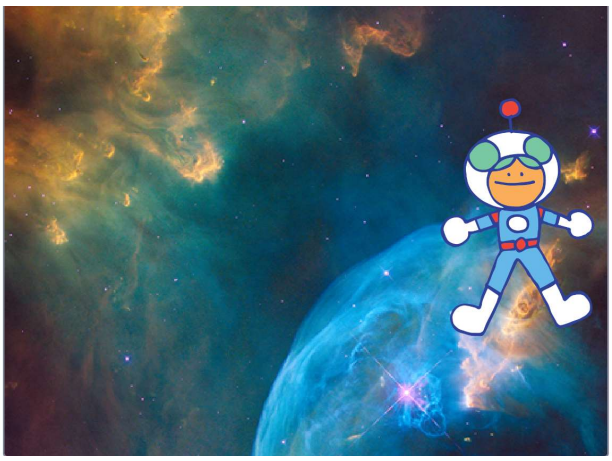
Content

Lesson 3

Space Travelling

Let's learn how to travel in space with Scratch!

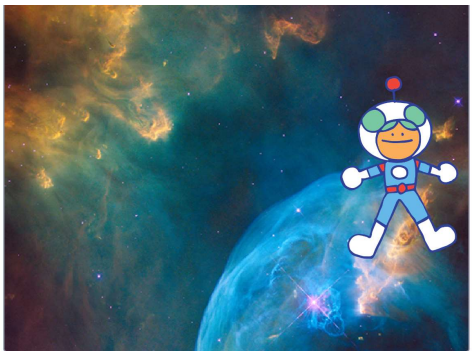
Through this trip, you will learn how to change the costumes of the sprite using the video sensing feature in Scratch.

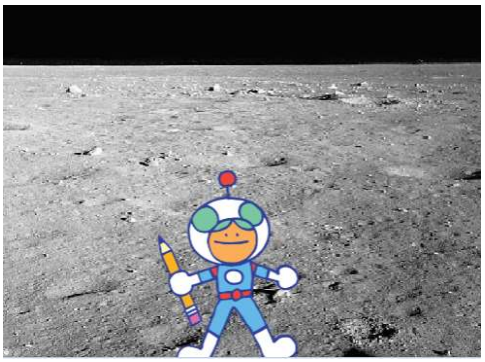


To Play

Play the Space Travelling (Demo): https://scratch.mit.edu/projects/727401089.

Try to move your hand in front of the webcam of your computer.

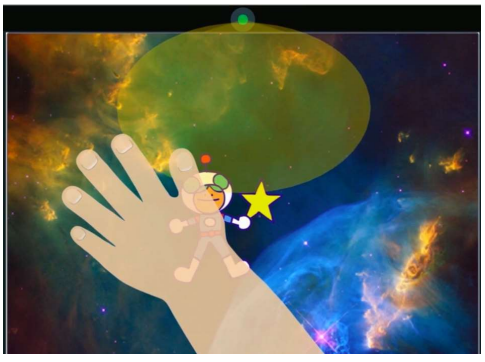




What happened to the sprite and backdrop?

How fast did you wave your hand?

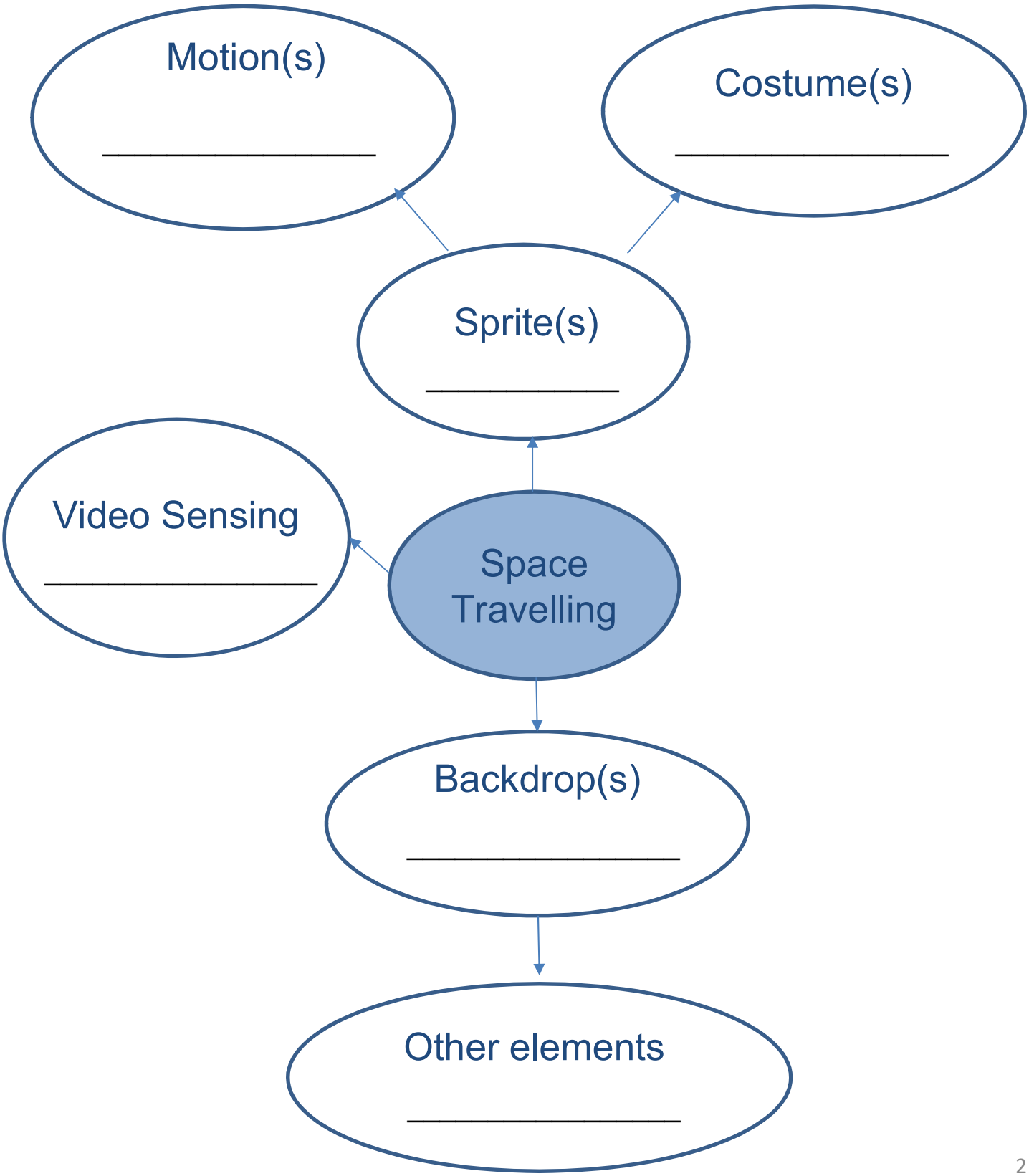
Did you hear any sound effect?



Space Travelling

To Think

Complete the mind map below before your start to code.

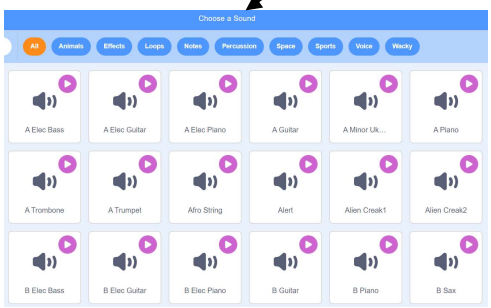


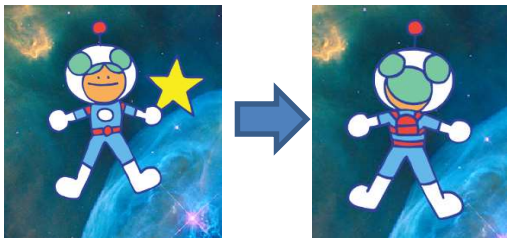
Space Travelling

To Think

What happened when you wave your hand in front of the camera?
Can you fill in the actions triggered below?

(A) Play Sound Effect	**(B)** Webcam detecting your movement
(C) Change Costumes	**(D)** Change Backdrop





Space Travelling

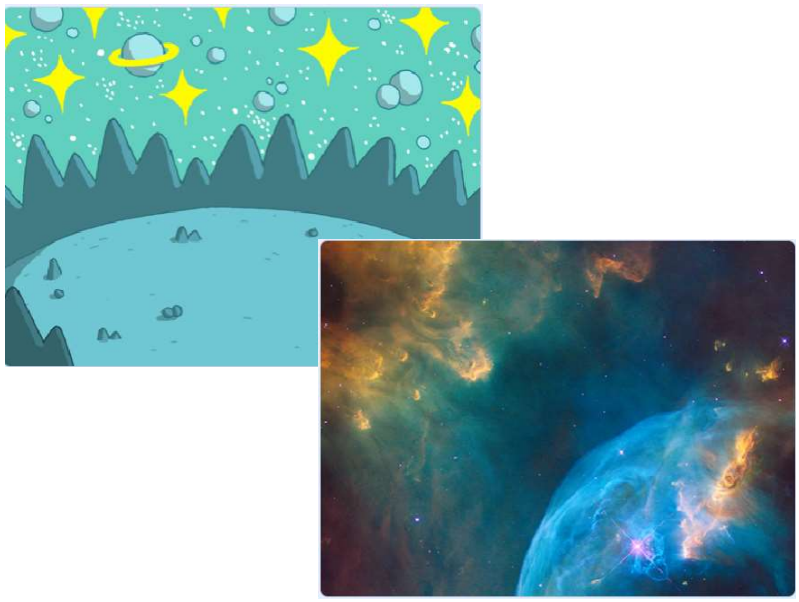
To Code: Adding the Backdrop and Sprite

1. Turn on the webcam of your computer.

See Appendix P.35-36

2. Sign in and create a new project. Name it Space Travelling.

3. Choose a good backdrop.



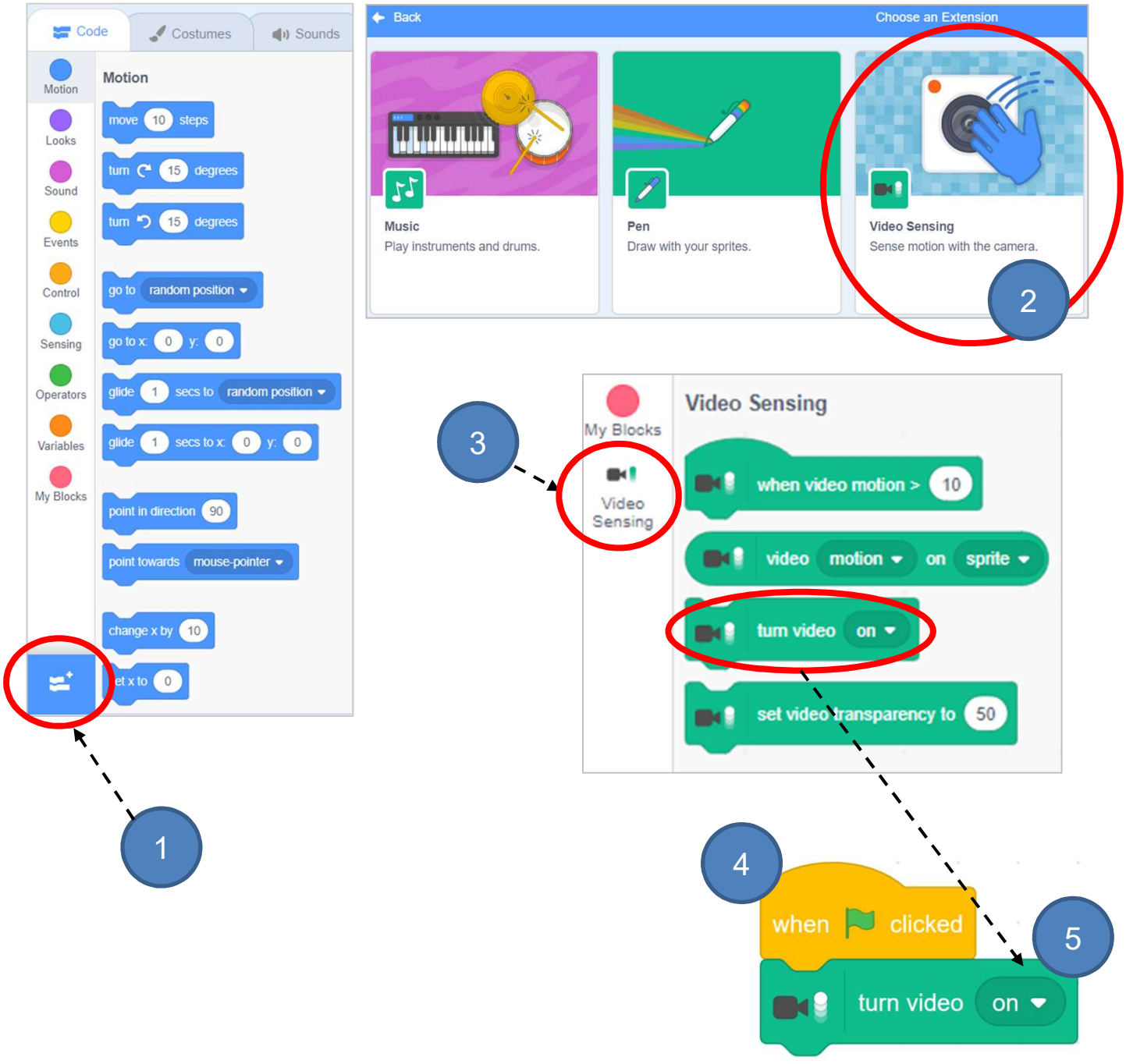
4. Add a sprite and some different costumes for it.

Space Travelling

To Code: Add Video Sensing

Let's find the video sensing blocks for space travelling!

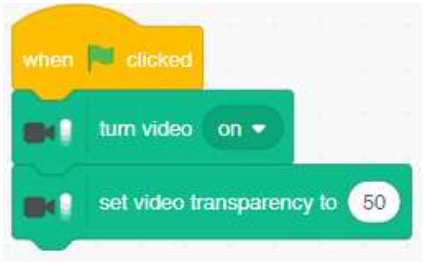
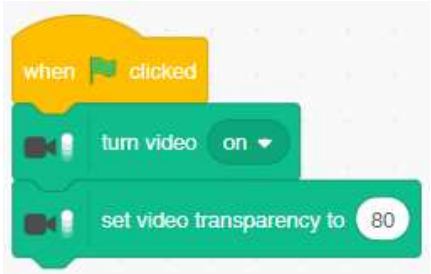
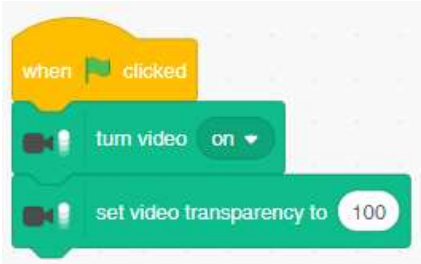
1. Click on the "Add Extension" icon at the bottom left of the page.
2. Choose the "Video Sensing" feature.
3. You will see a list of Video Sensing blocks in green.

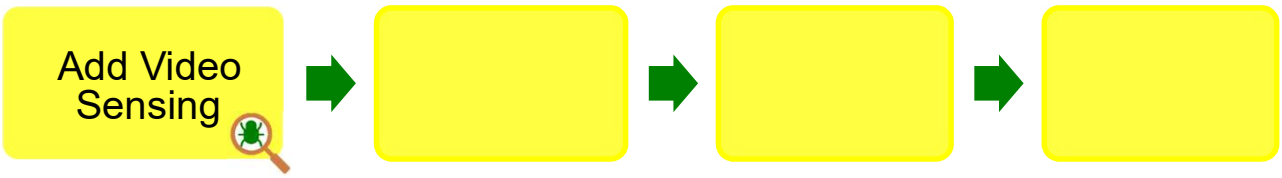


Space Travelling

To Code: Add Video Sensing

Let's try these three events and see what will happen?

1	2	3
		



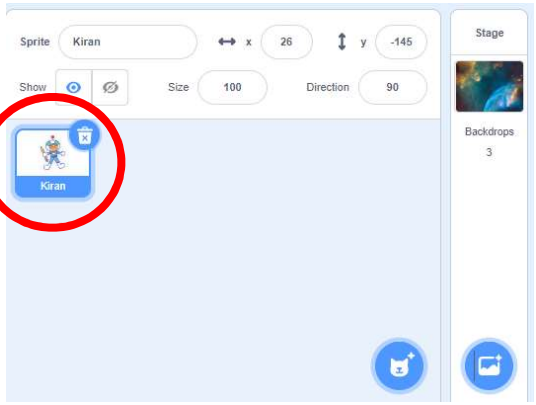
Testing and Debugging

Let's test it! Click the green flag.
Try to adjust the video transparency for your own project!

Space Travelling

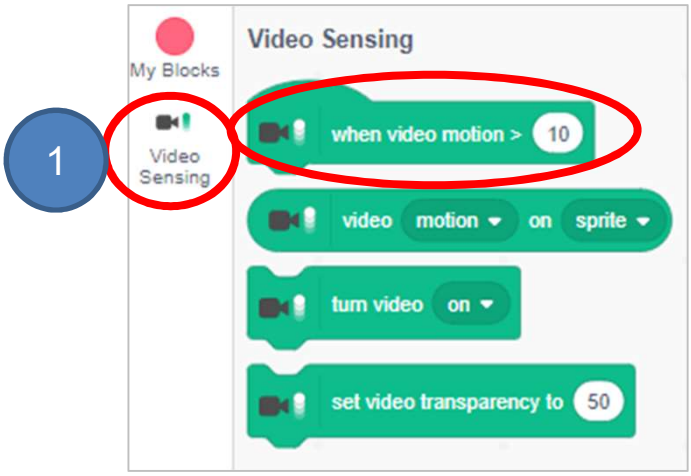
To Code: Change Costumes

Click the "Sprite", start to code for the sprite!

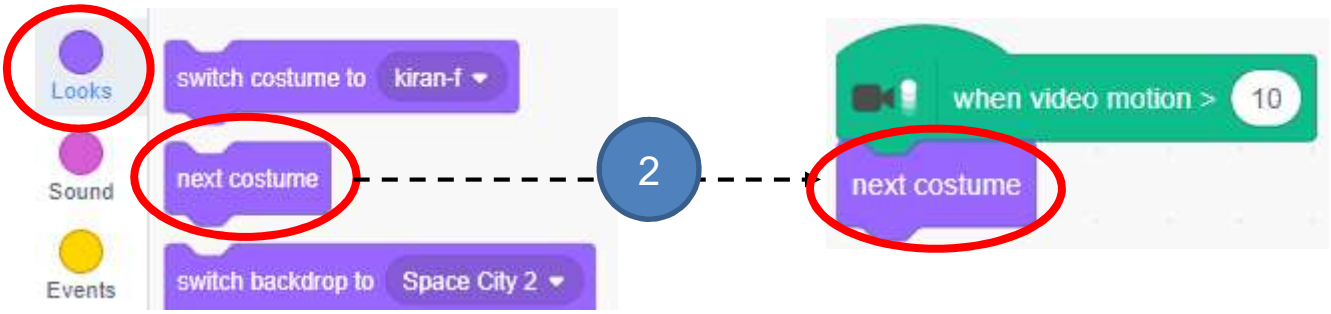


We want the sprite to do the action when the video motion meets the threshold.

1. Drag out a "when video motion > 10" block.



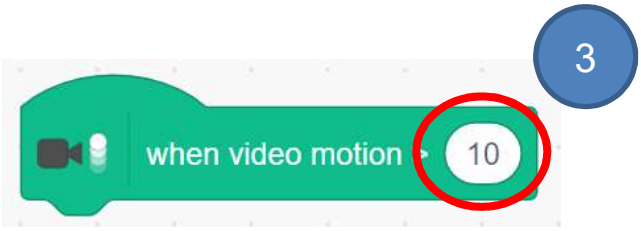
2. To change the costumes of your sprites.



Space Travelling

To Code: Change Costumes

3. Try to change the value of video motion level (e.g. 10, 30, 100 or other values). Wave your hand in front of the camera to see the difference. (Hint: 1 will start with very little movement, 100 requires a lot of movement.)



Please circle your observation below:

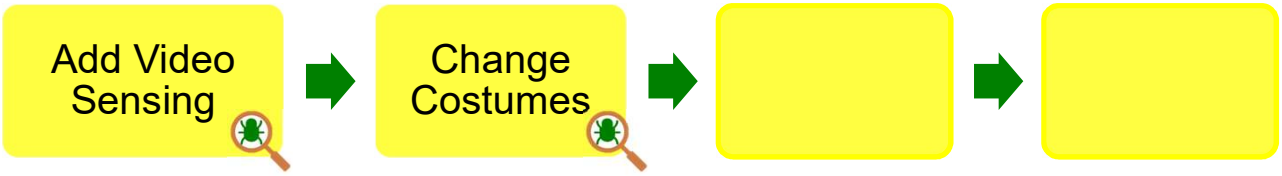
Value of Video Motion	e.g. >60	>30	>100	
What if you move your hand slower?	Costume changes / does not change	Costume changes / does not change	Costume changes / does not change	
What if you move your hand quicker?	Costume changes / does not change	Costume changes / does not change	Costume changes / does not change	

Space Travelling

To Code: Change Costumes

Testing and Debugging

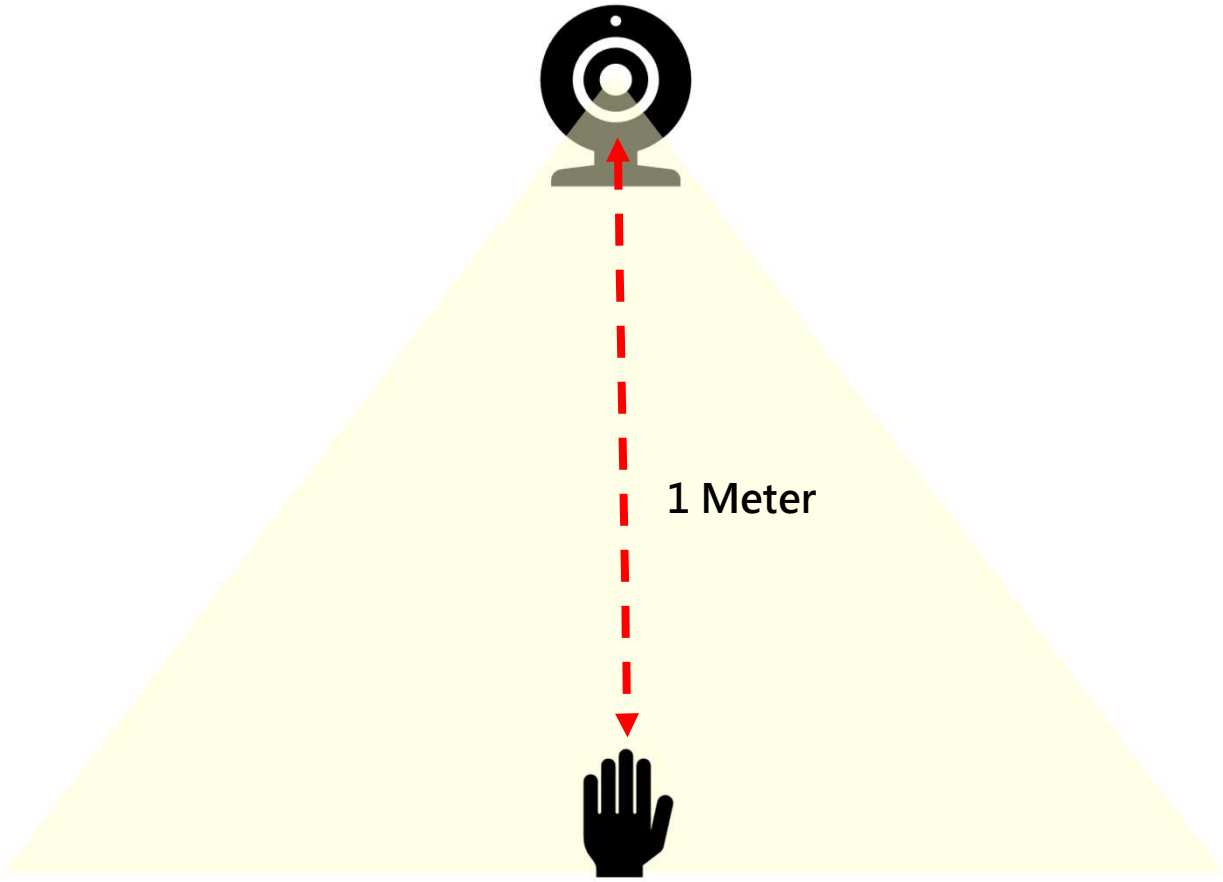
Let's test it!
If you enter a smaller value for the sprite, try to move your hand slowly.
Did you change it to the next costume?



Space Travelling

Relationship between Video Sensing and Video Motion

Place your hand 1 meter away from the webcam as follows:

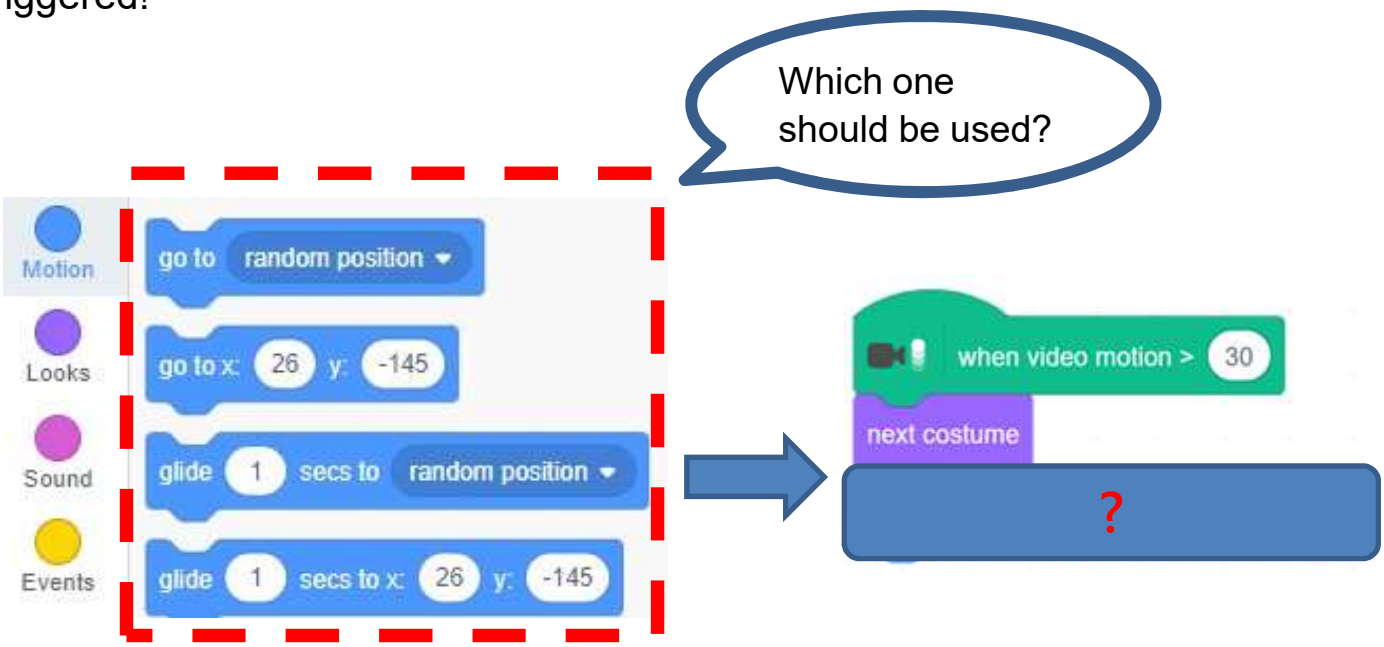


Move hands quicker or slower in front of the webcam to see what happens.

Space Travelling

To Code: Glide in the Space

Try to make your sprite glide to random position when the video motion is triggered!



Testing and Debugging

Wave your hand to test the project, see if the sprite changes its costume and glides to random position.

Add Video Sensing → Change Costumes → Glide in the Space →

Space Travelling

To Learn

Unplugged Activity: Guessing the Right Number

How to play?

A teacher thinks of a number between 1-25. Have some students try to guess the number by asking a series of questions. The teacher reacts according to students' questions.

Example:

Teacher selects (20).

Student: If the number is greater than (>) 13, please raise your hand.

Teacher: (Raises his/her hand)

Student: If the number is greater than (>) 19, please raise your hand.

Teacher: (Raises his/her hand)

Student: If the number is greater than (>) 22, please raise your hand.

Teacher: (No action)

Student: If the number is greater than (>) 20, please raise your hand.

Teacher: (No action)

Student: If the number is less than (<) 20, please raise your hand.

Teacher: (No action)

Student: If the number is equal to (=) 20, please raise your hand.

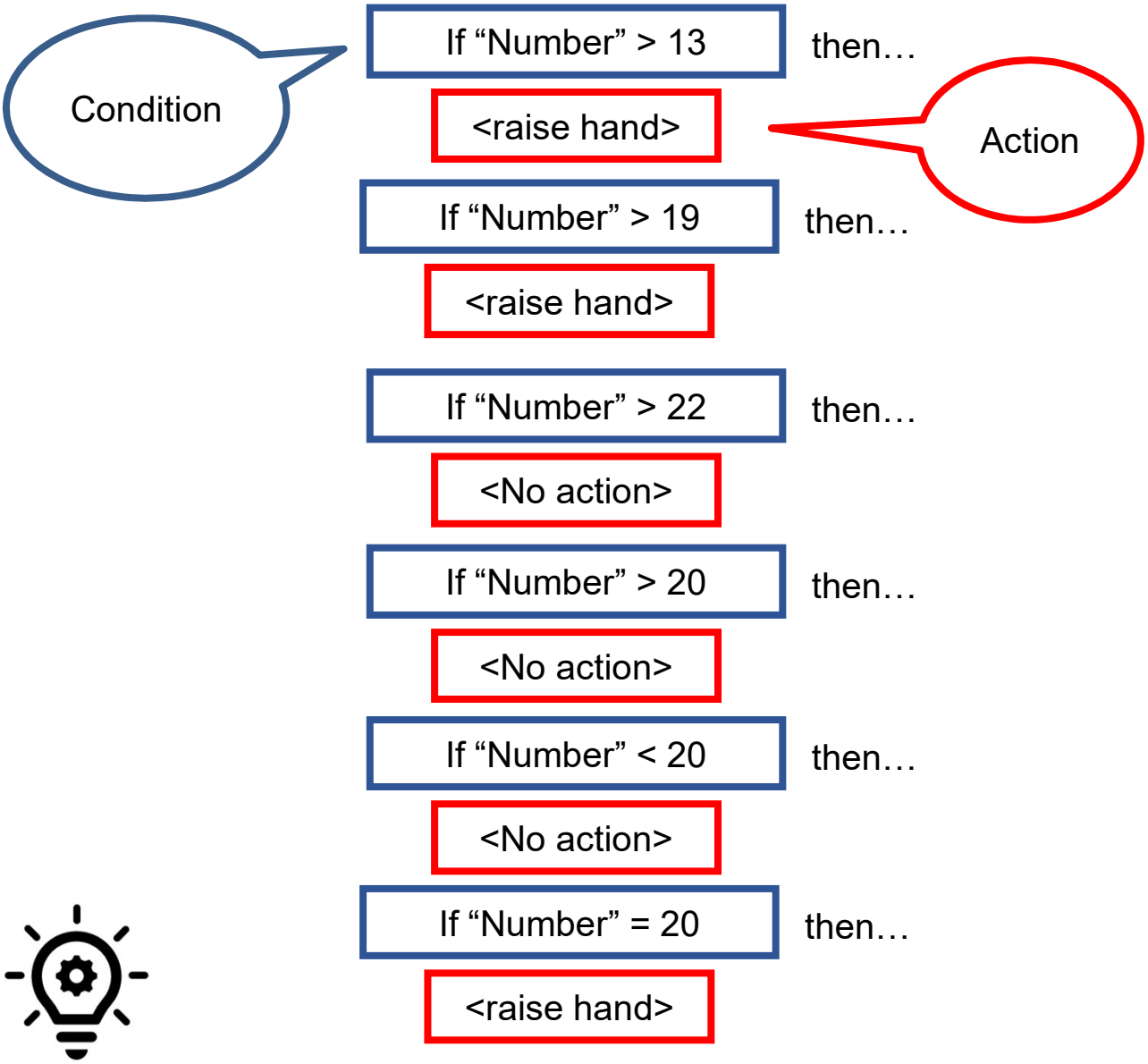
Teacher: (Raise his/her hand).

Space Travelling

To Learn

Unplugged Activity: Reflection

In the game, you learn to check your guess with a conditional expression and the teacher reacts based on the condition you provided in the question. Do you remember the conditional expressions made?



Knowledge builds up: Branching / Selection / Conditionals

We use conditional statements in programming to enable computers to make decisions. Conditionals always have an “if” part, which tells the program in the “then” part what to do when the condition is true.

Space Travelling

To Learn

Conditional Operators	Meaning
>	Is greater than
<	Is less than
=	Is equal to

Knowledge builds up: Conditional Operators

Conditional Operators: We use operators to evaluate whether a condition is true or false. Conditional expressions always use operators such as greater than (>), less than (<) or equal (=).

Question:

There is a height requirement for roller coasters at a theme park for safety reason. You need to be taller than 112cm in order to play those roller coasters.

Which **conditional operator** can be used to express this height requirement?

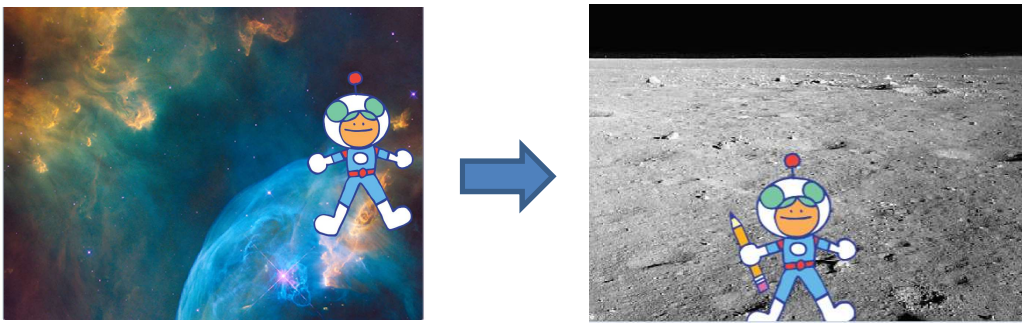
A. Height **>** 112 cm

B. Height **<** 112 cm

C. Height **=** 112 cm

Space Travelling

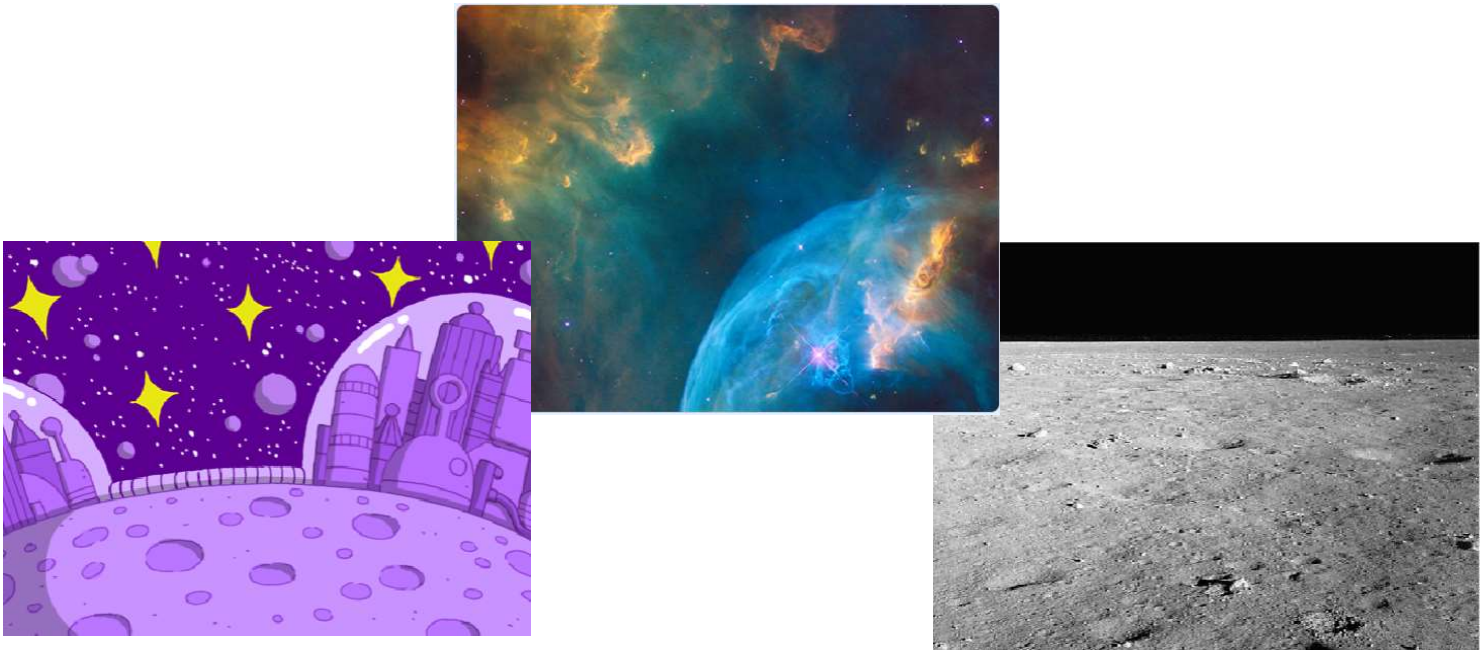
Let's continue to travel in space with Scratch!
We will travel to other planets when we move our hand fast.



To Code: Change Backdrop

See Appendix P.37

Try to add a total of 3 backdrops for your space travelling!

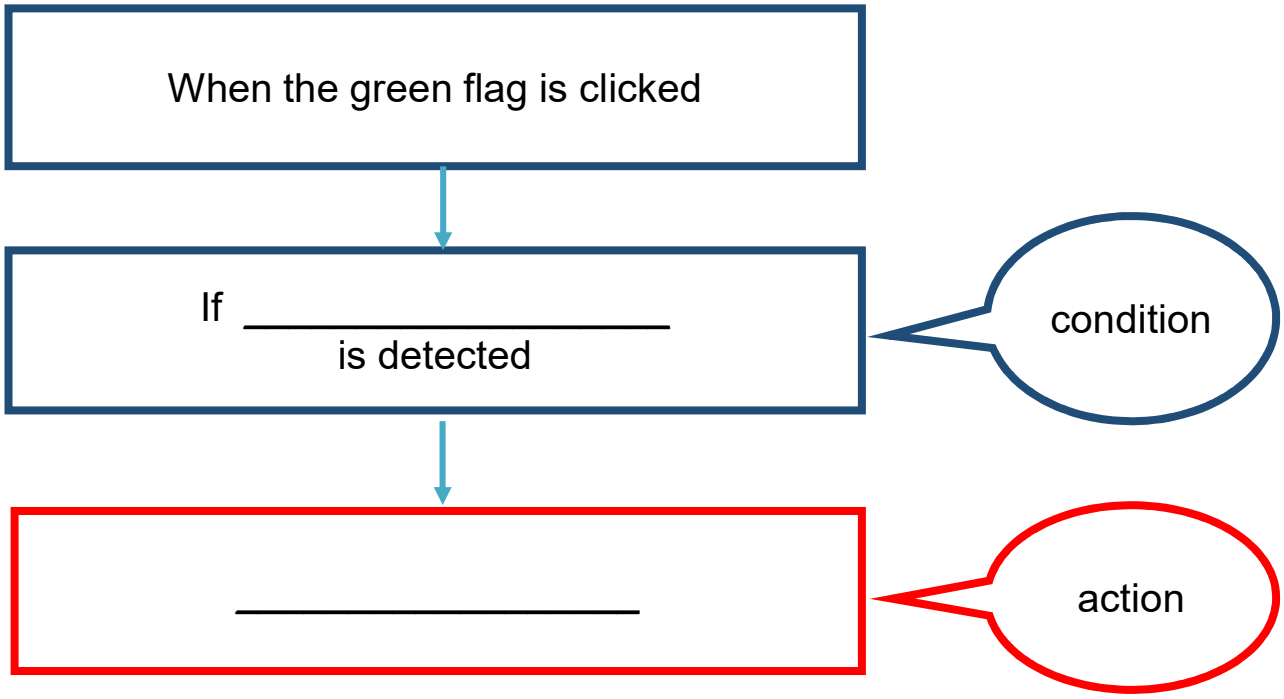


Space Travelling

To Code: Change Backdrop

Now, you can see a set of backdrop in backdrops.

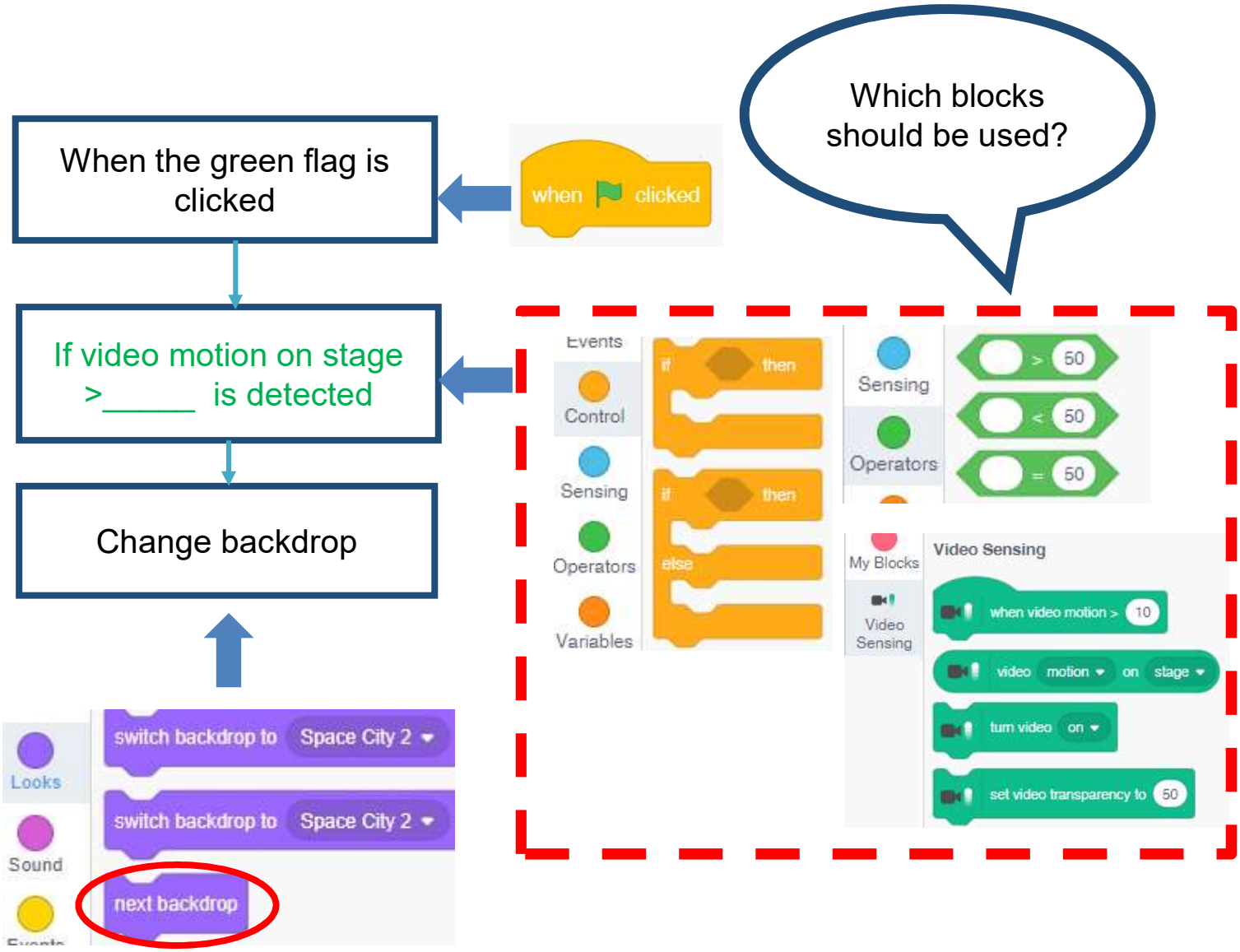
What is the sequence about changing Backdrop of this project? Please fill in the Flowchart.



Space Travelling

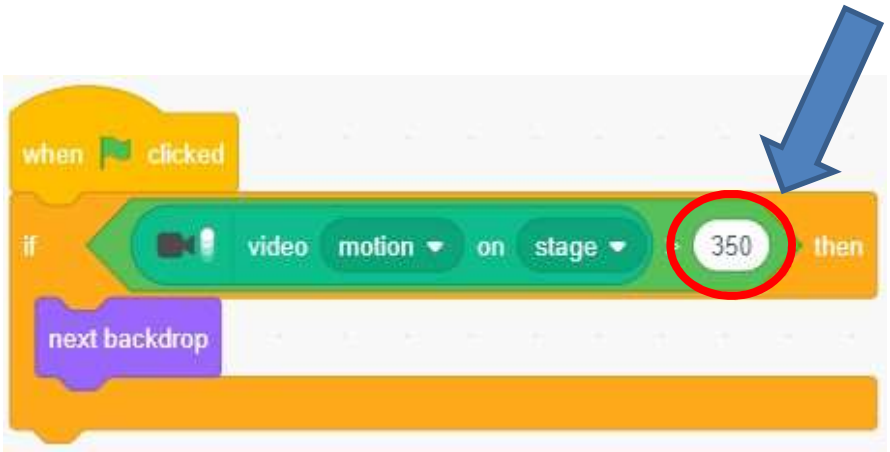
To Code: Change Backdrop

Let's have a look at the codes! Which blocks should be used to trigger the actions we want?



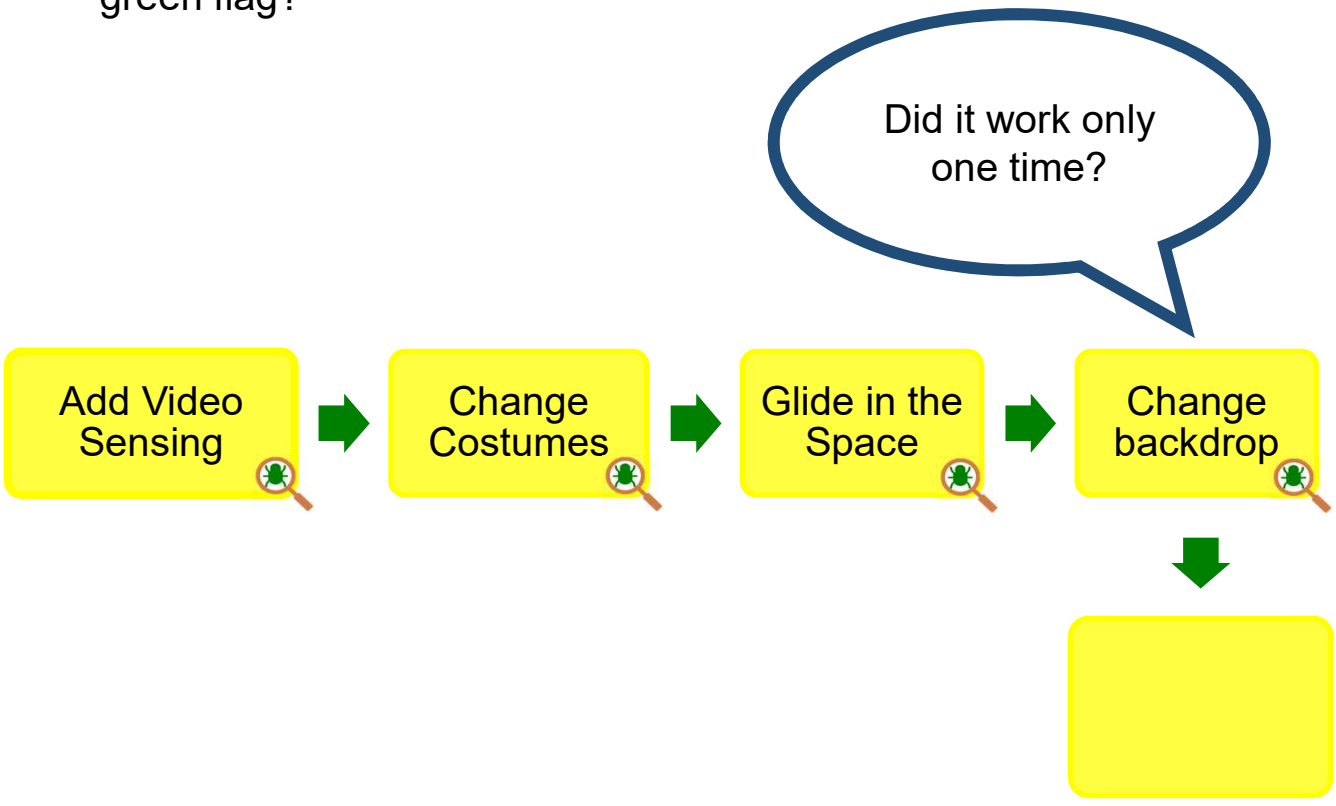
Space Travelling

Try different values and see what happens! Try to enter 100? 200? Or more?



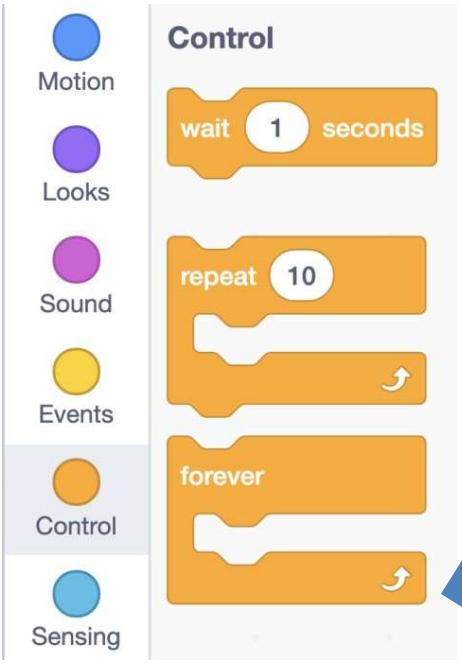
Testing and Debugging

Let's do the testing. What happened after you clicked the green flag?

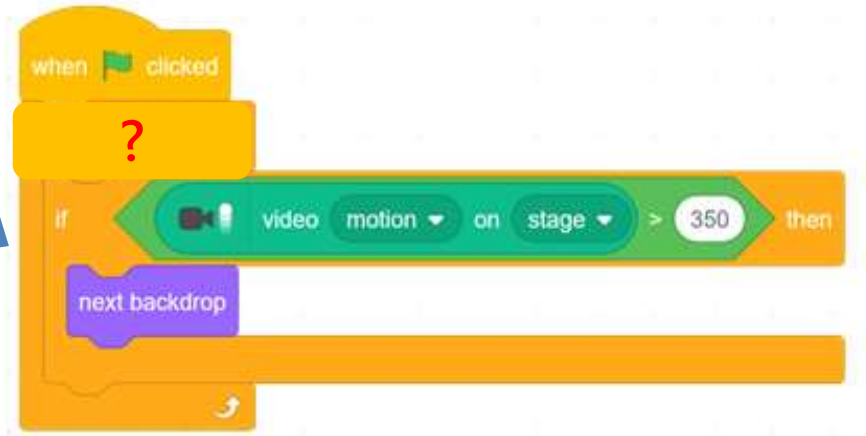


Space Travelling

To Code: Iteration

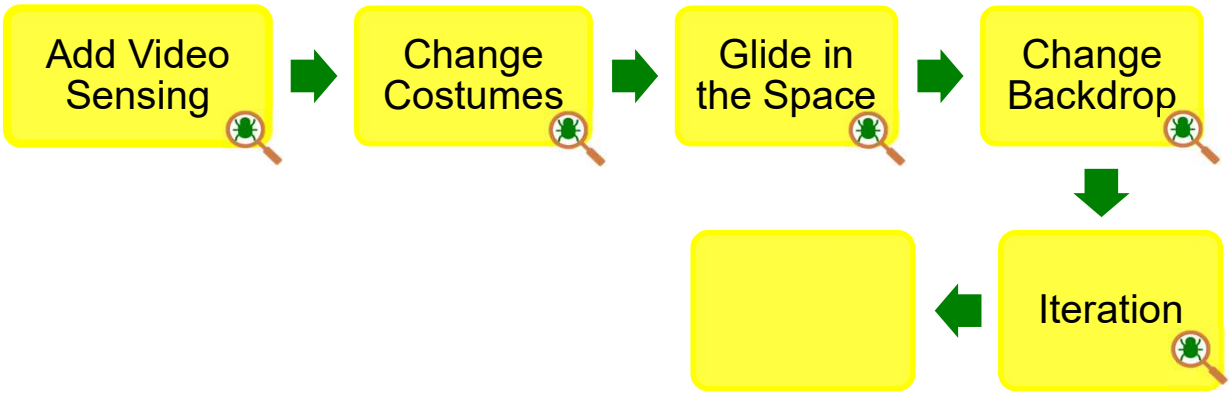


Which block should be added to continuously check for the condition?



Testing and Debugging

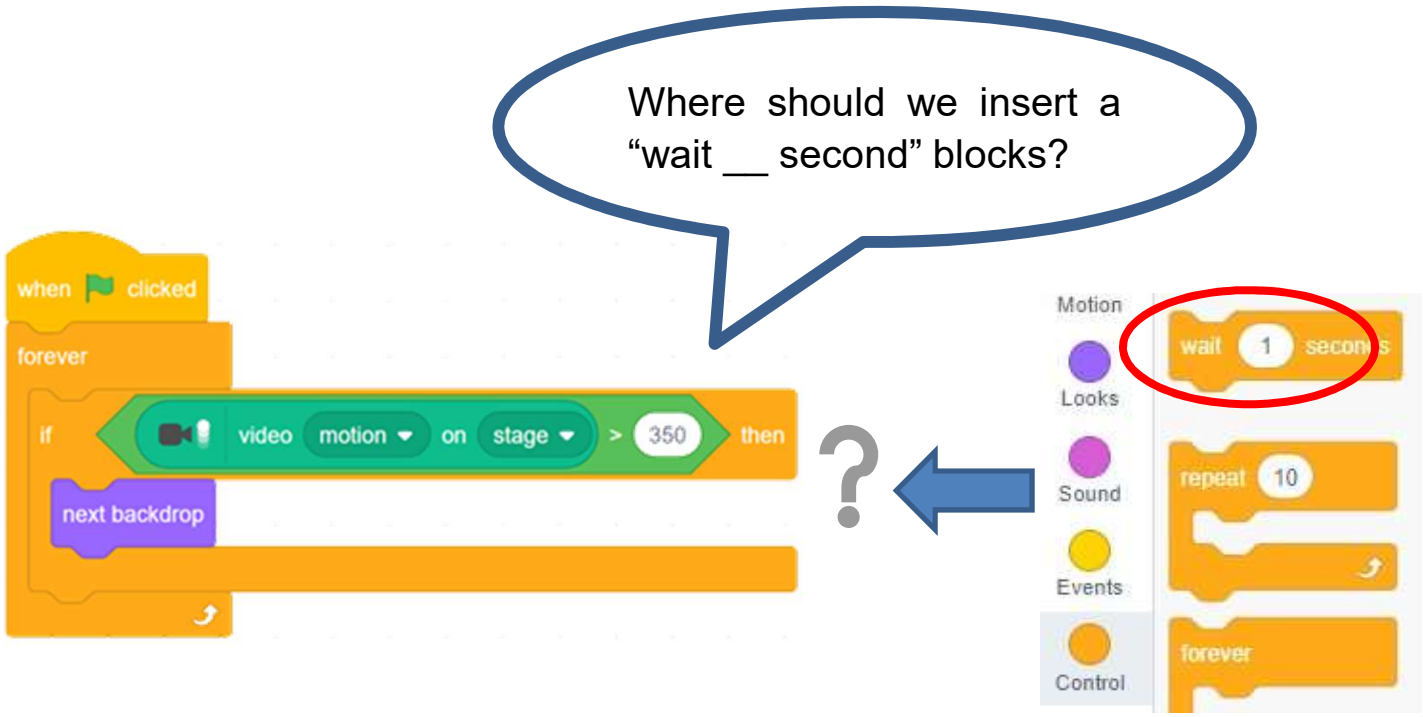
Test it again. Is the changing speed too fast or slow?
How to solve this issue?



Space Travelling

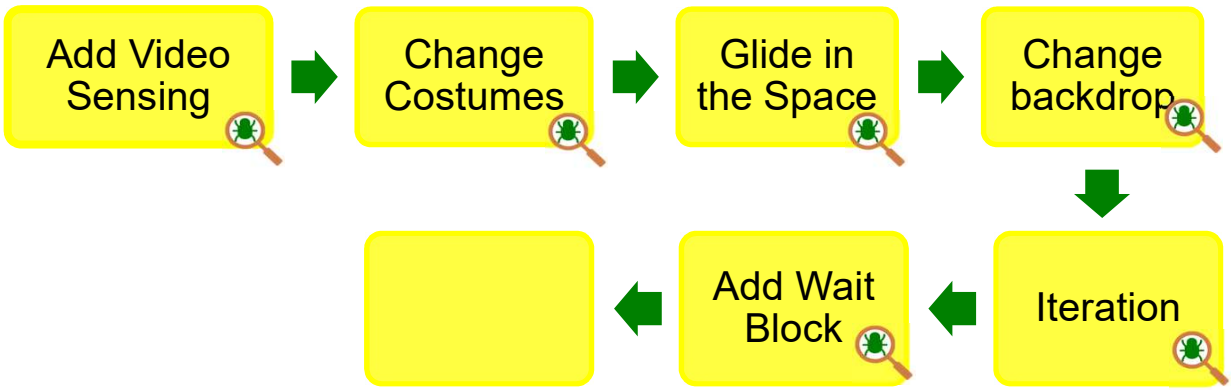
To Code: Add Wait Block

Where should we insert a “wait __ second” blocks?



Testing and Debugging

Can’t wait to see the changes? Let’s test it now!



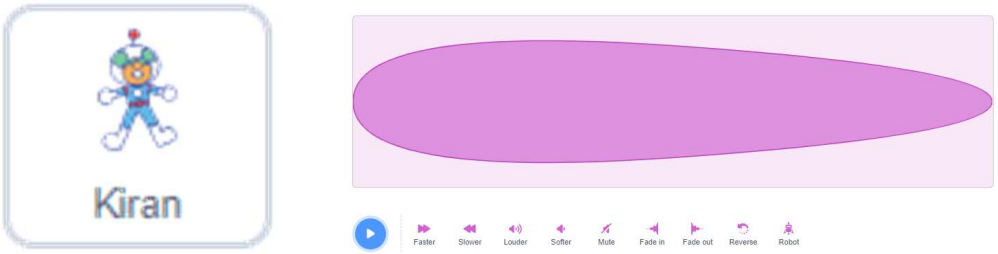
Try different value in “wait __ second”.

Space Travelling

To Code: Add Sound Effect (Sprite)

Remember how to add the sound? Try to make the sprite play a sound based on the video motion.

See Appendix P.38-39

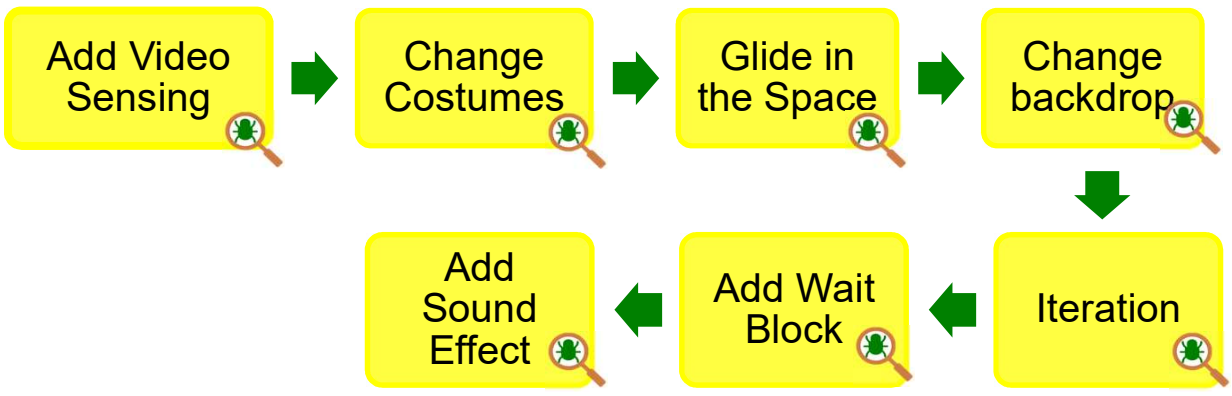


To Code: Add Sound Effect (Backdrop)

Repeat the previous step to make the backdrop play a sound based on the video motion.

Testing and Debugging

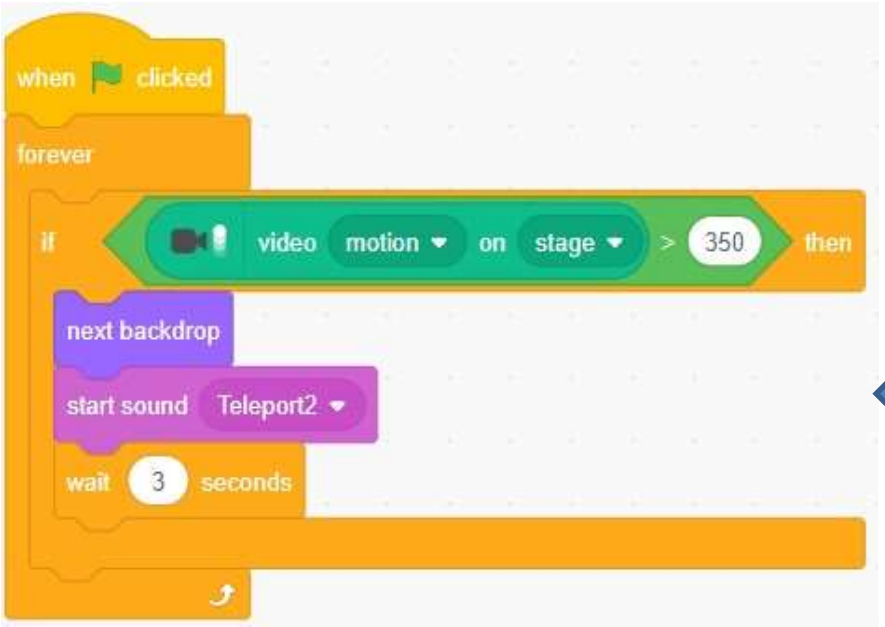
You can test anytime! Wave your hand and see if you hear the sound effect you just added.

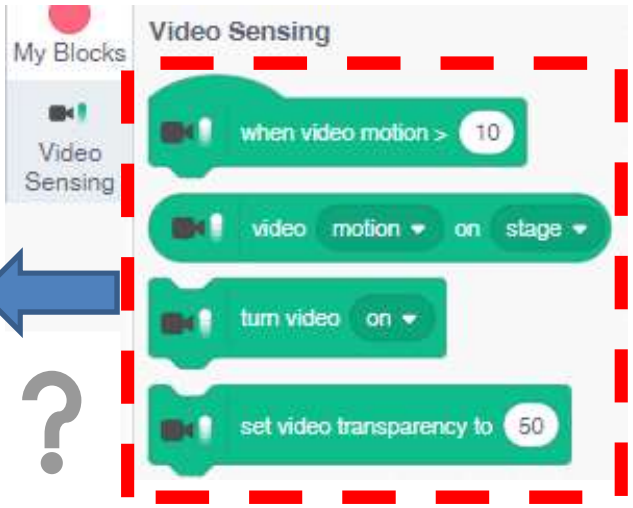


Space Travelling

To Reflect

Which one has the same effect with this event?





Space Travelling

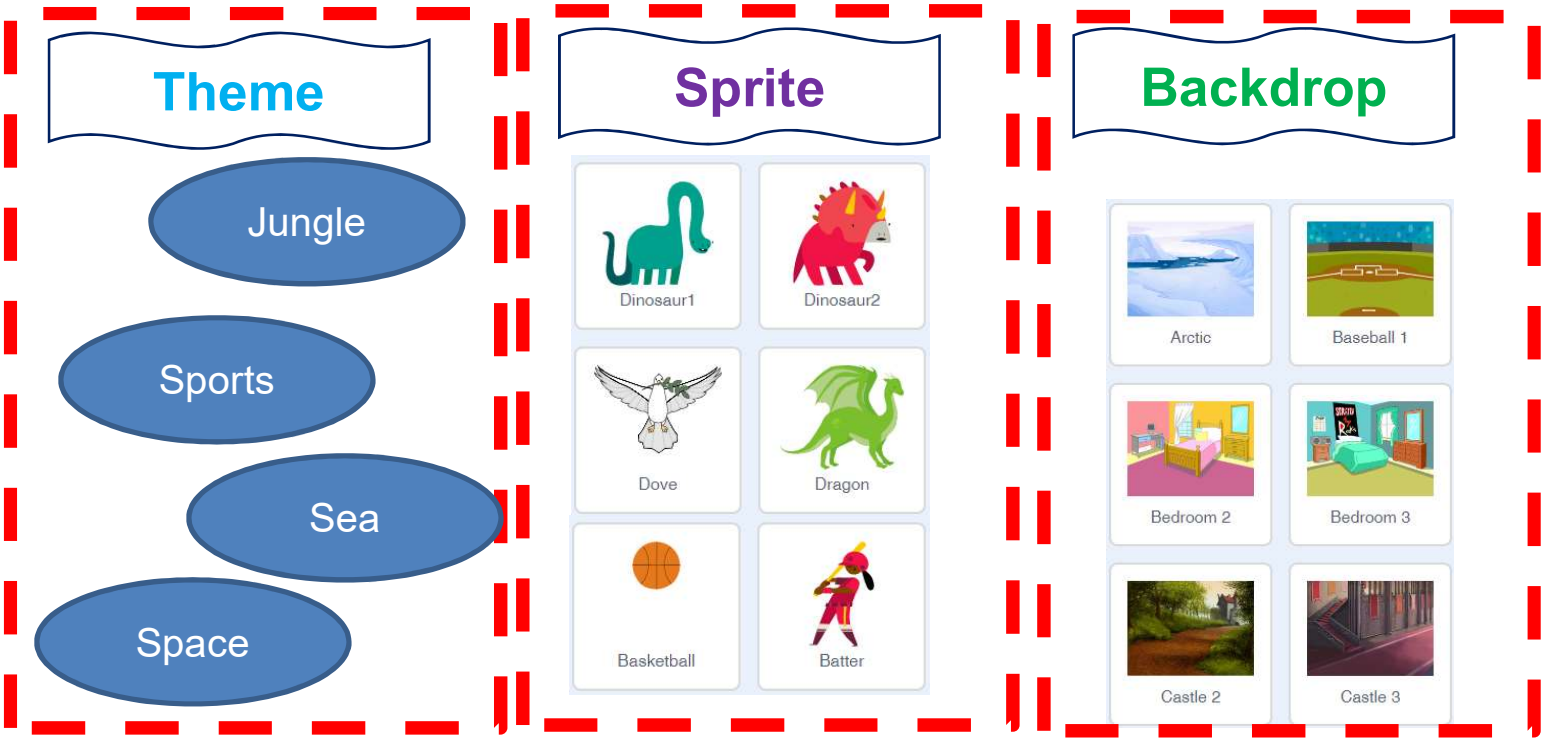
In this lesson, you will create your own story by using Video Sensing feature! What story would you like to create?

To Create: Make your own project

Task:

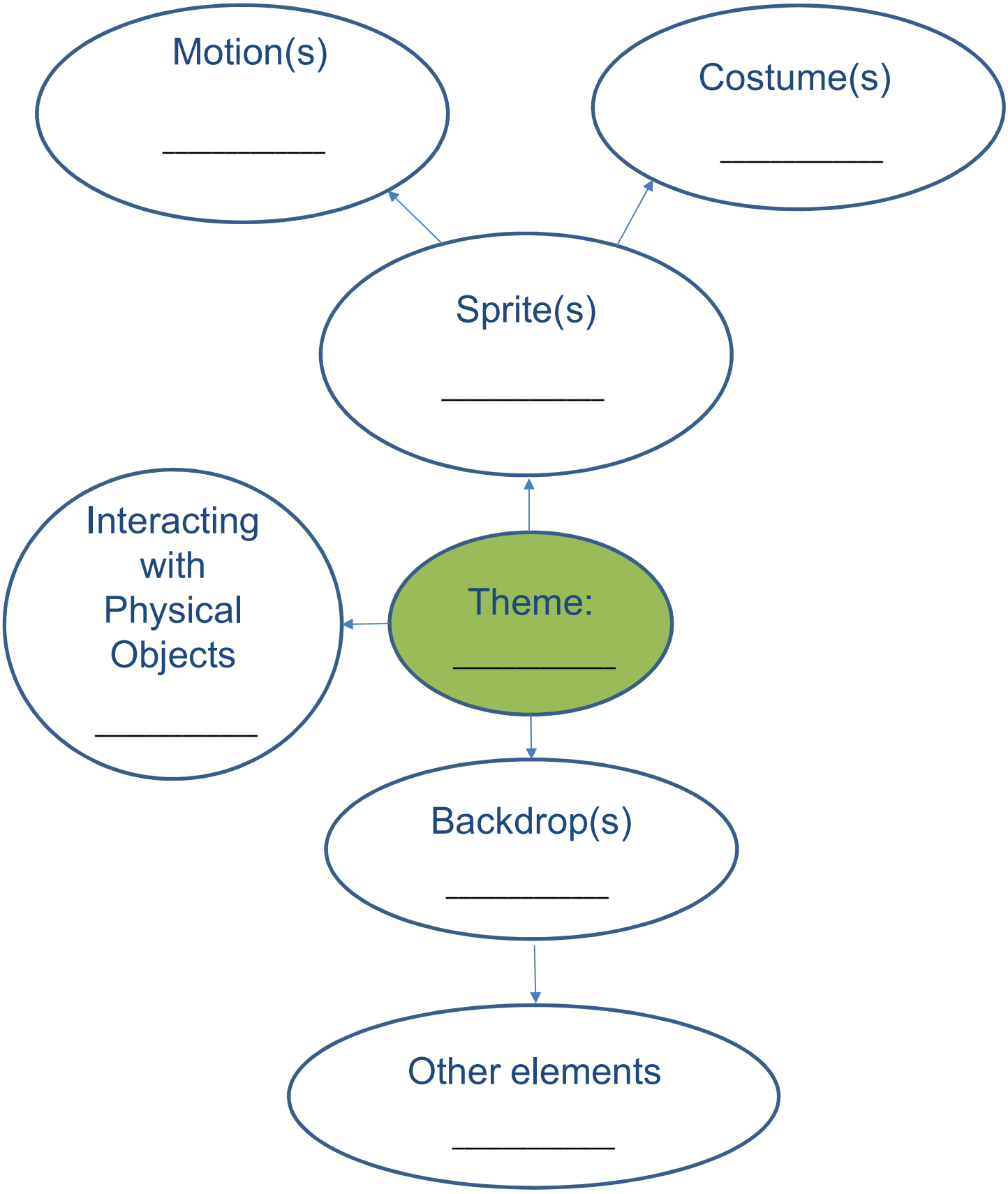
Go to “Create” to start a new project.

Let’s start to create your own project with different Theme, Sprite and Backdrop.



Space Travelling

Complete the mind map below before your start to code!



Space Travelling

To Reflect: Two Stars and a Wish Worksheet

Name of Project: _____ Name of Creator: _____

Please write down two things that you like about this project.

1

_____

2

_____

What is one thing you would like to add or change to make this project better?

_____

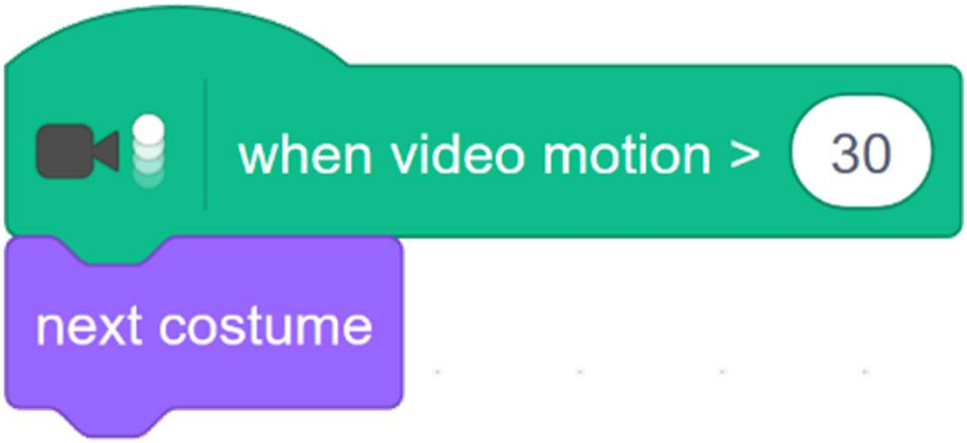
_____

_____

Space Travelling

Review Questions

1. The following blocks are **NOT** an example of:

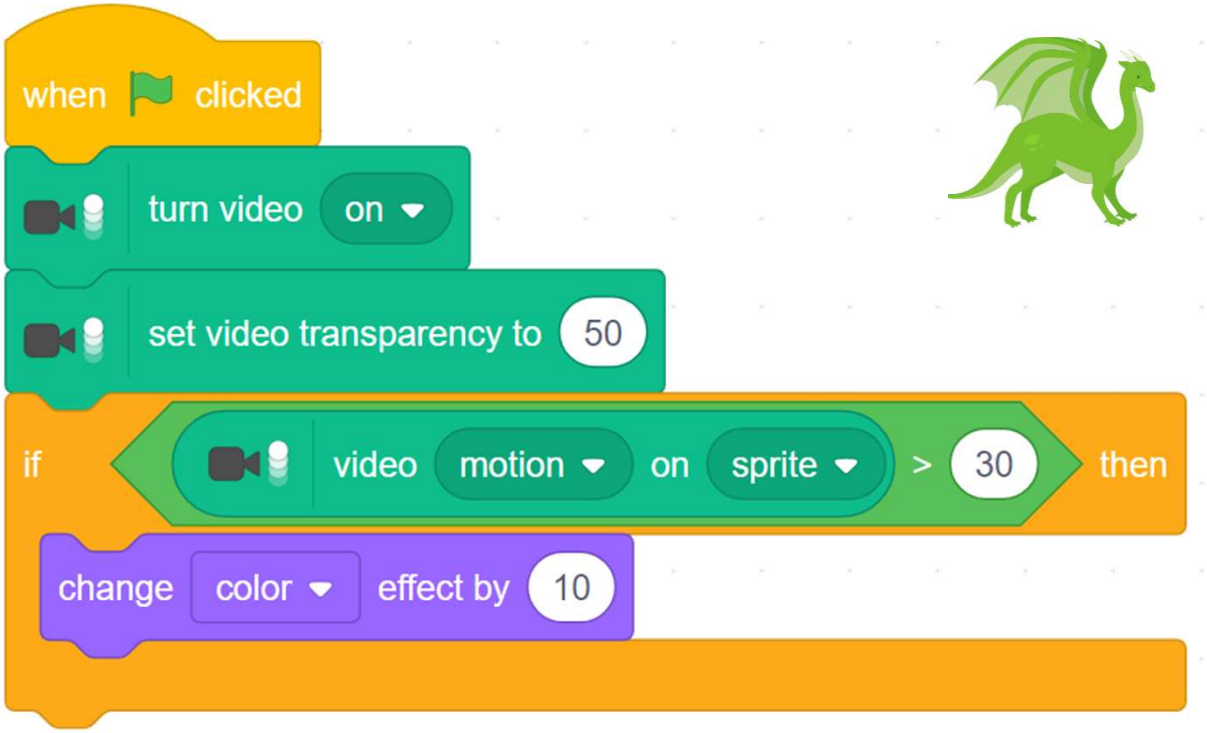


A. Events
B. Naming
C. Conditionals
D. Operators

Space Travelling

Review Questions

2. A student writes the following code and tests it by moving his hand in front of the webcam on his computer. He expects the dragon to change colour, but it only changes once and stops. What would you tell him to help him fix his problem?

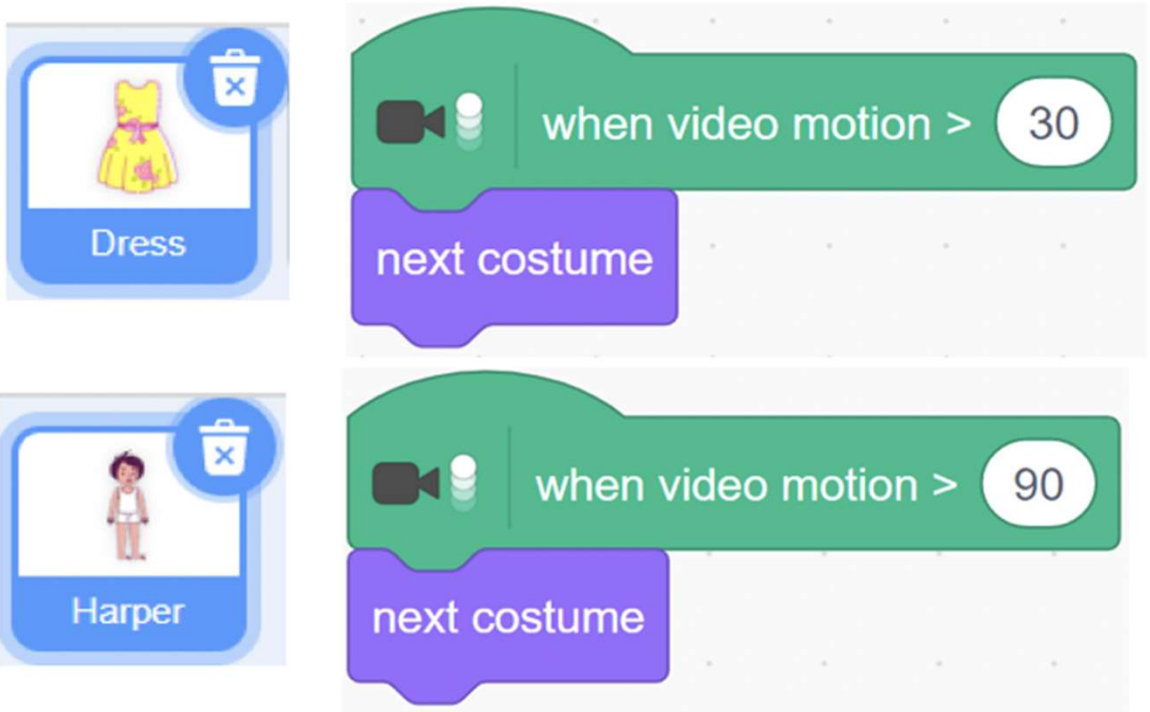


A. Change 30 to a higher number in the “if-then” block.
B. Change 10 in the “change colour effect by” block to a higher number.
C. Put the “if-then” block inside a “forever” block.
D. Use a “next costume” block instead of “change colour effect by” block.

Space Travelling

Review Questions

3. When the video sensing is on, if a user <u>waves his hand slowly</u>, Clothes (Dress) and Face (Harper), which one will be easier to change?

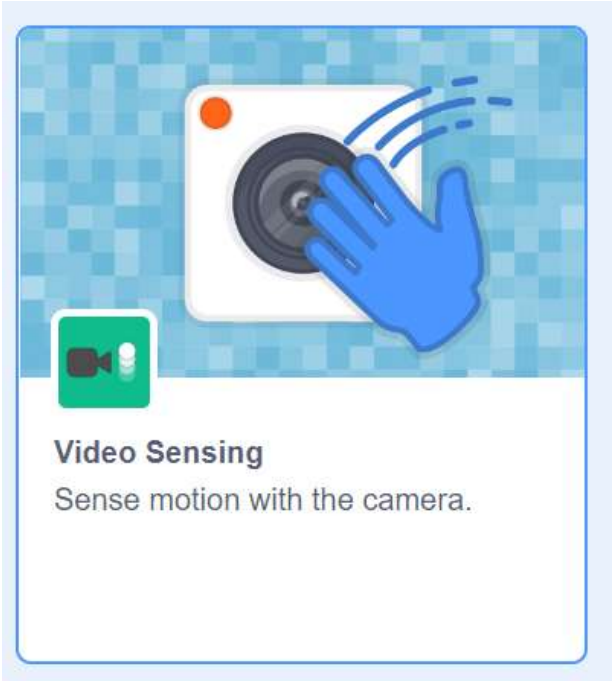


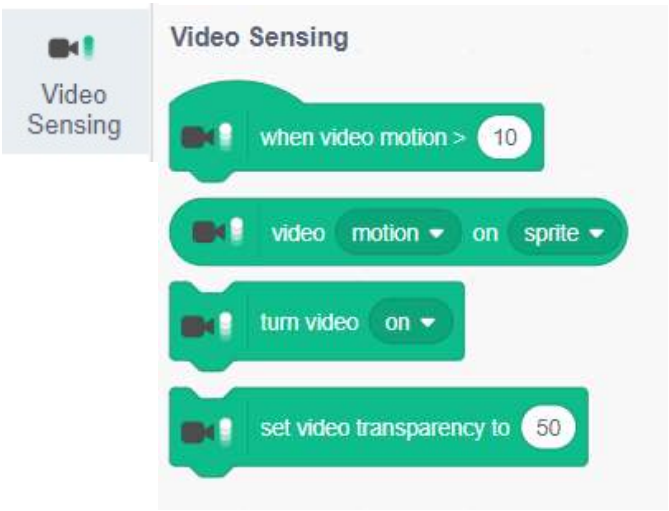
A. Dress.
B. Harper.
C. None will change.
D. No difference between them.

Space Travelling

Revision on Key Features

Video Sensing:

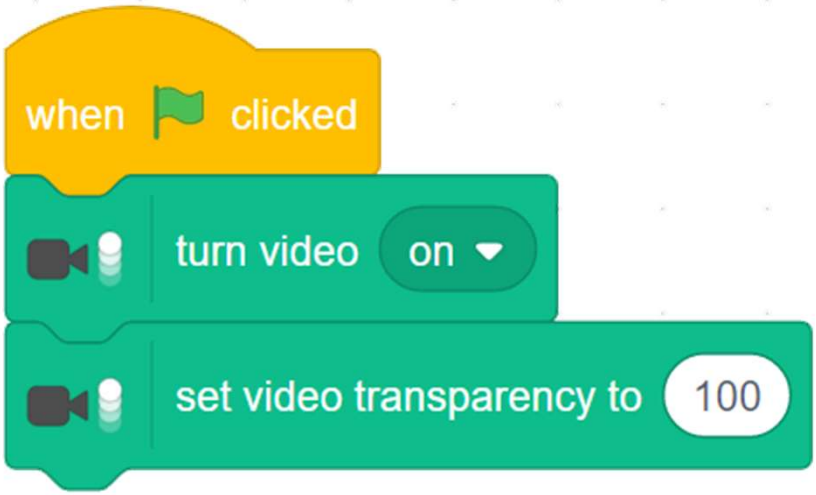




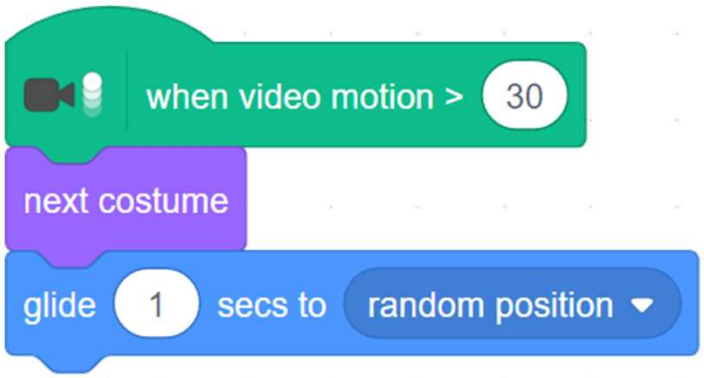
Space Travelling

Revision on Key Concepts & Practices

Sequences: It is the order in which the programming statements are executed. A wrong order would lead to incorrect programming results.



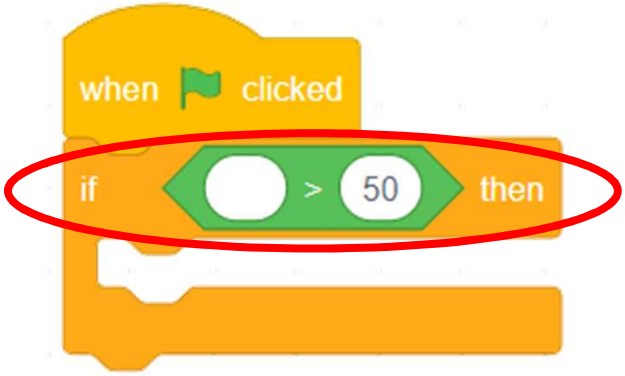
Events: We use event blocks to trigger Scratch to take actions.



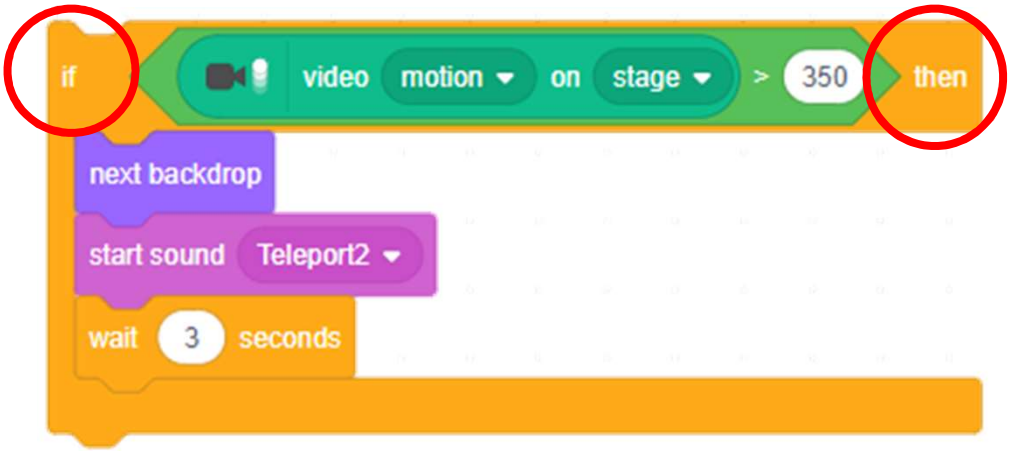
Space Travelling

Revision on Key Concepts & Practices

Operators: We use conditional operators to evaluate whether a condition is true or false. Conditional expressions always use operators such as greater than (>), less than (<) or equal (=).



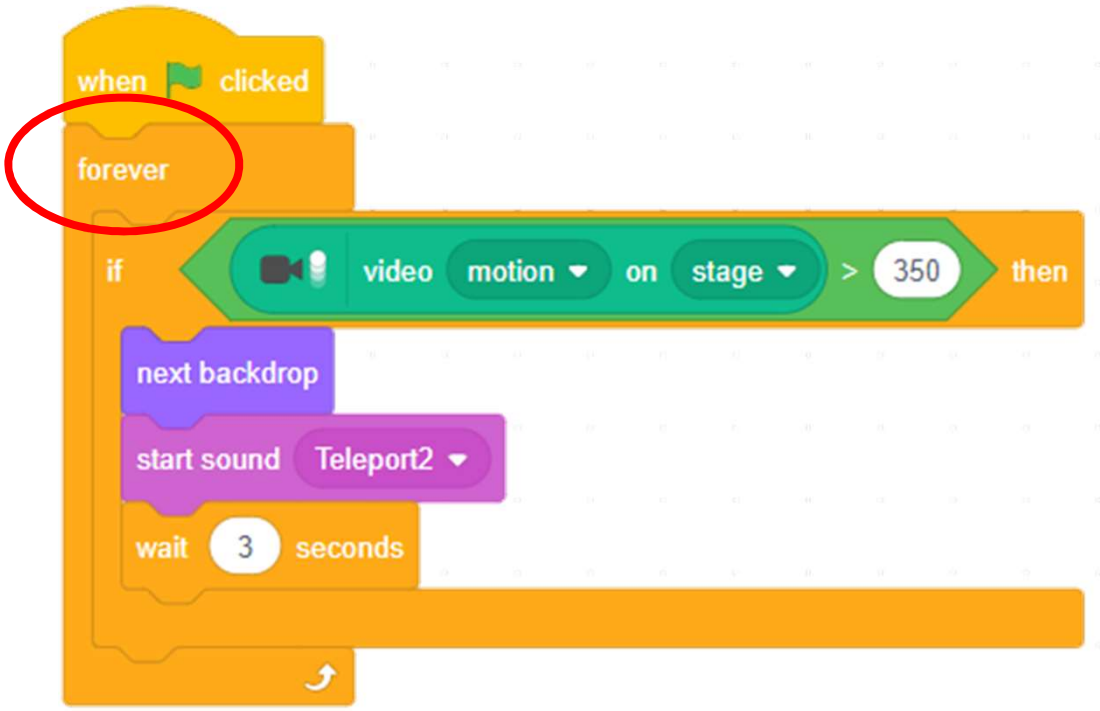
Branching/Selection: We use conditional statements in programming to enable computers to make decisions. Conditionals always have an "if" part, which tells the program in the "then" part what to do when the condition is true.



Space Travelling

Revision on Key Concepts & Practices

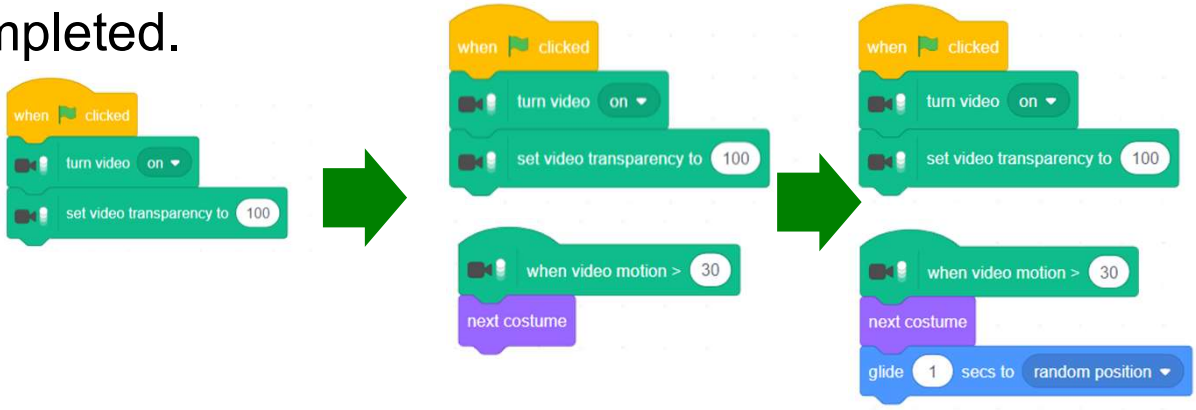
Iteration - Forever: Iteration is repeating a process in order to produce a sequence of outcomes. Forever and repeat blocks can trigger iteration in Scratch.



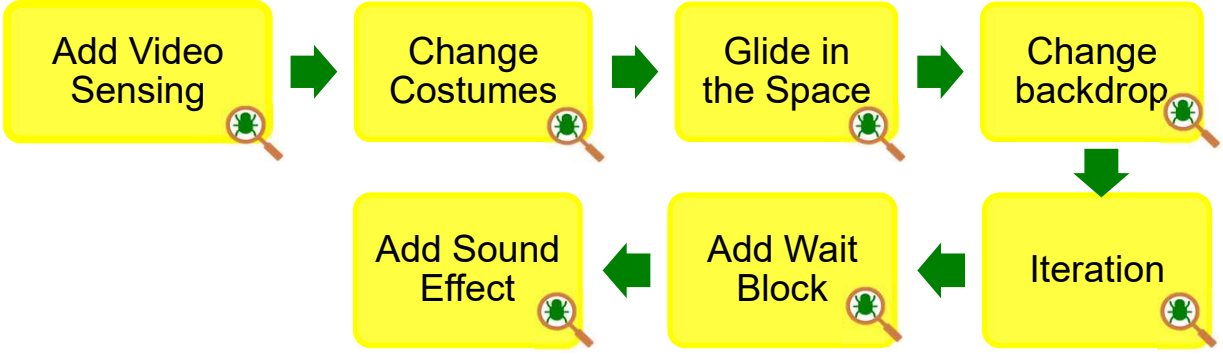
Space Travelling

Revision on Key Concepts & Practices

Being incremental and iterative: to work out a sub-task as an iteration, try it out, then work out another sub-task based on the codes of the previous sub-task in one more iteration until the whole programming task is completed.



Testing and debugging: Testing a computer program is the process of checking if it can produce results as designed. Debugging a computer program is the process of finding out ways to revise the program so that the bugs can be removed.



Appendix

Operation Manual

Space Travelling

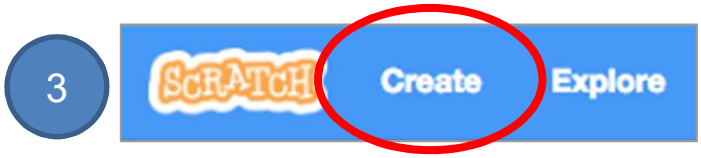
To Code: Adding the Backdrop and Sprite

1. Turn on the webcam of your computer.

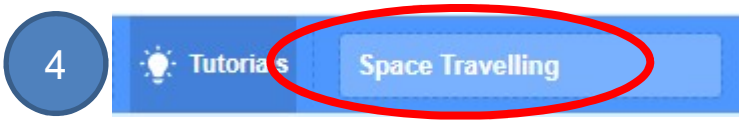
See Student Guide P.4

2. Sign into your account at scratch.mit.edu.

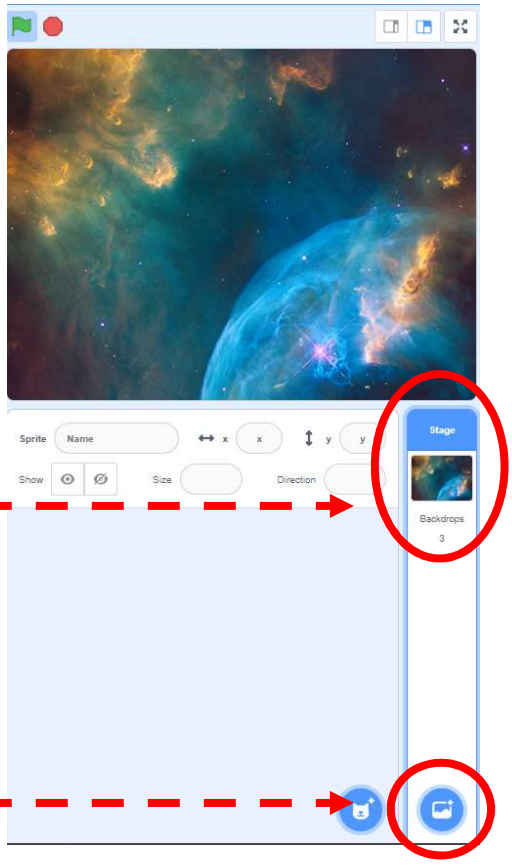
3. Go to "**Create**" to start a new project.



4. Name it "**Space Travelling**".



5. Choose a good backdrop for the space.



5 Click on the Stage

6 Click on the "Choose a Backdrop" icon

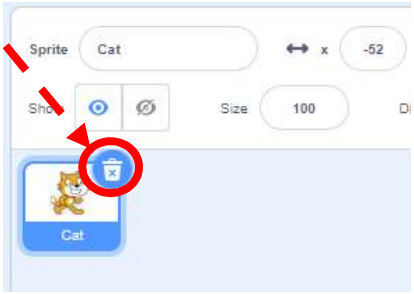
Space Travelling

To Code: Adding the Backdrop and Sprite

Choosing Sprites and Resizing

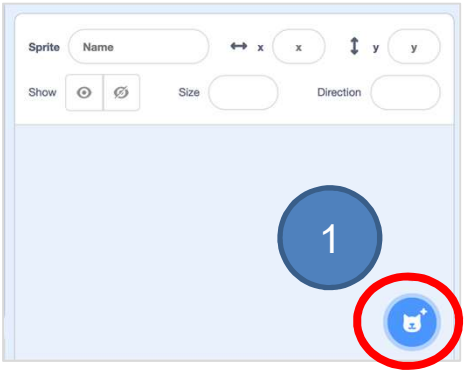
See Student Guide P.4

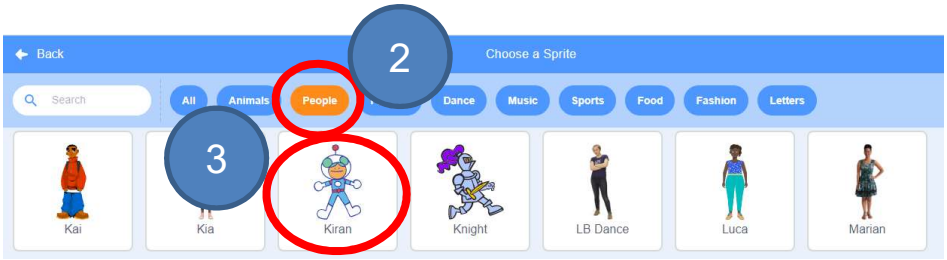
Delete the original Scratch cat sprite by clicking the “X” in the upper right corner of its image.



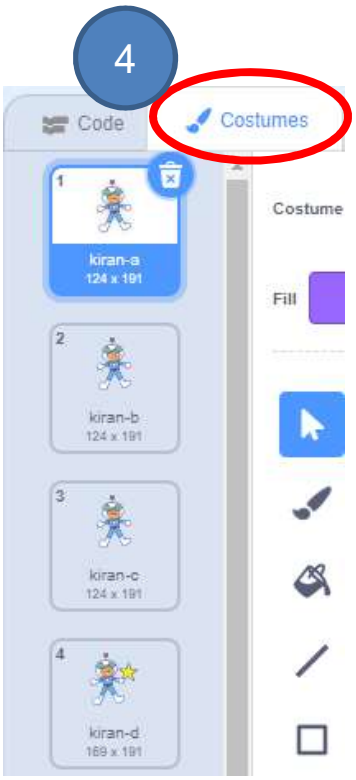
Let’s choose the new sprites for your project.

1. Click on the “**Choose a Sprite**” icon.
2. Click on the “**People**” category on the top to see a large set of sprites.
3. Add “**Kiran**" sprite to the project.





4. Now, you can see a set of costumes for Kiran.
5. You can also resize and rename “**Kiran**” sprite.



Space Travelling

To Code: Change Backdrop

See Student Guide P.15

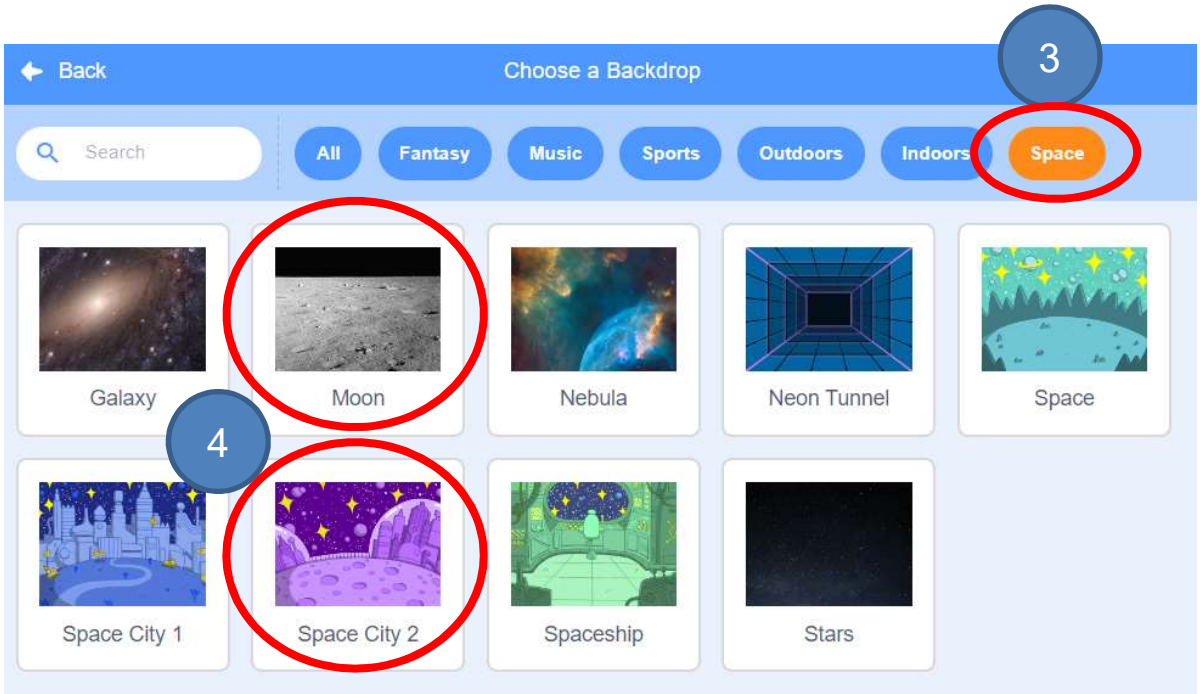
Try to add a total of 3 backdrops for your space travelling!

1. Click the "**Backdrop**" in **Stage.**

2. Click on the "**Choose a Backdrop**" icon.

3. Click "**Space**" category.

4. Choose two more backdrops.



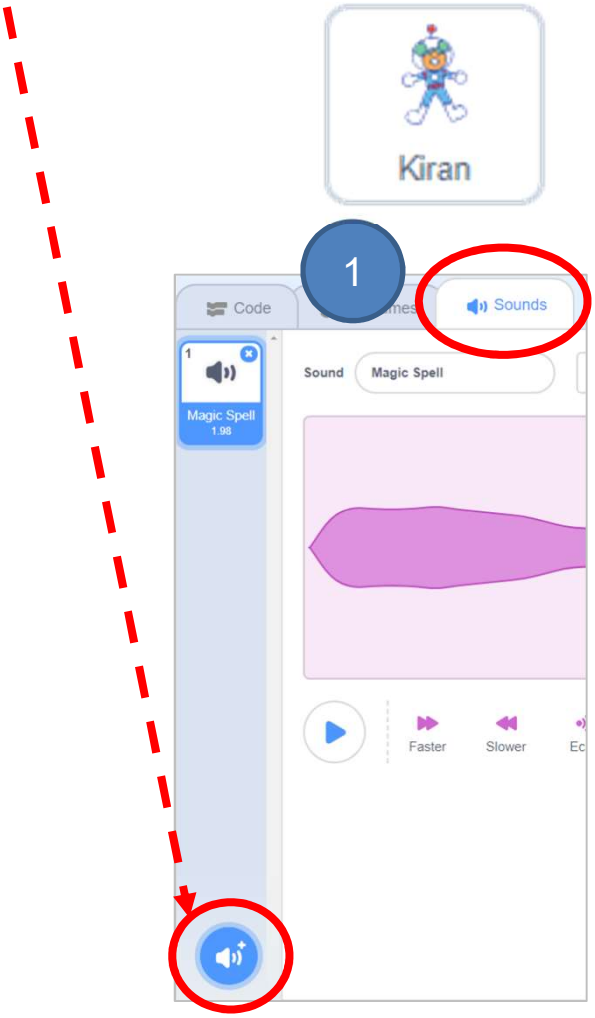
Space Travelling

To Code: Add Sound Effect (Sprite)

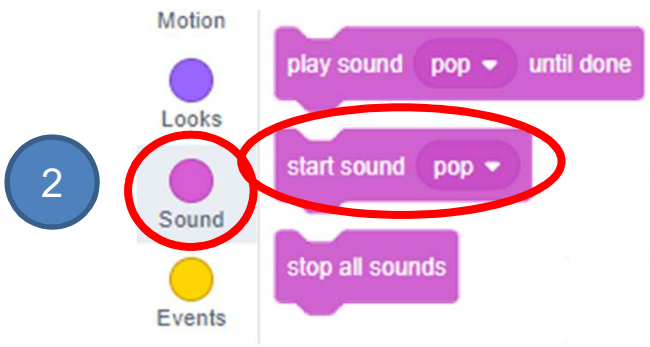
See Student Guide P.21

Make the sprite play a sound based on the video motion.

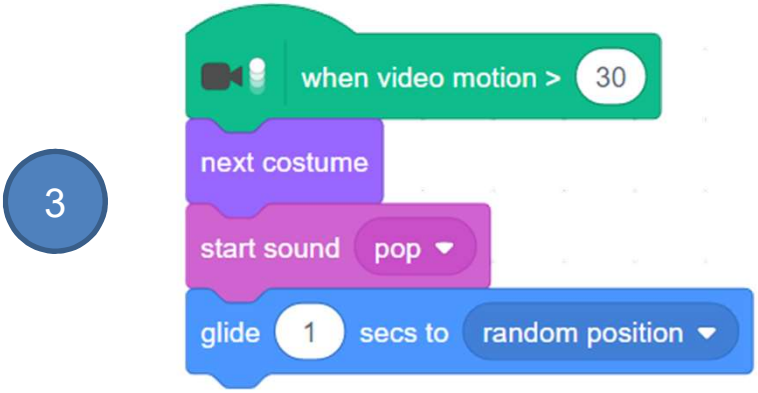
1. Go to the "**Sounds**" tab. Click on the "**Choose a Sound**" icon at the bottom left to select music from the library.



2. Drag out the "**start sound**" block from the "**Sound**" drawer.



3. Snap the "**start sound**" block after "**next costume**" block.



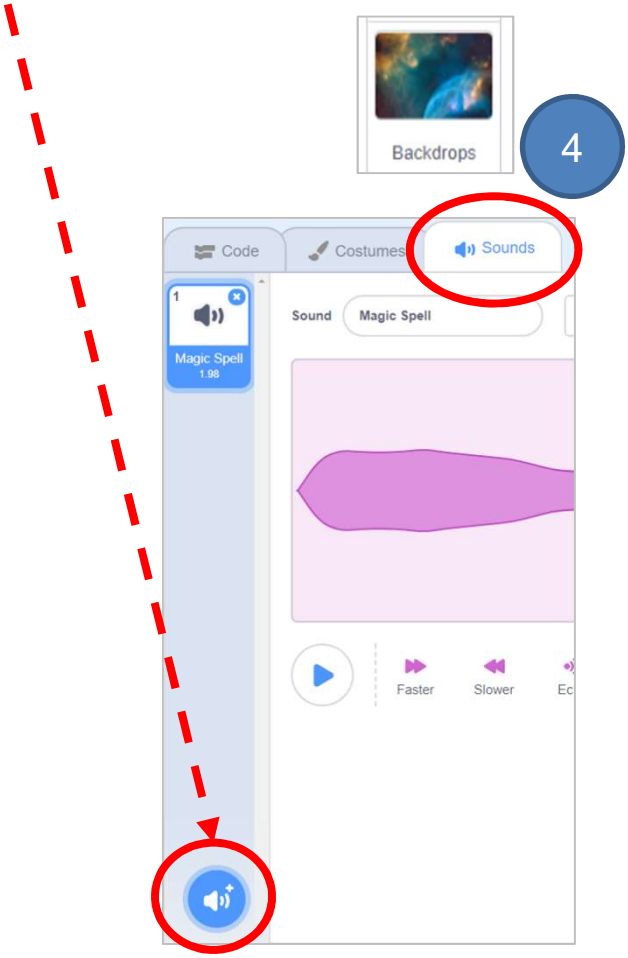
Space Travelling

To Code: Add Sound Effect (Backdrop)

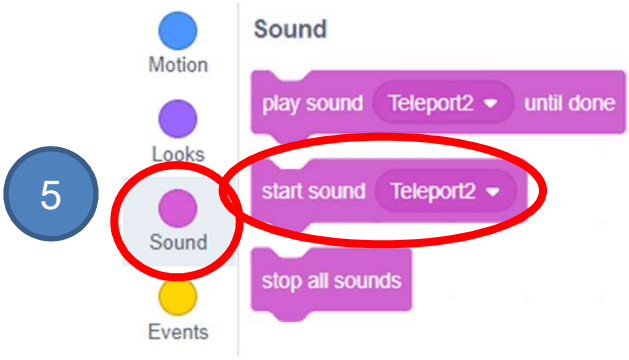
Repeat the previous step to make the backdrop play a sound based on the video motion.

See Student Guide P.21

4. Go to the "**Sounds**" tab. Click on the "**Choose a Sound**" icon at the bottom left to select music from the library.



5. Drag out the "**start sound**" block from the "**Sound**" drawer.



6. Snap the "**start sound**" block inside the **"if-then"** block.