Basic Coding Courses

Primary School

Chief Editor Xiaolan Tang

Editors Zhenyi Chen
Chujian Chen
Xiasheng Lin
Ying Ma

Published by Makeblock Co., Ltd.

CONTENTS /

Unit 1 Events and Sequences

Unit 2 Loops

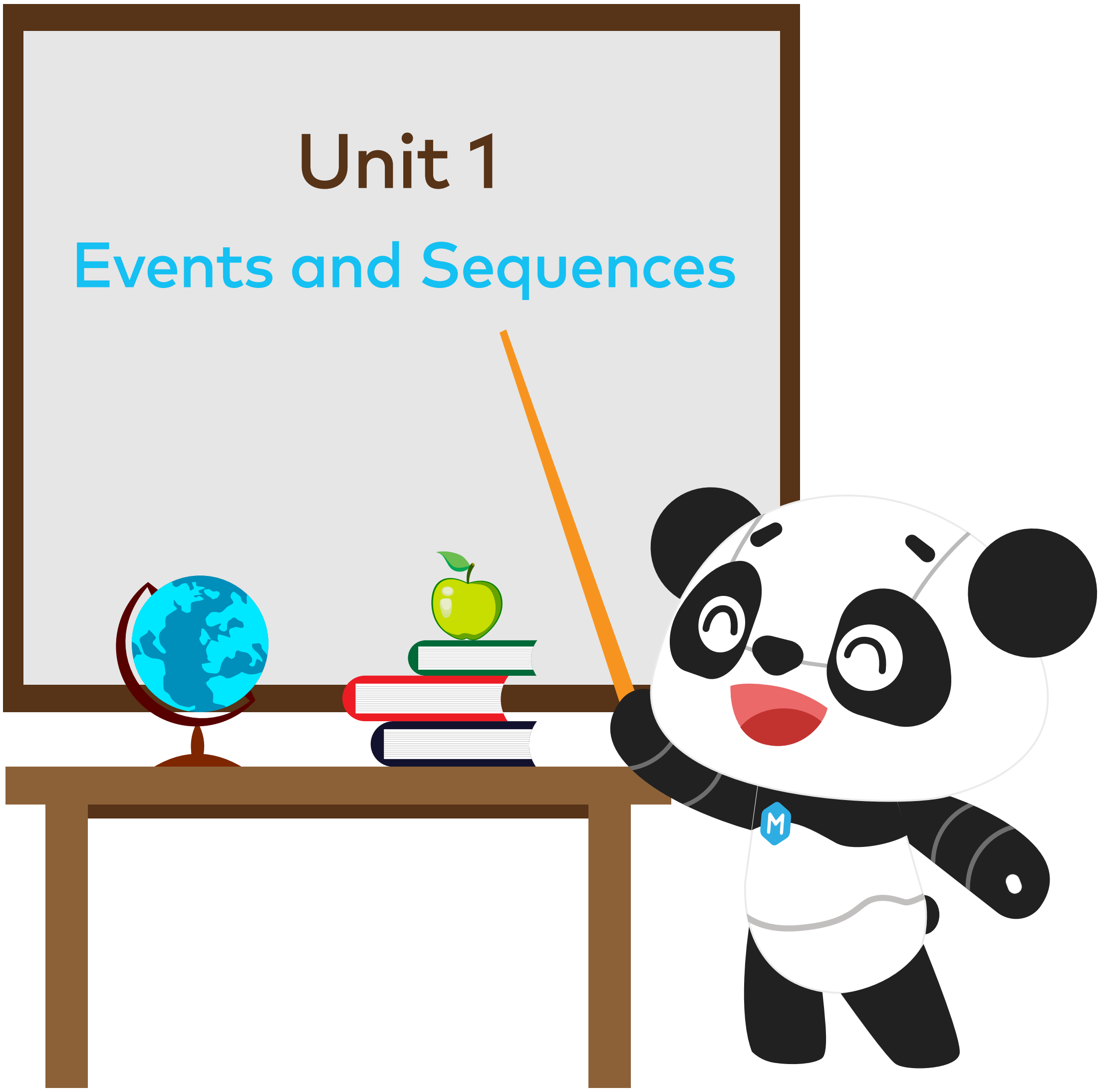
Unit 3 Conditionals

Unit 4 Functions

Unit 5 Variables

Unit 6 Game Design

Unit 7 Robots



Unit 1

Events and Sequences

Lesson 1 The Secret of Codey Rocky

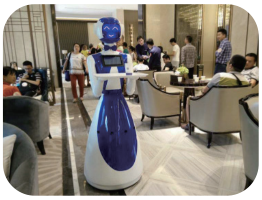
Learning Objectives

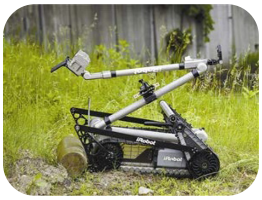
1. Understand the definition of Program and what Program can do.
2. Get to know Codey Rocky and its features.
3. Master the basics of mBlock 5.
4. Learn how to upload programs.

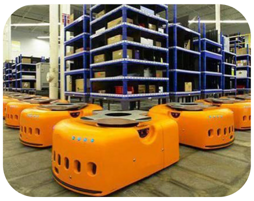
Situated Learning

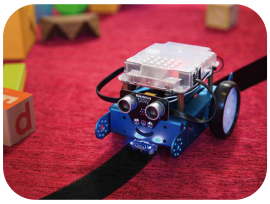
1.Have you ever seen a robot in daily life? What do you think robots are?

2.Try guessing what these robots are used for. Match the picture with the robot.









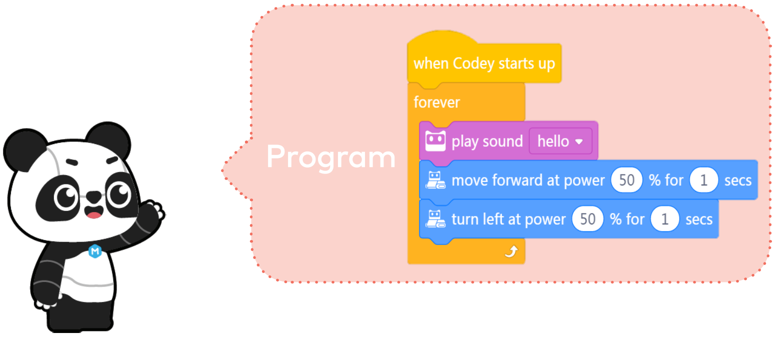
Educational robot   Robots in logistics   Delivery robot   Security robot

3.Unveil the mysteries of Codey Rocky.

Codey Rocky is a programmable educational robot. Tiny but mighty. It can not only avoid obstacles but can also follow black lines. Using mBlock 5, you can even add more abilities to Codey Rocky, such as face recognition.

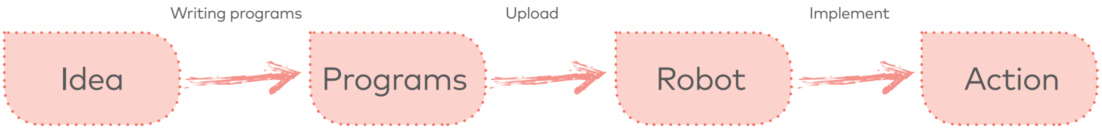
But how do we have Codey Rocky and other robots follow our instructions? Do you have any ideas? Write down your idea.

Key Concept



```
when Codey starts up
forever
    play sound hello
    move forward at power 50 % for 1 secs
    turn left at power 50 % for 1 secs
```

Program is an artificial language that we use to tell robots what to do. We translate our instructions into a piece of program. Then we upload the program to the robot, making it do a variety of things as programmed.



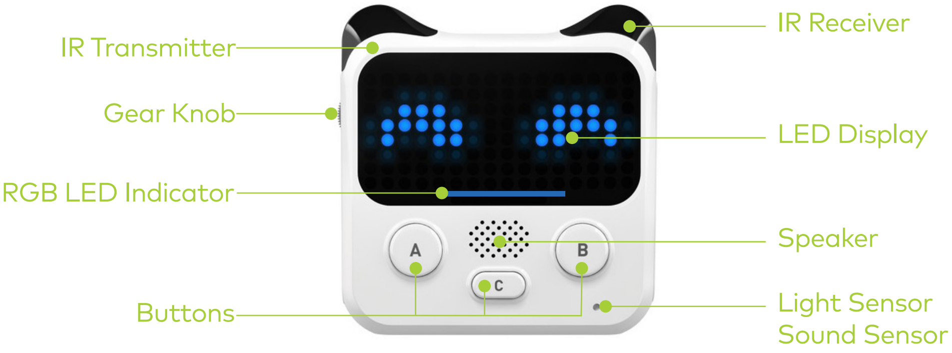
Idea → Writing programs → Programs → Upload → Robot → Implement → Action

Extension

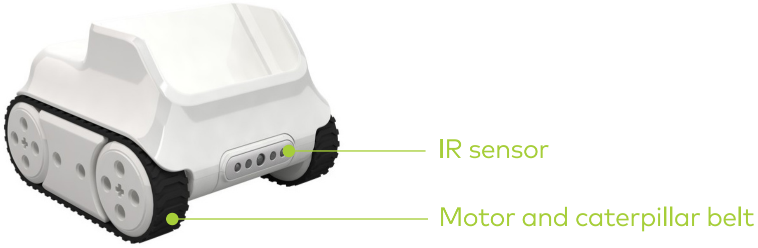
Codey Rocky is an educational programmable robot. You can use software to code the robot, manipulating it to do a variety of things you can imagine. It's also a good companion that can help children learn to code. With mBlock 5, children are able to master the basics of coding and develop logical thinking as well as computational thinking. In addition, Codey Rocky supports technologies like AI and IoT, which exposes children to the latest cutting-edge technologies.

Combine Codey with Rocky and then you get a Codey Rocky. Now let's take a look at them one by one.

(1) Codey: As the brain of the robot, Codey is equipped with a variety of sensors and programmable blocks. It can work individually and can also work with Rocky to perform more tasks. Now pick up your Codey. Let's take a look at what sensors it has.



(2) Rocky serves as the chassis of Codey. It adds more abilities to Codey, like avoiding obstacles, identifying colors, following lines and more.



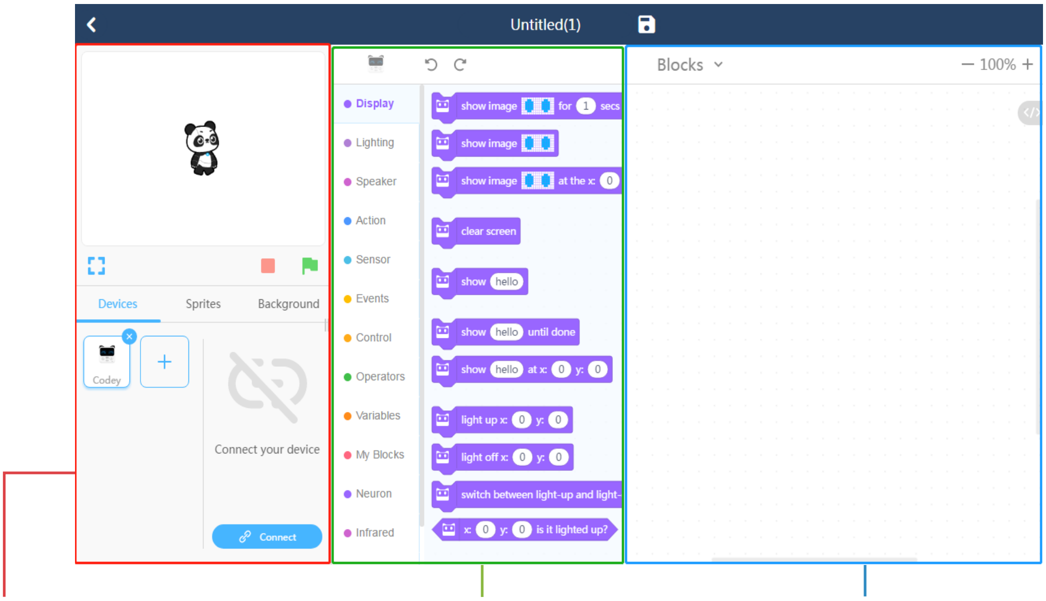
Coding Practice

mBlock 5 is a programming tool which supports block-based and Python programming languages. It's developed based on the Scratch 3.0, an open-source software tool that's developed jointly by MIT and Google.

Using mBlock 5, you can write programs that tell Codey Rocky or other robots do whatever you would like. You can even take advantage of the software to create stories, games, and animations that are engaging and unique. Moreover, mBlock 5 exposes children to technologies, like AI, deep learning and model training. In a word, mBlock 5 can be one of the best options for first-time coders.

mBlock 5 Logo

Take a Tour of the Interface：



Stage

In this area, you can show your projects, connect devices and upload programs, add sprites and backgrounds.

Blocks area

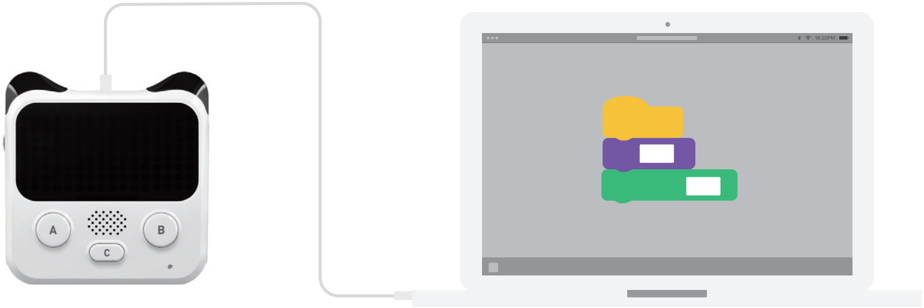
You can find the blocks you need by color or category.

Script area

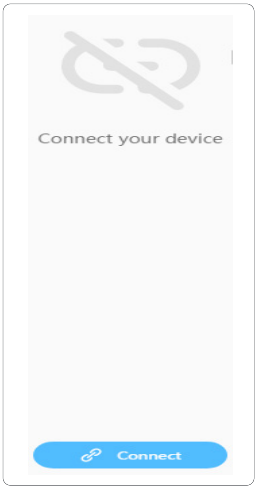
You drag blocks to this area to form programs.

Bring Your Codey Rocky to Life

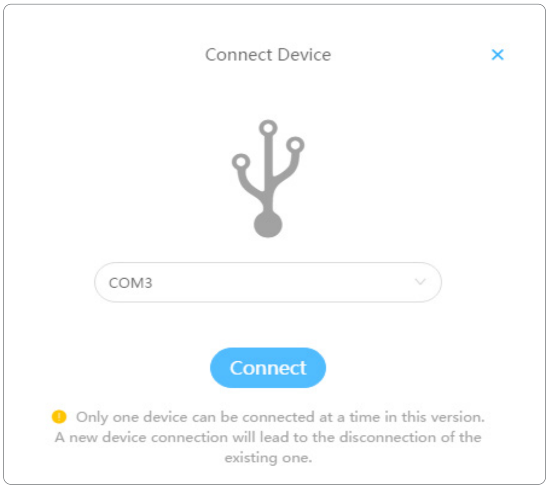
(1) Connecting to a computer: Connect Codey to the computer via the USB cable. Then power on Codey.



(2) Selecting the serial port: Open mBlock 5, click Connect, and select the correct serial port.

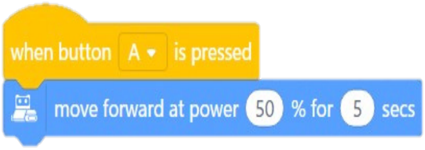


Select a serial port
Click Connect



Only one device can be connected at a time in this version. A new device connection will lead to the disconnection of the existing one.

(3) Practice yourself! Use mBlock 5 to create the program as shown below.



```
when button A ▾ is pressed
move forward at power 50 % for 5 secs
```

(4) Upload the completed program to Codey.

(5) Unplug the data cable and place Codey Rocky on a table or the ground. Press the button A and watch how Codey Rocky reacts.

Share

In this lesson, we created a project____ .

During the process, we came into an issue___________________________________ .

In the end, we solved the problem by _______________________________________.

Self-review

1.In this lesson, I learned_________________________

2.The part I like most about this lesson is_______________

3.Can you think of any situations where programs are used in daily life?

____

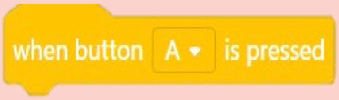
Press Buttons to Change Emotions

Learning Objectives

1. Understand the concept of Events.
2. Master how to use the Events blocks in a program.
3. Use the Events blocks to create buttons that can function as you would like.

Key Concept

Events

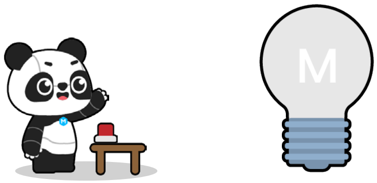


when button A ▾ is pressed

In programming languages, Event is an action that can cause things to happen. In a program, there might be one or several events.

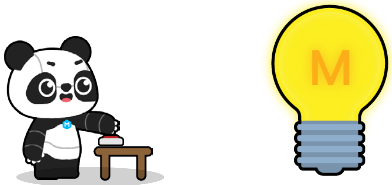
Situated learning

1 The panda is going to turn on the light.



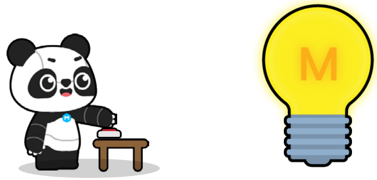
To turn on the light,the panda presses the button.

2 Once the panda presses the button,the light bulb lights up.



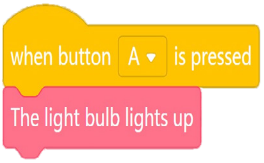
Once the panda presses the button, the power will be on and the light bulb will light up.

3 The action of pressing the button is an "Event".



Pressing the button leads to the light bulb lighting up.In this case, pressing the button is an event.

4 In computer programming, we need to first select an event to trigger the program.



```
when button A ▾ is pressed
The light bulb lights up
```

We can write programs to simulate this process.

Learn through play

1. Divide the students into 3 or 4 groups.
2. The teacher draws some figures on the blackboard, like triangle, circle, square and star.
3. The teacher defines the following actions as the events:

(1)When the teacher puts his or her hand on the triangle;

(2)When the teacher put his or her hand on the circle;

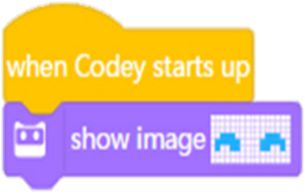
(3)When the teacher puts his or her hand on the square;

4.Students have to give different reactions based on the actions of the teacher. For instance, when the teacher puts his or her hand on the triangle, students have to stand up.

Coding Practice

Start up and Smile

How to make Codey Rocky【start up and smile】? Observe when the teacher is demonstrating how to do. Then practice yourself. Write the following program:



```
when Codey starts up
show image
```

★ The yellow block【when Codey starts up】is the Event;

★ The blue block【show image】is used to change the image displayed on the LED panel. You can pick or draw an image yourself with the block.

Imitate and Create

Try using the Event block to create a project of your own.

Task 1: Learn how to set events. Write programs to make Codey Rocky change its facial expressions based on the events. (when button A/B/ C pressed).

Follow the teacher's instructions and try completing your program. Take on a new challenge when you complete Task 1.

Task 2: Write programs to make Codey Rocky react in response to the events (when button A/B/C pressed), like changing its facial expression or making different sounds.

Share

In this lesson, we created a project____ .

During the process, we came into an issue __________________________________ .

In the end, we solved the problem by_______________________________________ .

Self-review

1.In this lesson, I learned________________________

2.The part I like most about this lesson is________________

3.Can you think of any events in daily life? (You can draw in the space below) ____________________________

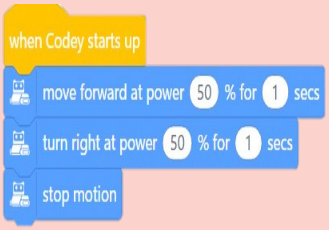
To Be an Animation Designer

Learning objectives

1. Understand the concept of Sequence.
2. Master the basics of Sequence and learn how to create animations using Sequence.

Key Concept

Sequence

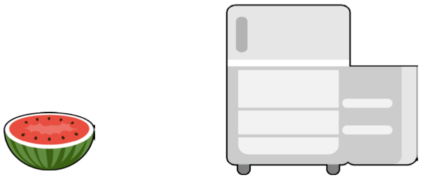


```
when Codey starts up
move forward at power (50) % for (1) secs
turn right at power (50) % for (1) secs
stop motion
```

To make Codey Rocky move forward for 1 second, turn right for 1 second and then stop moving, we need to make sure our coding blocks are arranged in the order as shown above. In brief, Sequence refers to a series of steps which are carried out in order to complete a task.

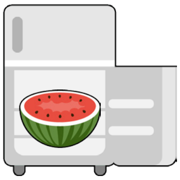
Situated Learning

1 If the panda wants to put the watermelon in the refrigerator, it should follow the steps as below:



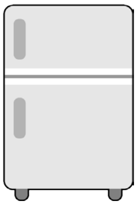
Step 1: Open the refrigerator.

2 Put the watermelon into the fridge.



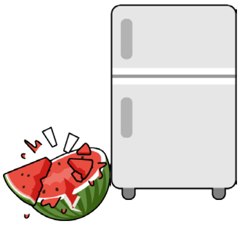
Step 2: Put the watermelon into the fridge.

3 Close the fridge door.



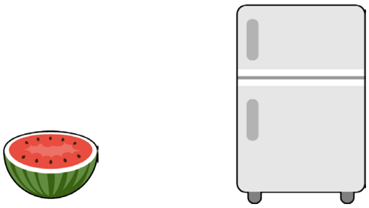
Step 3: Close the fridge door.

4 The panda has to follow the sequence of steps, otherwise...



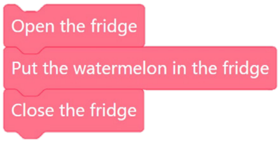
Otherwise,the panda will fail to put the watermelon into the fridge.

5 A set of ordered steps for accomplishing a task.



To put the watermelon into the fridge, we need to follow the steps above. The steps are called Sequence.

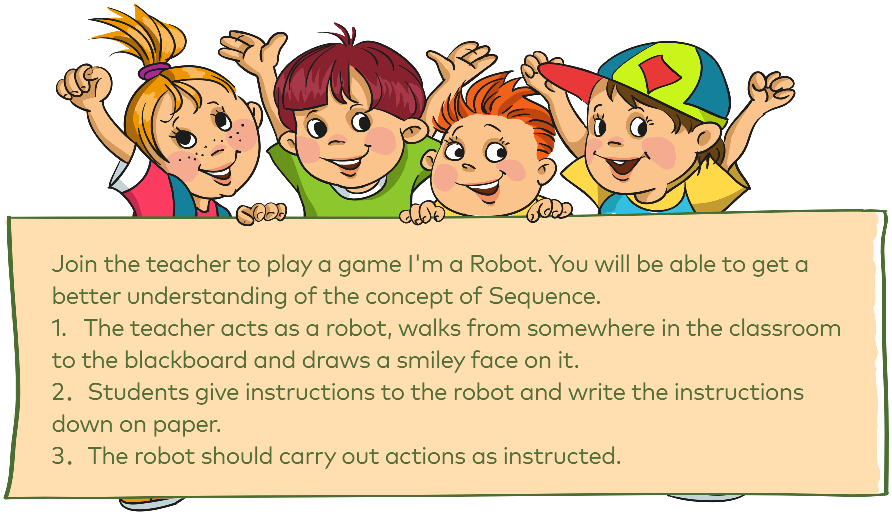
6 In computer programming, we should follow the sequence of steps as shown above when putting the watermelon in the fridge.



```
Open the fridge
Put the watermelon in the fridge
Close the fridge
```

We can write programs to simulate this process.

Learn through play



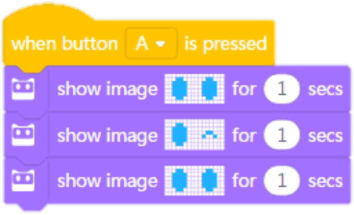
Join the teacher to play a game I'm a Robot. You will be able to get a better understanding of the concept of Sequence.

1. The teacher acts as a robot, walks from somewhere in the classroom to the blackboard and draws a smiley face on it.
2. Students give instructions to the robot and write the instructions down on paper.
3. The robot should carry out actions as instructed.

Coding Practice

Winking Eyes

How to make Codey Rocky【wink eyes】? Observe when the teacher is demonstrating how to do. Then practice yourself.



```
when button A ▾ is pressed
show image  for 1 secs
show image  for 1 secs
show image  for 1 secs
```

The Program Sequence

1.Add the Event block ——"When button A is pressed";

2.Use the block " show image ( ) for ( ) secs" to make Codey open its eyes;

3.Duplicate the block "show image ( ) for ( ) secs" to make Codey wink its eyes;

4.Add the block "show image ( ) for ( ) secs" to make Codey open its eyes;

5.Upload the program to Codey. When the button A is pressed, Codey will wink at you.

Traditionally, animation designers would make animations by following these steps: Put a sheet of static drawing on the table first and unfold another piece of drawing paper on top of the first paper. Designers would outline the frame and then change the drawing bit by bit at a time. Then another piece of paper, outline the frame and change the drawing slightly again. Designers repeat the steps over and over again until they complete a series of pictures that are slightly different from each other. Then, they flip the drawings quickly to animate the pictures. Based on the same principle, we use the block "show image ( ) for ( ) secs" in our program to create animations.

Imitate and Create

Students are supposed to complete the project 【Winking Eyes】 as the teacher did.

Task 1: Work in pairs. Add the block "show image ( ) for ( ) secs" to the code. Duplicate the block but modify the image to make it look different each time. Then arrange the coding blocks in the order as shown above. Ok, now the animation is ready. Of course, you can add some facial expressions to Codey if you want.

Create A New Animation

Task 2: Work in pairs to complete Task 2. You need to create an image by filling the following grid. Then use the block【show image ( ) for ( ) secs】 to create the same image; duplicate the block but change the image slightly; repeat the steps until you have a series of coding blocks. These coding blocks show different images and should be arranged in sequence.

Upload the program to Codey. Then invite classmates to watch your animation.

Share

In this lesson, we created a project ____ .

During the process, we came into an issue____________________________________ .

In the end, we solved the problem by _______________________________________ .

Self-review

1.In this lesson, I learned that____________________

2.The part I like most about this lesson is_________________

3.I want to design my own animation. It will be like_______.

(Or you can draw it out in the space below.)

____

Identify the Bug

Learning Objectives

1. Understand the concept of Bug.
2. Learn how to identify bugs and fix the bugs.

Key Concept

Bugs are inevitable when we write programs. A "bug" means an insect in its literal sense. But in programming, a bug refers to a mistake that will lead to the failure of a program. It's like a typo in an article. You need to correct the error because it might make your article difficult to understand.

Situated Learning

At the initial stage where computers were invented, unlike today's portable computers, they were extremely large in size back then.

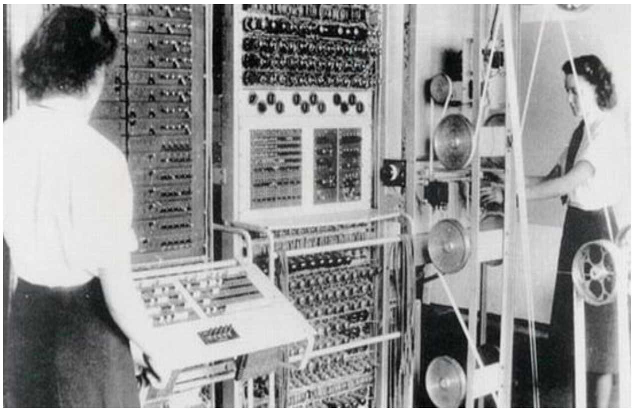
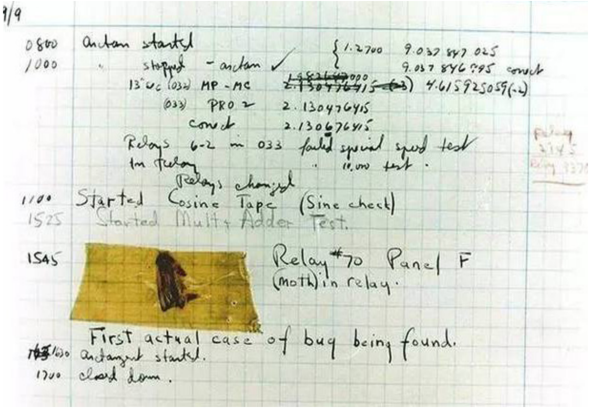


Figure 1 Colossus computers were used to decipher German codes during WW II (Picture from Wiki).

During this period, once, a colossus computer failed to work properly. The whole team of programmers tried to identify the problem but to no avail. In the end, Grace Murray Hopper, a female programmer, identified what the problem was: a moth flew to the inside of the computer and caused the glitch. When they removed the moth, everything was back on the right track. It was the first bug in a computer program that was found and programmers affixed it to the logbook (see the picture as shown below). Since then, the term bug becomes common in use when people refer to mistakes in a computer program. Naturally, Grace Murray Hopper was since then considered as the Mother of Debug.



9/9

0800 Arctan started
1000 " stopped - arctan ✓ {1.2700 9.037 847 025
9.037 846 795 correct
13" uc (032) MP - MC ~~1.982147000~~ ~~2.130476415~~ (-3) 4.615925059(-2)
(033) PRO 2 2.130476415
correct 2.130676415
Relays 6-2 in 033 failed special speed test
In relay " 11,000 test .
Relays changed
1100 Started Cosine Tape (Sine check)
1525 Started Mult+ Adder Test.
1545 Relay #70 Panel F
(moth) in relay.
First actual case of bug being found.
1630 arctangent started.
1700 closed down.

Relay 2145
Relay 3370

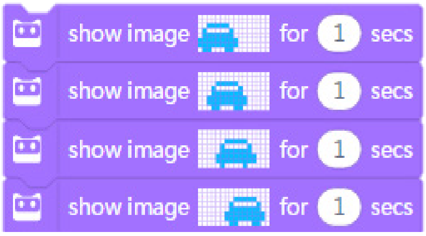
Figure 2 The moth caused the computer mistake and this is the first mistake in a computer program.(Picture from Wiki).

Coding Practice

Identify the bug

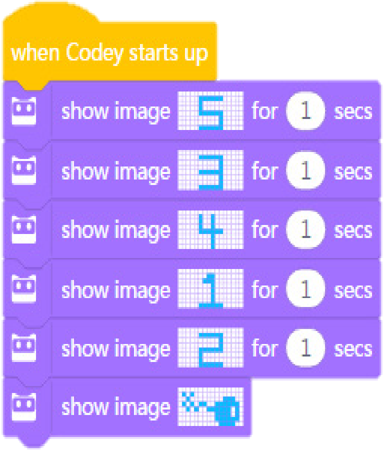
Here are three programs from the teacher. But there're some bugs hidden in the programs. Can you find them?

1 This is a car. But the key was stolen! Help it find the key.



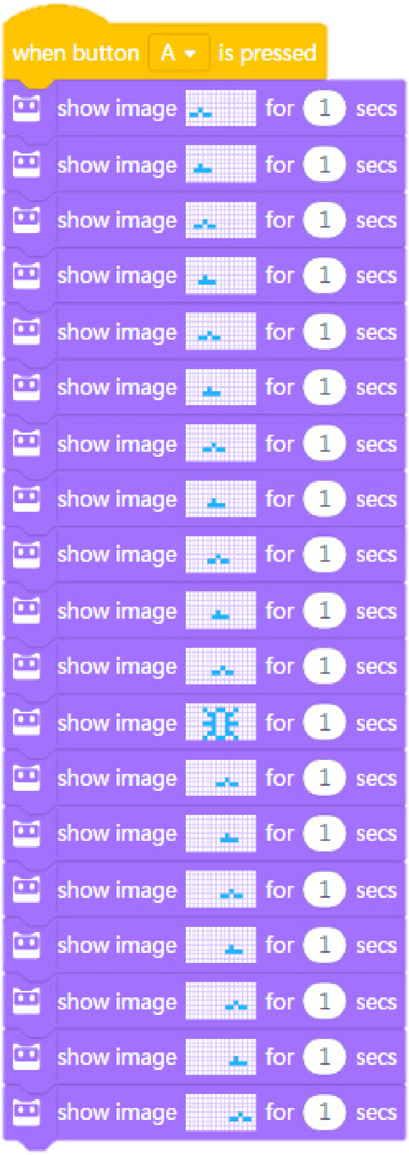
```
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
```

2 This is a bomb. But the bomb can't count down. Can you fix it?



```
when Codey starts up
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image
```

3 There is a little earthworm. It's crawling on the ground but has to stop when meeting a big bug. Remove the bug to let the earthworm move on.



```
when button A is pressed
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
show image for 1 secs
```

Share

In this lesson, we created a project ____ .

During the process, we came into an issue____ .

In the end, we solved the problem by ____ .

Self-review

1.In this lesson, I learned that_____________________

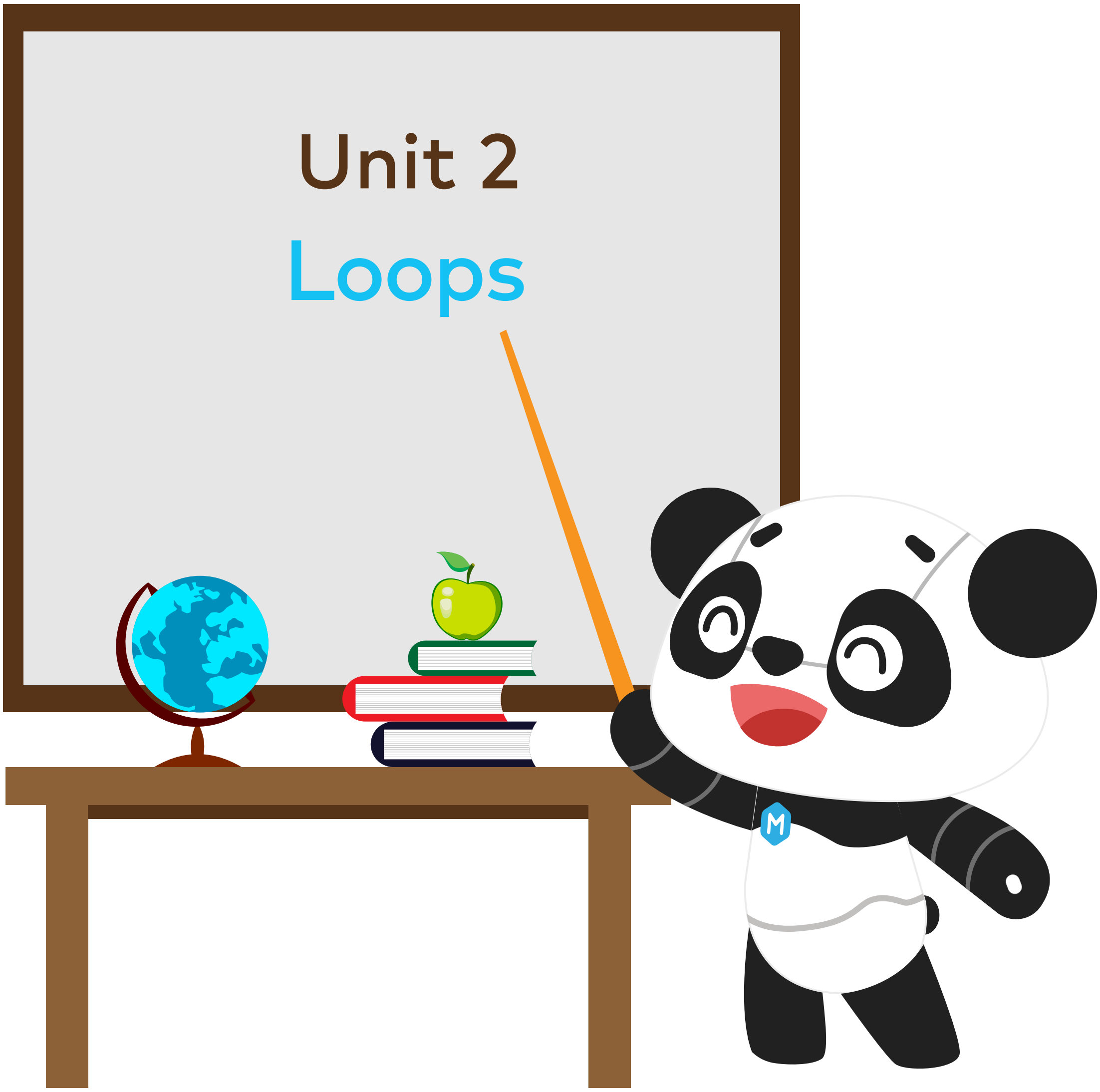
2.The part I like most about this lesson is__________________

3.Have you ever been asked to find bugs when you are learning other subjects? If yes, what subjects then? Write them down.

____

Unit 2

Loops

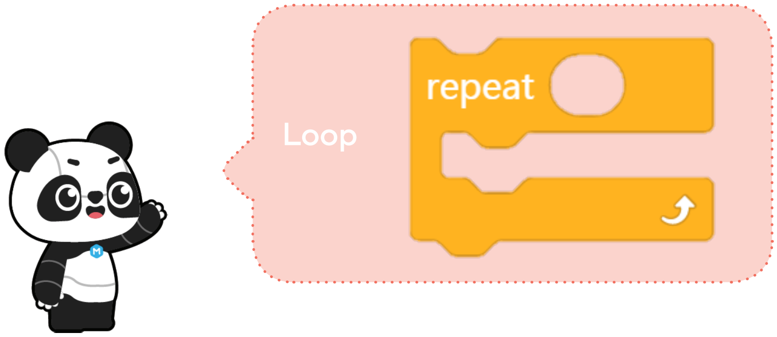


The Steamed Bread Can't Jump

Learning Objectives

1. Understand the concept of Loops.
2. Know how to use the Counting Loop blocks.
3. Learn how to create animations using the Counting Loop block in your program.

Key Concept



In programming languages, a Loop refers to a set of instructions that need to be executed repeatedly. When writing programs, we can use the Repeat block to replace those instructions that are executed repeatedly, making the code clean and concise. With the Repeat block, we can also create fun projects.

Situated Learning

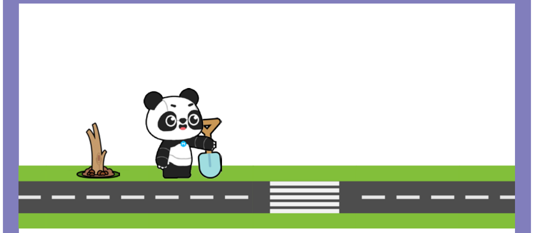
1 The panda is going to plant trees.

The panda is planting trees on the street.

2 To plant a tree, the panda has to take these steps.

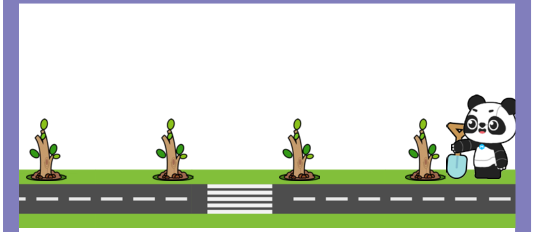
To plant a tree, the panda has to take these steps: dig a hole; drop down the sapling; cover it with soil; move forward 5 meters.

3 If the street is 20 meters long.



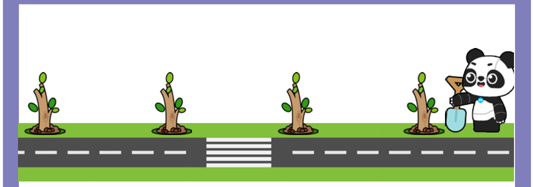
If the street is 20 meters long, then what should the panda do?

4 Repeat the steps 4 times.



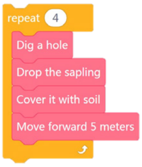
The panda will carry out the actions over and over again.

5 In programming, a loop refers to a piece of code that is executed repeatedly by the computer.



In computer programming, a loop means the computer repeats a piece of code over and over again.

6 Use the counting loop block to repeat a piece of code specific times.



```
repeat 4
    Dig a hole
    Drop the sapling
    Cover it with soil
    Move forward 5 meters
```

You can use the Counting Loop block to repeat a piece of code specific times. For instance, the panda will repeat the steps to plant trees four times.

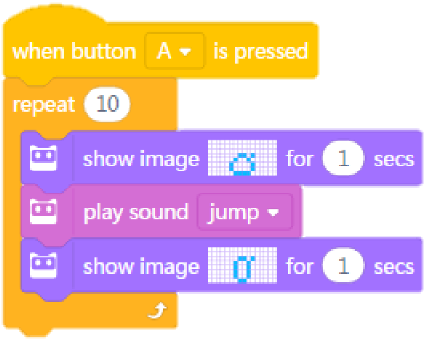
Learn through play

1. A represents tapping your left leg with the left hand; B represents tapping your right leg with the right hand; C represents tapping your legs with both hands. Follow the beats:
ABABABC ABABABC
Think: How to make the beats simpler if we want to make the game easier to play?
2. What if we want to add a beat D? It represents clapping your hands. Try it out.
ABABABC ABABABC ABCABC ABABABC ABABABD ABABABD
ABDABD ABABABD ABABABC ABABABC ABCABC ABABABC
ABABABD ABABABD ABDABD ABABABD

We can use a loop to make the beats look neat and the game easier to play. In programming, the Repeat block is frequently used to make the code neat. With the Repeat block, Codey Rocky can carry out the same sequence of actions repeatedly. When the Repeat block is executed once, we count it as one cycle or one iteration.

Coding Practice

The Steamed Bread Can't Jump



```
when button A ▾ is pressed
repeat 10
    show image for 1 secs
    play sound jump ▾
    show image for 1 secs
```

Story Line: The steamed bread is persisting in learning how to jump. It tried many times but failed to make it.

Imitate and Create

Try using the Counting Loop block to create an animation yourself.

(1) Improve on the teacher's program. You can change the event, the sound or the number of times.

(2) Change the image. The software has many built-in images. Pick one.

(3) Give your animation a storyline.

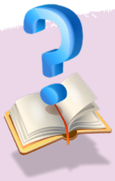
Share

In this lesson, we created a project ____.

During the process, we came into an issue____________________________________.

In the end, we solved the problem by _______________________________________.

Self-review



1.In this lesson, I learned that_________________________

2.The part I like most about this lesson is_____________________

3.In daily life, counting loops are all around us. For example,

(or you can draw it out in the space below) .

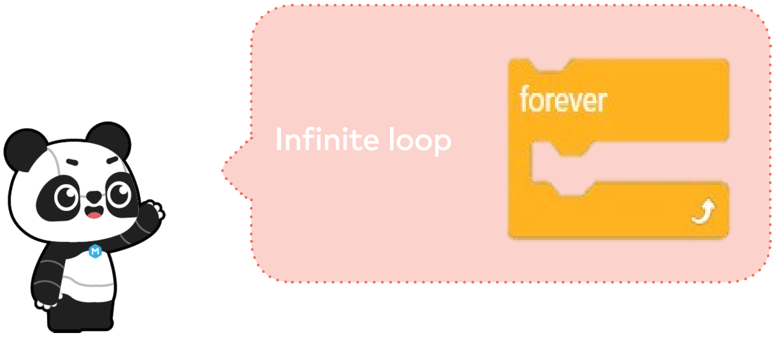
____

The Jumping Steamed Bread

Learning Objectives

1. Understand the concept of Infinite Loop.
2. Learn how to use the Forever block.
3. Use the Forever block to create your own animation.

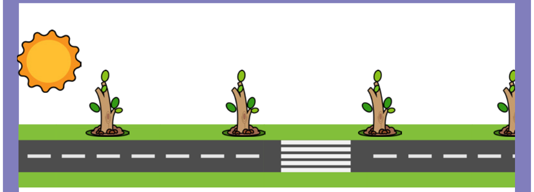
Key Concept



Infinite Loop refers to a piece of code that is repeated endlessly. Take a close look at the Forever block. It has not bump at the bottom, which means you can't add another block to the end. This is because the code inside the Forever block will be executed endlessly. Using the Forever block, you will be able to create projects that are engaging and fun.

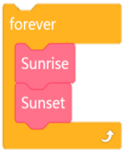
Situated Learning

1 A loop can be infinite,like the sunrise and the sunset.



In daily life, some events might happen just multiple times but some evens might repeat over and over again, for instance, sunset and sunrise.

2 Use the infinite loop block to repeat a piece of code forever.

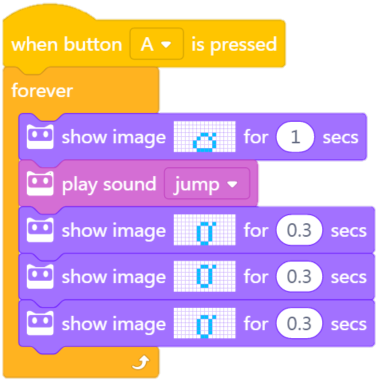


```
forever
    Sunrise
    Sunset
```

We use the Forever block to repeat a piece of code endlessly. For instance, we can simulate the sunrise and sunset using the Forever block.

Coding Practice

The Jumping Steamed Bread



```
when button A ▾ is pressed
forever
    show image [ ] for 1 secs
    play sound jump ▾
    show image [ ] for 0.3 secs
    show image [ ] for 0.3 secs
    show image [ ] for 0.3 secs
```

Story line: One day, the steamed bread finally knows how to jump. It keeps hopping and can go anywhere it likes. Here's how your program should look like:

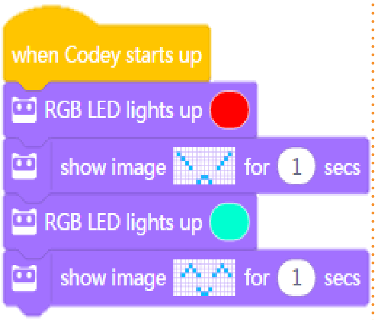
Imitate and Create

Use the Loop blocks to create animations of your own.

(1) Design two animations. Give each animation a specific event (button pressed). But use the Repeat block to create one animation and use the Forever block to program the other animation.

(2) You can give the animations storylines, like one being The Steamed Bread Can't Jump and the other being The Jumping Steamed Bread; or the two animations can be mutually independent.

(3) You can also spice up your animations by adding some sounds or lights.



```
when Codey starts up
RGB LED lights up ●
show image for 1 secs
RGB LED lights up ●
show image for 1 secs
```

Share

In this lesson, we created a project ____ .

During the process, we came into an issue____ .

In the end, we solved the problem by ____ .

Self-review

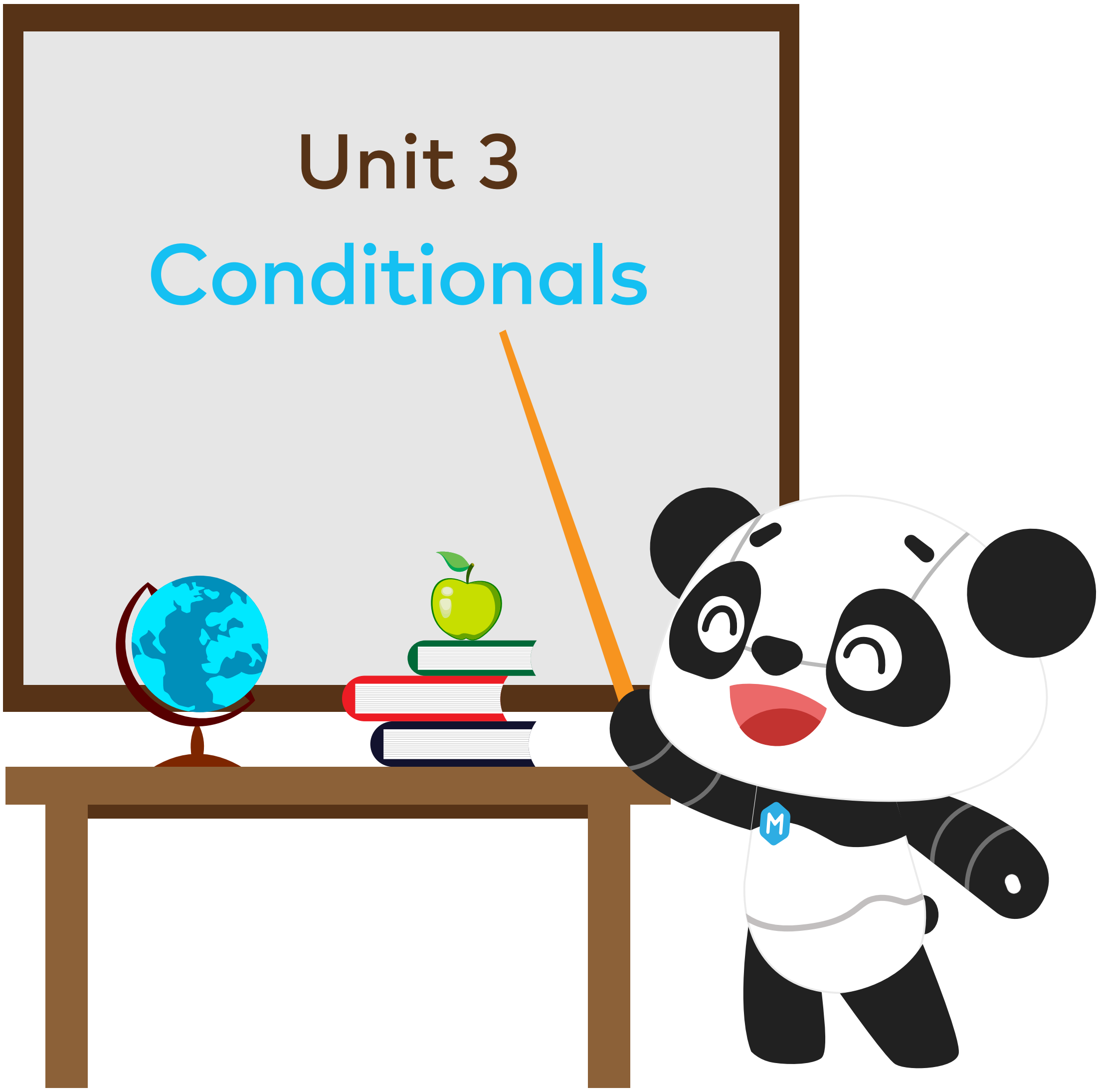
1.In this lesson, I learned that_______________________

2.The part I like most about this lesson is___________________

3.In daily life, infinite loops are all around us. For example,

(or you can draw it out in the space below) .

____



Unit 3

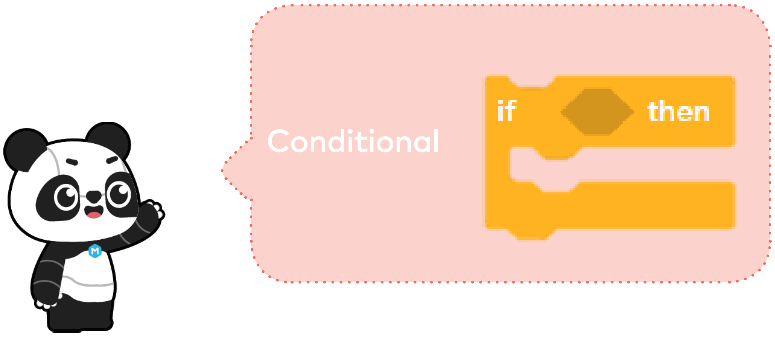
Conditionals

The Racing Game I

Learning Objectives

1. Understand the two concepts: Conditional and Boolean.
2. Use the Conditional blocks to complete tasks.
3. Identify the Color Sensor and the IR Proximity Sensor.

Key Concept



Conditional refers to the instructions that depend on whether something is true or false. The instruction will be executed only when the if statement is true; otherwise, the program will skip the instruction. In the conditional block, there is a hexagonal hole, you need to put the condition code inside the hole.

Hexagonal blocks return Boolean values (with only two possible values: true or false). A Boolean variable has only two values, 1(true) and 0 (false). If the if statement is true, the Boolean variable will return the value "true". Otherwise, it will return the value "false". Therefore, we also refer to the hexagonal blocks as Boolean blocks.

Situated Learning

1 The panda is going to buy a cup of tea. It opens the door to see whether it's raining.

The panda is going to buy a cup of tea. It opens the door to see whether it's raining.

2 If it's raining, it will put on its raincoat before leaving home.

What if it's raining? If it's raining, it will put on its raincoat and then leave home.

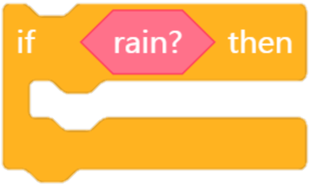
3 If it's not raining, the panda will go out straight away.

What if it's not raining? If it's not raining, the panda will go out straight away.

4 Conditional refers to the instructions that depend on whether something is true or false.

The instruction will be executed only when the if statement is true; otherwise, the instruction will be ignored.

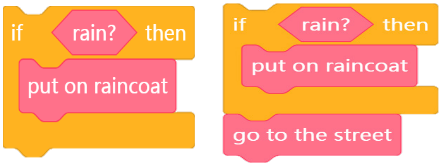
5 In programming, we use the if statement block to make decisions based on different situations.



```
if rain? then
```

First, add the condition block.

6 Add two instructions.

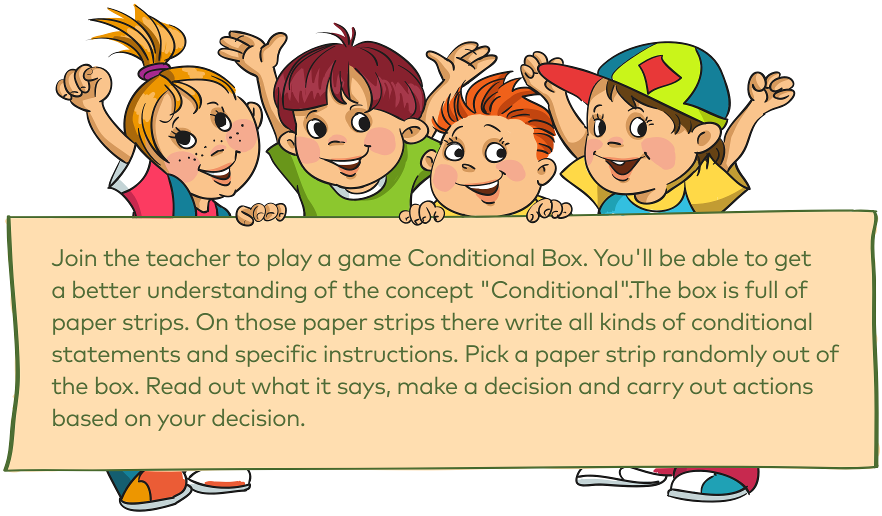


```
if rain? then
    put on raincoat
```

```
if rain? then
    put on raincoat
go to the street
```

Then, add the instruction that will be executed when the condition is true. Outside the if statement block is the instruction that will be executed when the condition is false.

Learn through play



Join the teacher to play a game Conditional Box. You'll be able to get a better understanding of the concept "Conditional".The box is full of paper strips. On those paper strips there write all kinds of conditional statements and specific instructions. Pick a paper strip randomly out of the box. Read out what it says, make a decision and carry out actions based on your decision.

Follow the steps:

1 The teacher will hand out some paper strips. Write down some conditional statements on the paper strips. Please make your statements easy to identify. Here are some examples for you: If you have long hair; if you wear glasses; if you are dressed in black; if your name includes the letter "A"; if you were born in June; if someone puts his or her hands up. Don't make your statements hard to identify (like "if it rains tomorrow" or "if the amount of your hair is an odd number"). These statements will be considered invalid.

2 Fold the paper in half and put it into the box;

3 Follow the teacher's instructions and pick a paper strip out of the box. Open it, read out loud what it says and make a decision. If the conditional statement is true, carry out the action. If it's false, then skip it;

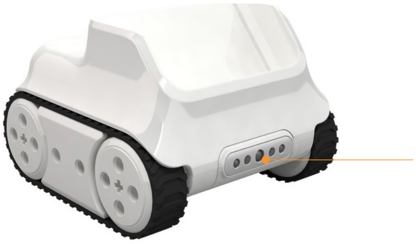
4 Fold the paper in half and put it back into the box. Go back to your seat or hand the box to the next student.

Coding Practice

Start off when the flag is waving

Before writing programs, you need to identify where the color sensor is located.

There is a row of sensors at the front bottom of Rocky, including Color Sensor, Grayscale Sensor, IR Proximity Sensor and other sensors.

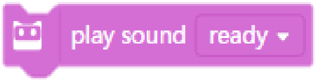


Infrared Color Sensor

The Color Sensor is in the middle. It can identify multiple colors, like red, green and blue.

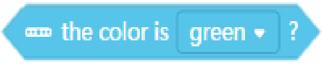
The game will begin soon. If Codey Rocky sees the green flag waving, it will start off at its top speed.

(1) When button A is pressed, Codey Rocky is getting ready at the starting line (play the sound ready).

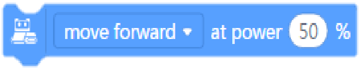


play sound ready ▾

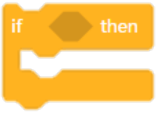
(2) If the color detected is green, Codey Rocky will move forward at its top speed.



the color is green ▾ ?

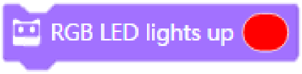


move forward ▾ at power 50 %



if then

(3) The RGB LED will turn red once the program makes a decision based on the situation.



RGB LED lights up 🟥

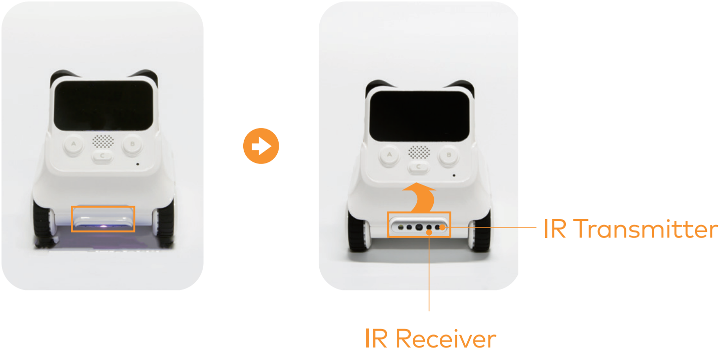
Extension activities:

(1) Give Codey Rocky facial expressions and sounds when it is running.

(2) If Codey Rocky identifies a red item, it will move backward.

Avoid the Obstacle

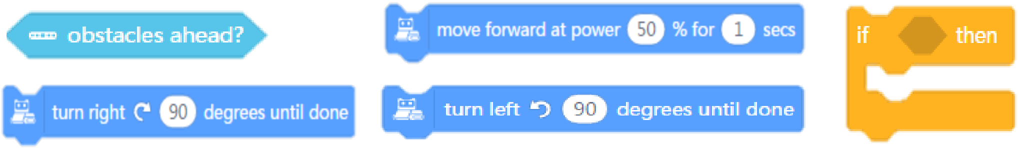
Before writing programs, let's first take a look at the IR Proximity Sensor. It's inside the Color Sensor (the left arrow indicates the IR receiver and the right arrow indicates the IR transmitter). With the IR Proximity Sensor, Codey Rocky is able to avoid obstacles. But you need to make sure the Color Sensor always faces forward when it is detecting obstacles.



When meeting an obstacle, Codey Rocky will avoid it and keep moving forward.

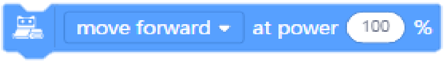
(1) Place an item in front of Codey Rocky.

(2) When the button A is pressed, if Codey Rocky detects an obstacle, it will turn right by 90 degrees, move forward, turn left by 90 degrees and keep moving forward at a rapid speed.



```
obstacles ahead?
move forward at power 50 % for 1 secs
if ◆ then
turn right ↻ 90 degrees until done
turn left ↺ 90 degrees until done
```

(3) If Codey Rocky fails to detect any items, it will move forward at its top speed.



```
move forward ▾ at power 100 %
```

Extension activities:

(1) Take a detour to avoid the obstacle, that is to say, turn left and then turn right.

(2) Let Codey Rocky display facial expressions and make a sound when it meets an obstacle.

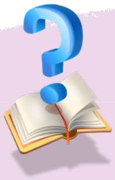
Share

In this lesson, we created a project ____

During the process, we came into an issue____

In the end, we solved the problem by ____

Self-review



1.In this lesson, I learned that_________________________

2.The part I like most about this lesson is_________________

3.Try to describe where the color sensor and the IR proximity sensor are located (or you can draw it out)

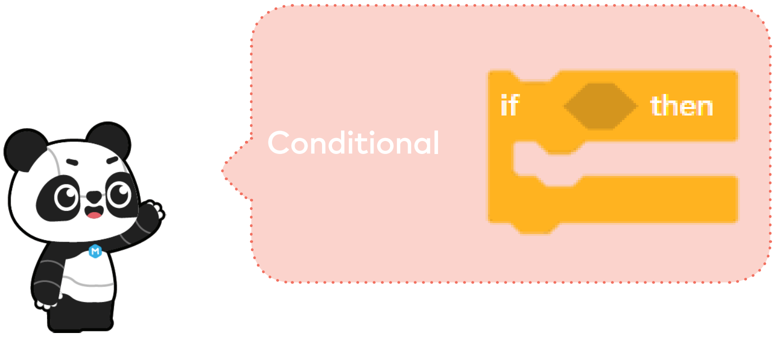
____

The Racing Game II

Learning Objectives

1. Learn how to use the if statement block and the repeat block to make your code concise.
2. Learn how to use the comparison operators inside the if statement block.
3.I dentify the Light Sensor and the RGB Indicator.

Review



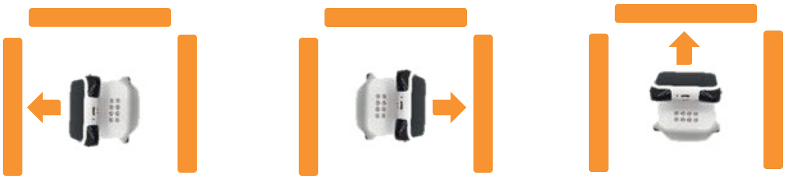
In programming languages, Conditional refers to the instructions that depend on whether something is true or false. The instruction will be executed only when the if statement is true; otherwise, the program will skip the instruction. In the if statement block, there is a hexagonal hole. You need to put the condition block inside the hole.

Coding Practice

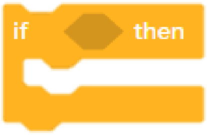
Service Station

During the race, Codey Rocky needs to refuel at service stations for plenty of times. However, the ways it pulls into the station might vary from time to time. Write programs to ensure that Codey Rocky is able to re-enter the race track at all events.

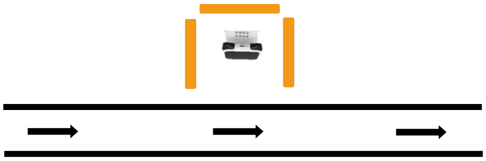
(1) Use books or other items to encircle your Codey Rocky as shown below. This is to simulate a scene in which Codey Rocky pulls over at a service station. The position of Codey Rocky will be one of the following randomly:



(2) Write programs: when the button A is pressed, if Codey Rocky meets an obstacle, it will keep making turns until it finds the exit. You need to use the if statement block more than once.



(3) In front of the exit is the race track. If Codey Rocky succeeds in finding the exit, it will turn left and run forward at its top speed.



Extension activities:

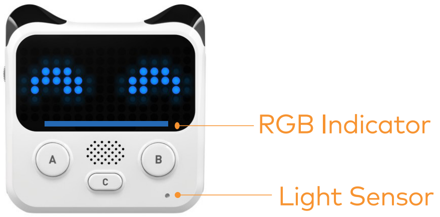
(1) Give Codey Rocky facial expressions, sounds and lights effects.

The Tunnel

Here's another challenge! But before working on the challenge, you need to take a look at the Light Sensor and the RGB Indicator. Besides, you have to learn about a new concept, Comparison Operators.

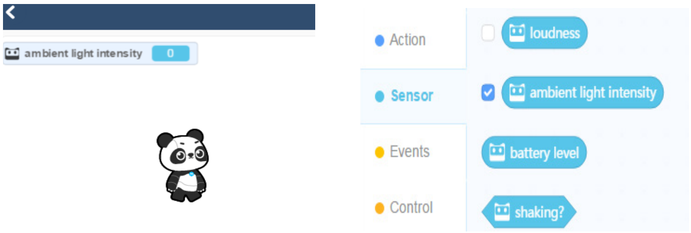
1. The Light Sensor and the RGB Indicator

At the bottom right corner of Codey, there is a black dot. The black dot is the Light Sensor. It is used to measure the light intensity of surrounding environments. In the middle of Codey, you will find the RGB Indicator. It can shine in colors.

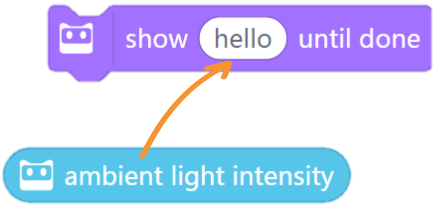


There are two ways to access the value of the Light Sensor:

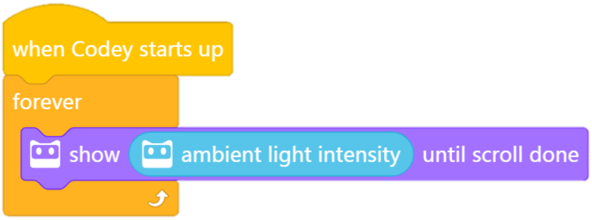
1 When the device is connected, tick the 【ambient light intensity】 block to display the value on the stage.



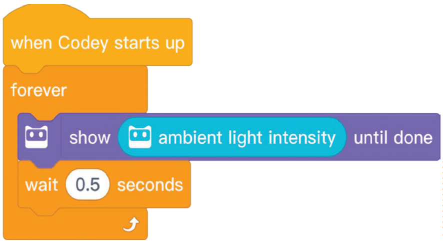
2 Or you can use the scripts below to display the value on Codey's LED screen. Upload the programs above to Codey. Observe how the value changes on the stage and on the LED screen.



```
show (hello) until done
ambient light intensity
```



```
when Codey starts up
forever
    show (ambient light intensity) until scroll done
```



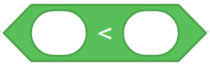
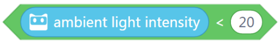
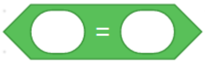
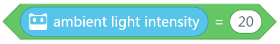
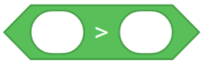
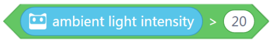
```
when Codey starts up
forever
    show (ambient light intensity) until done
    wait (0.5) seconds
```

tip： Add the wait block inside the forever block to make the value change at a slower speed.

2.Comparison Operators

In daily life, we need to compare values. Based on the results, we make decisions. Different decisions often lead to different actions. For instance, "If my temperature is beyond 38, I need to go to the doctor." Based on the thermometer value, we'll decide whether it's necessary to go to the doctor.

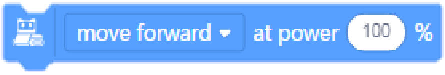
Similarly, we can use Comparison Operators in mBlock 5 to compare two values. The Comparison Operators include: < operator, = operator and > operator

Comparison Operators	Implication	Example	
 <	Less than	 ambient light intensity < 20	The ambient light intensity value is less than 20.
 =	Equality	 ambient light intensity = 20	The ambient light intensity value equals 20.
 >	Greater than	 ambient light intensity > 20	The ambient light intensity value is greater than 20.

When entering a dark tunnel, Codey Rocky will turn on its light and slow its speed.

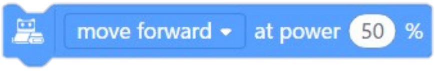
(1) Stick a black paper on the Light Sensor of Codey Rocky.

(2) When the button A is pressed, Codey Rocky will move forward at its top speed.

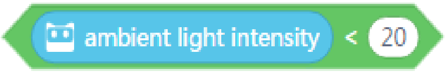


move forward ▾ at power 100 %

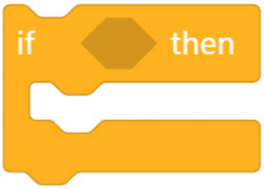
(3) If the light intensity is below 20, Codey Rocky will turn on its white RGB indicator and drive at a lower speed.



move forward ▾ at power 50 %



ambient light intensity < 20



if then

Extension activities:

(1) Add sounds and facial expressions to Codey when it's in a dark environment.

Share

In this lesson, we created a project ____

During the process, we came into an issue____

In the end, we solved the problem by ____

Self-review

1. In this lesson, I learned that_______________________

2. The part I like most about this lesson is_______________

3. Describe where the Light Sensor or the RGB Indicator is located (or you can draw it out in the space below).

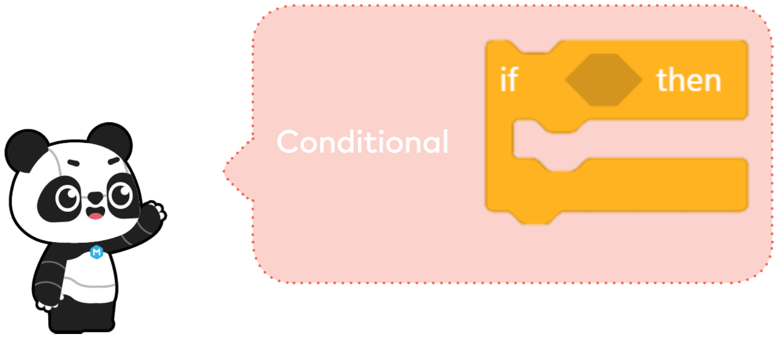
____

Volume Bar

Learning Objectives

1. Learn how to nest the if statement block inside the Forever block in your code.
2. Know how to complete the challenges using the Logic Operators and the if...then...else... block.
3. Able to identify where the Sound Sensor is located.

Review



In programming languages, Conditional refers to the instructions that depend on whether something is true or false. The instruction will be executed only when the if statement is true; otherwise, the program will skip the instruction. In the if statement block, there is a hexagonal hole. You need to put the condition block inside the hole.

Coding Practice

Volume Bar

Before writing programs, let's take a look at the Logic Operators and the Sound Sensor of Codey Rocky.

1. Logic Operators

Using a combination of logic operators and comparison operators, you can form different value ranges. The logical operators can be divided into three types: "and" "or" " not".

1. Logical operator "and": The logical operator "and" is used with two Boolean values. Only when the two Boolean values are true will the logical operator "and" returns the result "true", otherwise the result will be "false". If you want to set the range of loudness to be "10<x<20", you'll need to use the logical operator "and".

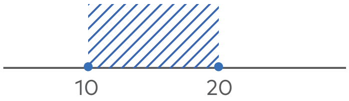
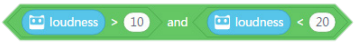


Figure 1

Here's an example: If the loudness falls among the range of 10~20, the logical operator "and" will return a value "true" and the instruction inside the if statement block will be executed.



```
loudness > 10 and loudness < 20
```

Figure 2

2. Logical operator "or": The logical operator "or" is used with two Boolean values as well. The logical operator "or" will return the result "true" when one or both of the Boolean values is true; the logical operator "or" will return the result "false" only when both of Boolean values are "false". If you want to set the loudness range to be "x<10 or x>20", you'll need the logical operator "or".

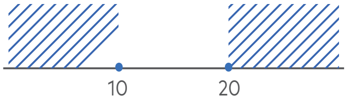
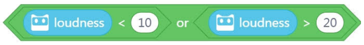


Figure 3

Here's an example: If the loudness is beyond 10 or is below 20, the logical operator "or" will return a value "true" and the instruction inside the if statement block will be executed.



```
loudness < 10 or loudness > 20
```

Figure 4

3. Logical operator "not": The logical operator "not" is used with only one Boolean value. When the Boolean value is "false", the logical operator "not" will return a value "true"; when the Boolean value is "true", the logical operator will return a value "false". If you want to set the loudness range to be "x≥20", you need to use the logical operator "not".

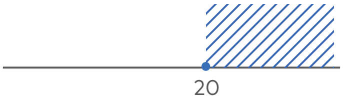
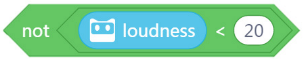


Figure 5

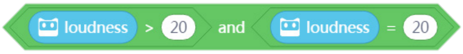
Here's an example: If the loudness is not less than 20 (that is to say, ≥20), the instruction inside the if statement block will be executed.



not loudness < 20

Figure 6

It has the same effect as the block which is shown in the Figure 7.



loudness > 20 and loudness = 20

Figure 7

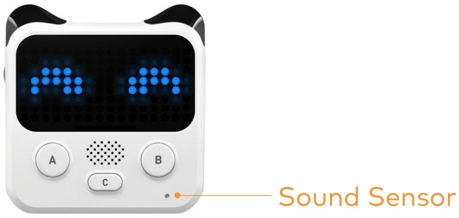
The figure below is a wrap-up.

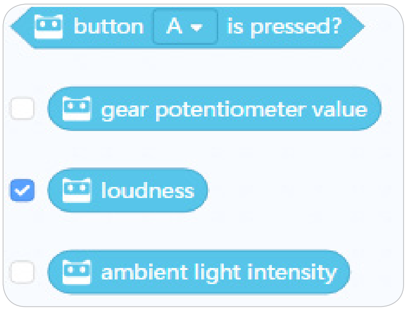
Logical operators	Implications
and	The result will be true when both of the Boolean values are true.
or	The result will be true when either of the Boolean values is true.
not	The result will be true when the Boolean value is false.

2. Sound Sensor

The Sound Sensor is located at the bottom right corner of the Codey as well.

The sensor is used to measure the loudness of sounds in surrounding environments.





button A ▾ is pressed?

□ gear potentiometer value

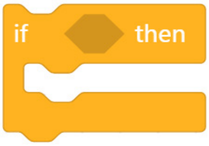
☑ loudness

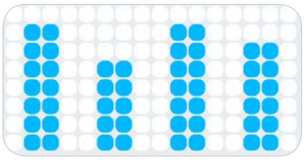
□ ambient light intensity

Imitate and Create

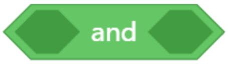
Codey Rocky wins the game and people are applauding. The volume bar on the screen will change its height based on the loudness of the applause.

(1) When Codey Rocky starts up, if the applause loudness exceeds 20, the volume bar will reach its top height.



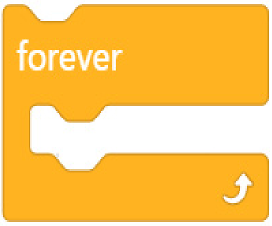


(2) If the loudness falls between 10 and 20, the volume bar will go down. In this case, you need to use the "and" block.



(3) If the applause loudness is below 10, the volume bar will fall to its lowest height.

(4) Encircle all the code with a "forever" block to make Codey Rocky keep detecting the sound around it.



Extension tasks:

(1) Program the RGB LED indicator to change its color based on the sound volume;

(2) Make the Volume Bar more sensitive by subdividing the value range of loudness.

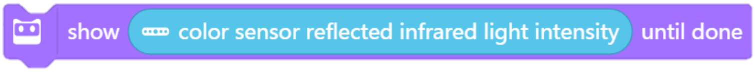
Imagine Codey is a cute cat. When you approach it, it will follow you; otherwise, it will stay there waiting for you.

(1) The IR Color Sensor is used to measure the intensity of reflected IR lights. The closer an item is to Codey Rocky, the higher the light intensity will be.

Infrared Color Sensor

Multiple sensors are integrated into one single Infrared Color Sensor, including a color sensor, a grayscale sensor, an IR proximity sensor and more. You can toggle the sensor up or down to make Codey Rocky perform a variety of tasks, like avoiding obstacles or following lines.

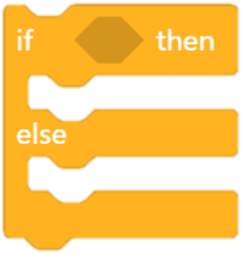
(2) When Codey Rocky starts up, the LED Panel will display the intensity value of the reflected infrared light.



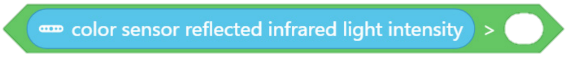
```
show (color sensor reflected infrared light intensity) until done
```

(3) Put the show block inside the forever block. In this way, Codey Rocky is able to monitor the reflected infrared lights in real time.

(4) Add an if statement block to the end of the show block inside the forever block. If the light intensity is beyond a certain value (customizable), Codey Rocky will run forward at its top speed. Otherwise, Codey Rocky will just stay there. In this case, you need to use the **if...then...else...** block and the Operator block ">".



```
if < > then

else

```



```
< (color sensor reflected infrared light intensity) > ( ) >
```

Extension activities: If the loudness exceeds one certain value, Codey Rocky moves forward or turns left at the greatest power.

Think:: What will happen if you make the IR color sensor face down?

Share

In this lesson, we created a project ____

During the process, we came into an issue____________________________________

In the end, we solved the problem by ______________________________________

Self-review

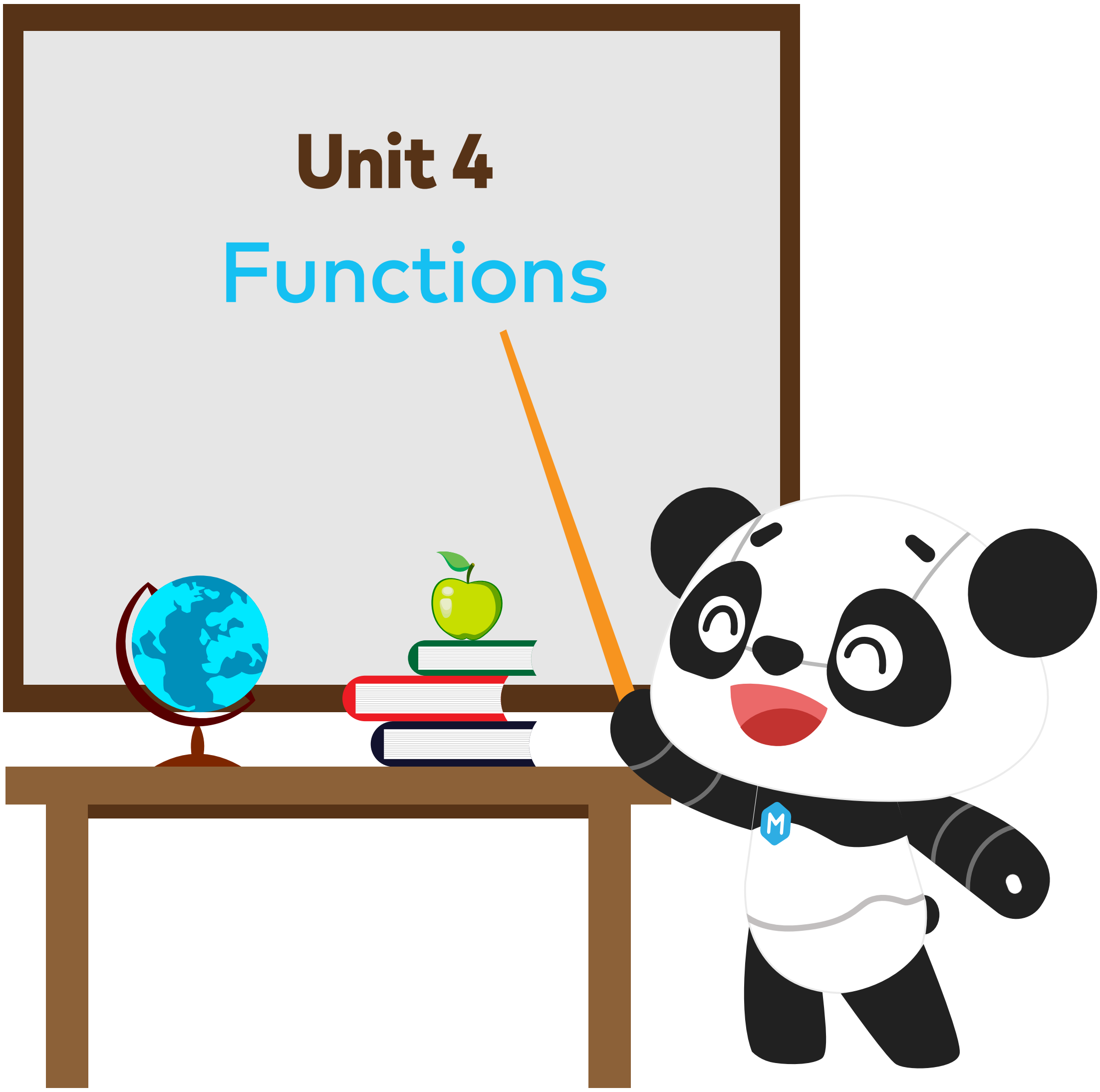
1. In this lesson, I learned that______________________

2. The part I like most about this lesson is__________________

3. Describe where the Sound Sensor is located (or you can draw it out).__________________________________

Unit 4

Functions

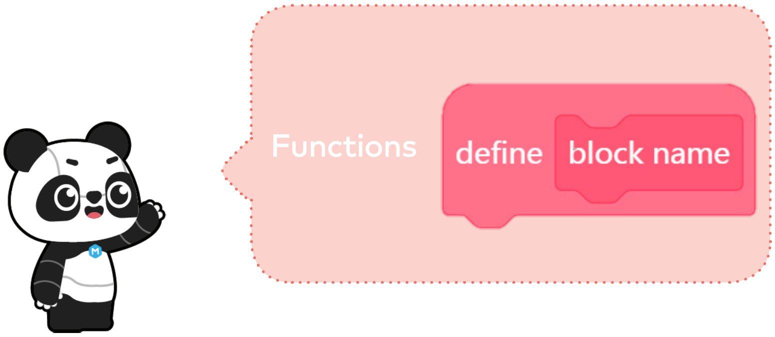


Good Morning! Functions

Learning Objectives

1. Understand the concept of Functions.
2. Grasp how to apply the Functions block (My Blocks) in your code.
3. Create a function and call the function in the code to give Codey a custom boot animation.

Key Concept



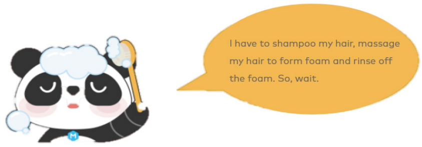
In programming, Function is a custom coding block. Function refers to a set of instructions that can be called repeatedly in the code.

Situated Learning

1 The steps to wash hair

Washing hair requires three steps: shampoo your hair, massage hair to form foam and rinse foam off. But if we don't use a phrase "wash hair" to describe the set of steps, what will be the situation then?

2 When friends are asking the panda out, it will say:



You might come into this situation: When a friend comes to ask the panda out, it'll say: "I have to shampoo my hair, massage my hair to form foam and rinse off the foam. So, wait."

3 Name the set of actions as "wash hair".

In everyday life, we often give a set of actions a single name and use the name to refer to the whole set of actions when necessary.After we name the set of actions as "wash hair",the situation will be like:

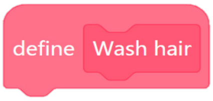
4 When friends are asking the panda out, it will say:

When your friends are asking you out, you will say, "I'm going to wash hair. Wait."

5 Make the communication simple and pleasant.

Use a simple phrase to name the set of actions. The name is what we call a function.

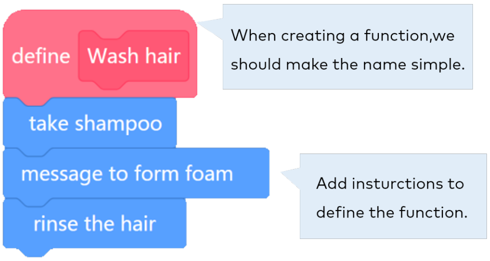
6 In programming, we use a function to name a set of instructions.



```
define Wash hair
```

In programming, we use a function to name a set of instructions and call the function in the code if necessary. The first thing to create a function is to give the function a name. Next, you need to define the function by adding instructions.

7 The name of a function needs to be simple.

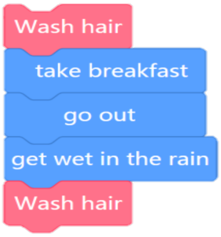


```
define Wash hair
take shampoo
message to form foam
rinse the hair
```

When creating a function,we should make the name simple.

Add insturctions to define the function.

8 Use functions to make programs neat.



```
Wash hair
take breakfast
go out
get wet in the rain
Wash hair
```

A function is used to refer to a series of instructions. By using functions, we are able to make our code concise and thusly easy to understand. If necessary, programmers can call the functions in the code any time.

Learn Through Play

Join the teacher to play the game Functions of Morning. This will help you have a clearer picture of the concept of Functions.

Functions of Morning, as the name suggests, are a group of functions that indicate what you do in the morning every day, from getting up to leaving home for school. Now, you need to give the following functions names based on the instructions. Then call these functions to describe the whole process of what you do in everyday morning, from getting up to leaving home for school.

Name		Get the left feet into the left shoe; Get the right feet into the right shoe and stand up.
Name		Take the alarm clock; Take a look at the time; Put down the alarm clock.
Name		Stretch yourself; Uncover the quilt; Sit up.
Name		Cover head with the quilt; Wait for 5 seconds; Uncover the quilt; Stretch out one hand to stop the alarm clock;
Name		Put the left arm into the left sleeve; Put the right arm into the right sleeve; If it's a T-shirt, get your head into the shirt; If the coat has buttons, fasten the buttons.
Name		Grab the toothbrush; Squeeze the toothpaste; Rinse the mouth; Brush the teeth; Rinse the mouth.
Name		Get the left feet into the left shoe; Get the right feet into the right shoe; Lace the left shoe; Lace the right shoe.
Name		If it's a T-shirt, take it off from top; If the coat has buttons, unbutton it first, then get the left arm out of the left sleeve, then the right arm out of the right sleeve.
Name		Lift the left foot up; Step on the ground; Lift the right foot up; Step on the ground.
Name		Take off the coat; Take off pants.
Name		Pull down the pants; Get the left leg out of the pants; Get the right leg out of the pants.
Name		Get the left leg into the left pant leg; The right leg goes into the right pant leg; Pull up.

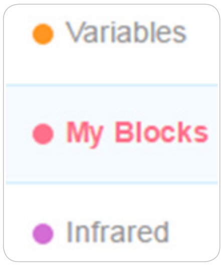
Coding Practice

Startup Animation of Codey

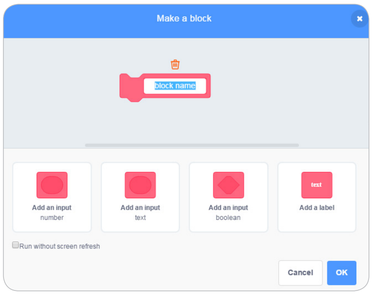
Create a startup function for Codey Rocky, making sure the function will automatically run when Codey Rocky starts up. Open the mBlock 5 and connect Codey to the software. Follow the instructions to complete the challenge.

Task: **Create a startup function for Codey Rocky**

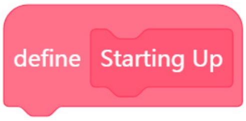
(1) Click My Blocks at the category bar and select Make a Block.



(2) Click Make a Block. Create a function and give it a name.



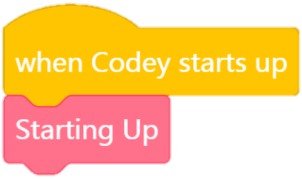
(3) Then, the **define starting up** block will appear in the Scripts area.



```
define Starting Up
```

(4) What instructions should be run when Codey Rocky starts up? Design programs under the define starting up block.

(5) After defining the function, you can directly call the function by adding the **starting up** block to the bottom of the event block **when Codey Rocky starts up**.



```
when Codey starts up
Starting Up
```

Imitate and Create

Create a startup animation using the function block.

(1) Improve on the sample project. You can change the animation or the sound;

(2) Select an image from the built-in images and refine the image as you would like. Then create your own animation using the image.

(3) Give your animation a compelling story line.

Share

In this lesson, we created a project ____

During the process, we came into an issue____

In the end, we solved the problem by ____

Self-review

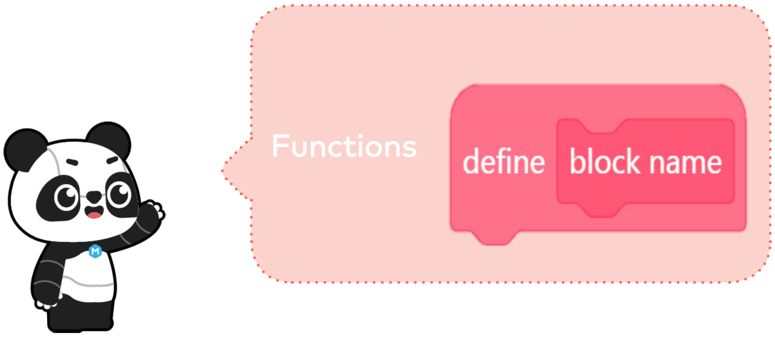
1. In this lesson, I learned that_____________________
2. The part I like most about this lesson is________________

The Tiny Patroller I

Learning Objectives

1. Master how to use the Functions blocks (My Blocks) in your code.
2. Know how to create a function and call the function.
3. Able to complete the challenge by applying functions and mathematical knowledge.

Review

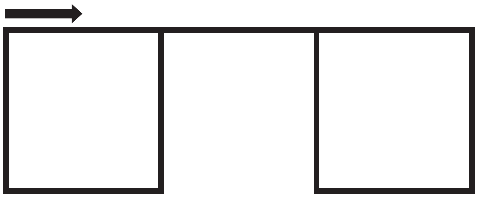


In programming, Function is a custom coding block. Function refers to a set of instructions that can be called repeatedly in the code.

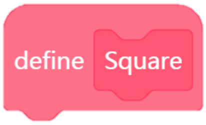
Coding Practice

Imagine Codey Rocky is a security guard. It is patrolling the passages in the building to make sure that all the properties are safe. Now, it's patrolling on the 1st floor.

(1) Design programs to make Codey Rocky follow the black lines as shown below.

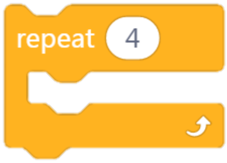


(2) Create a function and name it as Square.

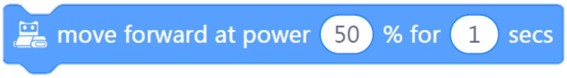


define Square

(3) You might need to use the following blocks.



repeat 4



move forward at power 50 % for 1 secs

(4) Figure out how much time it takes Codey Rocky to drive along the square-shaped line and along the linking line.

✓ Measure the length of one side of the square.

✓ Measure the length of the linking line.

✓ Calculate how much time it takes Codey Rocky to drive along a square.

✓ Calculate how much time it takes Codey Rocky to drive along the linking line.

✓ Calculate the speed of Codey Rocky (per second). (For instance, if Codey Rocky is programmed to keep moving forward at power of 50% for 1 second, what is the driving distance then?)

(5) Call the function Square two times under the event block when button A pressed.

Extension activities:

Add facial expressions, sounds and lights effects to Codey Rocky.

Notes

(1) The first step is to measure the length and width of the roadmap. Based on the measurements, you need to figure out how much time it takes Codey Rocky to complete the route.

(2) The motor of Codey Rocky is a DC motor so Codey Rocky is not able to make turns or follow lines precisely. In this case, it's fine that Codey Rocky follows the line in a rough manner.

(3) Codey Rocky is programmed to move forward and turn right. After Codey Rocky repeats the behaviors four times, it will go back to the starting position.

(4) Since there are two squares in the roadmap, you are supposed to call the function Square at least twice in the code.

Share

In this lesson, we created a project ____

During the process, we came into an issue___________________________________

In the end, we solved the problem by ______________________________________

Self-review

1. In this lesson, I learned that_____________________

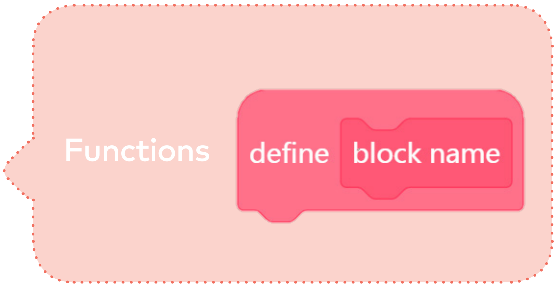
2. The part I like most about this lesson is________________

The Tiny Patroller II

Learning Objectives

1. Master how to use the Functions blocks (My Blocks) in your code.
2. Know how to create a function and call the function.
3. Able to complete the challenge by applying functions and mathematical knowledge.

Review



In programming, Function is a custom coding block. Function refers to a set of instructions that can be called repeatedly in the code.

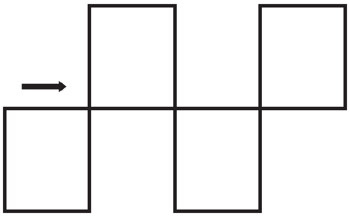
Coding Practice

In the previous lesson, Codey Rocky is programmed to patrol on the 1st floor. In this lesson, you face a greater challenge, which is to have Codey Rocky patrol on the 2nd floor.

Follow the teacher's instructions and what the Challenge Card says. Try completing the challenge!

Now, Codey Rocky comes to the 2nd floor. There are more rooms and the route is more complex.

(1) Design programs to make Codey Rocky drive along the black line as shown below.



(2) You need to create two functions, Upper square and Bottom square.

(3) Figure out how much time it takes Codey Rocky to drive along the square-shaped line and along the linking line.

✓ Measure the length of one side of the square. ✓ Measure the length of the linking line.

✓ Calculate how much time it takes Codey Rocky to drive along a square.

✓ Calculate how much time it takes Codey Rocky to drive along the linking line.

✓ Calculate the speed of Codey Rocky (per second). (For instance, if Codey Rocky is programmed to keep moving forward at power of 50% for 1 second, what is the driving distance then?)

(4) Call the function Square two times under the **Events** block **when button A is pressed**. Using the repeat block will make your code concise.

Extension activities:

Add facial expressions, sounds and lights effects to Codey Rocky.

Notes

(1) Remember to create two functions, Upper square and Bottom square, in your code.

(2) Measure the length and width of the roadmap. Based on the measurements, you need to figure out how much time it takes Codey Rocky to complete the route.

(3) The motor of Codey Rocky is a DC motor so Codey Rocky is not able to make turns or follow lines precisely. In this case, all you need to do is to make sure your Codey Rocky follows the line in a rough manner.

(4) There are multiple ways to make Codey Rocky take the route as shown above. You can work on your own to figure out the solution first; or you can complete the challenge based on the following pseudocode:

Bottom square	Upper square	When button A pressed
move forward 1 second turn right by 90° Repeat four times	move forward 1 second turn left by 90° Repeat four times	bottom square move forward 1 second upper square move forward 1 second Repeat two times

Share

In this lesson, we created a project ____

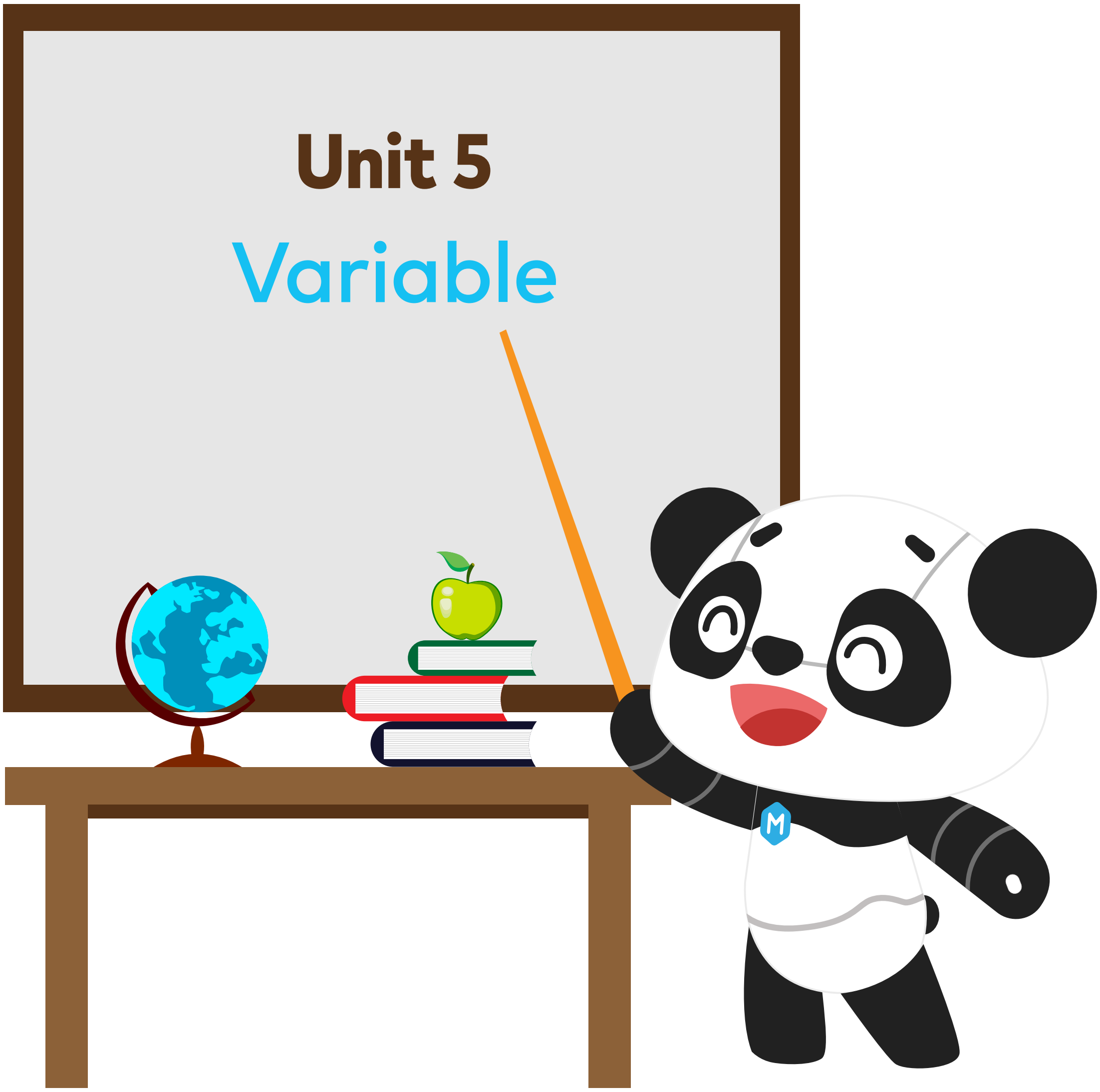
During the process, we came into an issue______________________________________

In the end, we solved the problem by ____

Self-review

1. In this lesson, I learned that______________________

2. The part I like most about this lesson is__________________



Unit 5

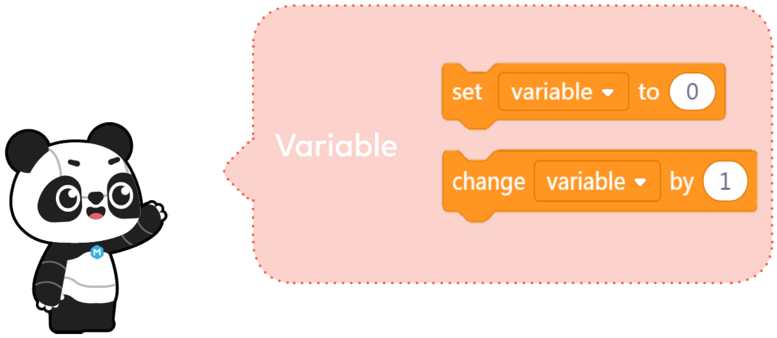
Variable

The Squirrel's Nuts Box

Learning Objectives

1. Understand the concept of Variable.
2. Learn how to create a variable and use the variable in the code.

Key Concept



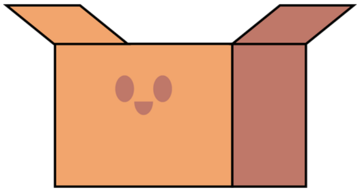
Variable

```
set variable ▾ to 0
change variable ▾ by 1
```

Imagine that a variable is a box. You can put a value into the box and replace it with another value anytime. It works like a scoreboard which is used to record the scores during a contest. As the contest proceeds, the scores keep changing. So you can change the scores on the board accordingly. In computer programming, a variable is a storage location paired with an associated name. When a variable is created, the computer will leave a storage location to store the variable and give the location a symbolic name. By calling the variable name, programs can read and change the value in the storage location.

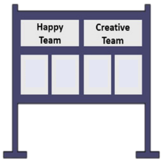
Situated Learning

1 Variable is a container



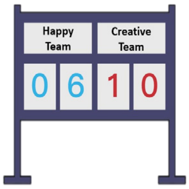
A variable is a container which stores information that can be changed.

2 A simple name helps to ensure the variable can work properly.



You need to give the variable an appropriate and simple name so that it is easy to understand. We name the two variables as Happy Team and Creative Team.

3 The scoreboard is a container that stores scores in a match.

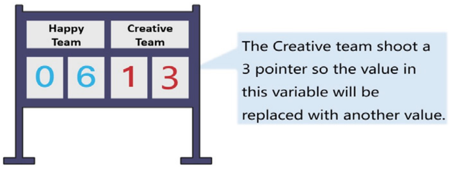


The scores on the scoreboard can act as two variables. The values stored in the two variables are the scores of the two teams.

4 When either of the teams gets a shot, the scores will....

The Creative team shoots a 3 pointer so how will the score change?

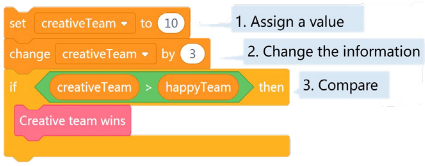
5 The value of this variable will be replaced with another value.



The Creative team shoot a 3 pointer so the value in this variable will be replaced with another value.

The value of this variable will be replaced with another value. It can be seen that the score of the Creative Team is higher than that of Happy Team.

6 The purpose of a variable.



```
set creativeTeam ▾ to 10
change creativeTeam ▾ by 3
if creativeTeam > happyTeam then
    Creative team wins
```

1. Assign a value
2. Change the information
3. Compare

The purpose of a variable is to assign values, change the information and compare values.

Learn Through Play

Join the teacher to play the game The Squirrel's Nuts Box.

Take these steps:

(1) There is a little squirrel. He has a nuts box. Every day, the squirrel puts nuts into the box and takes nuts out of the box, so the number of the nuts keeps changing all the time. One day morning, the little squirrel gets up, opens the box and finds out that there are 10 nuts left. The amount of the nuts will change based on different situations.

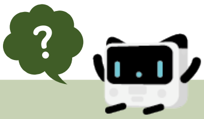
(2) For instance, the squirrel is hungry and eats two nuts, so there are 8 nuts left. Then the squirrel picks another 5 nuts so there are 13 nuts in total. Now let's play a game, counting how many nuts are left.

(3) The teacher writes down the number 10 on the blackboard and tells students there are 10 nuts left in the box.

(4) Students take turns picking a paper strip out of the box.

(5) Students read out what the paper strips say and try to figure out the amount of the nuts left based on the information. Then write down the result on the blackboard.

In the morning, the little squirrel eats 2 nuts for breakfast.
If the maximum temperature exceeds 5°C , the squirrel will pick 10 nuts in the wild.
At noon, the squirrel eats 3 nuts for lunch.
In the afternoon, a little bird pays a visit and gives the squirrel 5 nuts.
Today is the monkey's birthday. The squirrel gives him 3 nuts as a gift.
In the evening, a rat steals 4 nuts.
The squirrel plays the rock-paper-scissors game with a friend three times. If he loses the game, he gives away a nut; if he wins the game, he gets a nut; if it's a draw, then no gain or loss for both sides.
In the evening, the squirrel eats only 1 nut to keep fit.
It's hot today. The squirrel buys a straw hat from Aunt Bear. It costs him 7 nuts.
If the amount of the nuts in the box is less than 5, the squirrel will pick 10 nuts outdoors.
If the amount of the nuts in the box exceeds 12, the little squirrel will turn in a circle happily.



Think:

(1) What's the variable in the game? How did you name it?

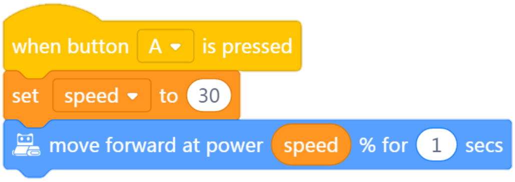
(2) What's the initial value for the variable?

(3) What operations were done with the value?

Coding Practice

Beginner Project

Using the variable to assign a value, you can make Codey Rocky move forward at any speed you would like. When the button A is pressed, set the speed to 30, and Codey Rocky will move forward at a speed as set for 1 second.

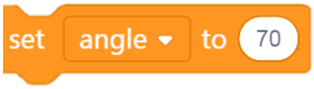


```
when button A ▾ is pressed
set speed ▾ to 30
move forward at power speed % for 1 secs
```

Codey Rocky, standing on the stage, is saying hello to the audience.

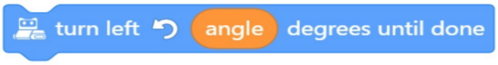
(1) Codey Rocky turns left 70 degrees and then turns right 140 degrees. Finally, it returns to the starting point. To make Codey Rocky turn specific degrees, you need to create a variable named "angle".

(2) Set the angle to the value of 70.



```
set angle ▾ to 70
```

(3) Make Codey Rocky turn left by specific degrees as programmed.



```
turn left ↺ angle degrees until done
```

(4) Assign a new value to the variable "angle". As programmed, Codey Rocky will turn right 140 degrees and then turn left 70 degrees.

(5) Add facial expressions, sounds and light effects to Codey Rocky.

Imitate and Create

(1) Improve on the sample project. You can reset the value and the facial expressions.

(2) By using different combinations of **Events** blocks and variables, you can make Codey Rocky drive in the way you'd like.

(3) Design a program that includes 3 variables. Use the coding blocks you learned previously to create programs.

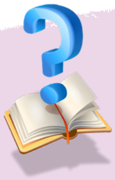
>>> Share

In this lesson, we created a project ____

During the process, we came into an issue____

In the end, we solved the problem by ____

>>> Self-review



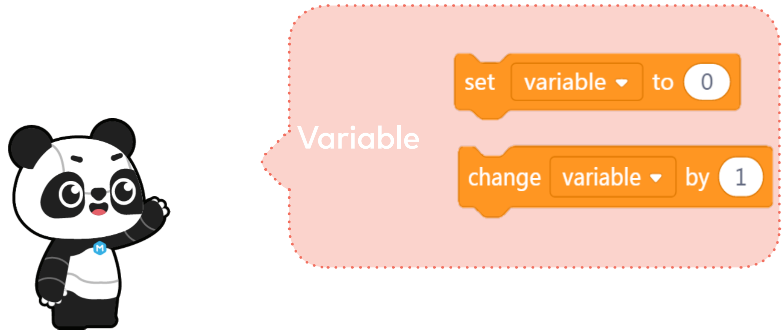
1. In this lesson, I learned that_______________________
2. The part I like most about this lesson is________________
3. In daily life, variables are around us. One example is (or you can draw it out in the space below)______________

Mathematical Operations

Learning Objectives

1. Understand the concept of Variables.
2. Learn how to assign values to variables and how to do operations with the values.

Review



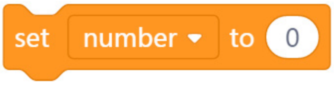
```
set variable ▾ to 0
change variable ▾ by 1
```

In computer programming, a variable is a container that can store information. You can duplicate the variable, change the information stored in the variable and make comparisons between variables.

Coding Practice

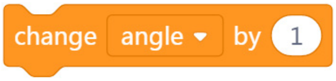
Codey Rocky is learning how to do mathematical operations. Now it masters how to add 1 to a value and subtract 1 from a value.

(1) Codey Rocky would like to start with the number 0, so the value of the variable "number" is set to be 0 when it starts up.



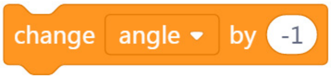
```
set number ▾ to 0
```

(2) When button A is pressed, the number increases by 1.



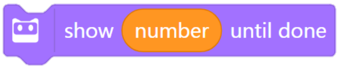
```
change angle ▾ by 1
```

(3) When button B is pressed, the number decreases by 1. A negative number refers to subtracting.



```
change angle ▾ by -1
```

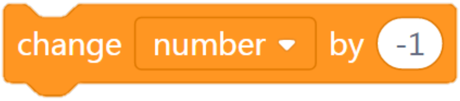
(4) If you would like to make the LED screen show the calculation result, you can add the **block show ( ) until done** under each event block.



```
show number until done
```

Note

(1) In the **change (number) by ( )** block, negative numbers represent subtracting.



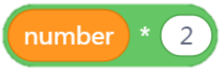
```
change number ▾ by -1
```

(2) The parameter dent has three kinds of shapes: hexagon, rectangle, and round rectangle. However, a variable block can only fit into the round rectangle kind.

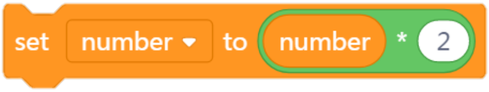
Codey Rocky also learned how to multiply and divide.

(1) Assign an initial value to the variable.

(2) When button A is pressed, the number is multiplied by 2. We need to use the **Operators** block multiply here to get a calculation result. Then, assign the result to the variable.



```
number * 2
```



```
set number ▾ to number * 2
```

(3) When button B is pressed, the number is divided by 2.

(4) Show the calculation result on the LED display each time. Remember, the LED display can only show a value ranging from -999 to 9999.

Extension activities:

Change the initial value and the factor (replace the number 2 with another value).

Note

(1) To do operations with the variable, you'll need to use the Operators blocks to get a calculation result. Then replace the previous value by reassigning the calculation result to the variable.

(2) Make sure the variable you use under the three different events (when Codey Rocky starts up; when button A is pressed; when button B is pressed) is the same one.

Share

In this lesson, we created a project ____

During the process, we came into an issue___________________________________

In the end, we solved the problem by _______________________________________

Self-review

1. In this lesson, I learned that________________________
2. The part I like most about this lesson is________________

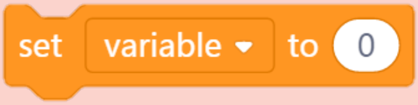
The Bomb

Learning Objectives

1. Understand the concept of Variables.
2. Know how to increase or decrease the value of a variable.

Review

Variable



```
set variable ▾ to 0
```

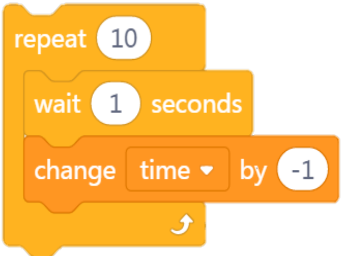
With the Operators blocks, you can use variables to store the calculation results.

Coding Practice

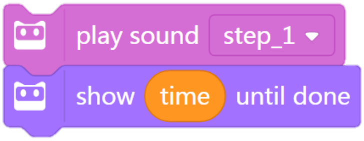
Codey is counting down. During the period, anyone who is holding Codey has to give an animal name. The game ends the moment the bomb explodes.

(1) When Codey Rocky starts up, the initial value of the variable "time" is set to be 30.

(2) When button A is pressed, the "timer" will start counting down. After 1 second, the value of "time" will decrease by 1. Repeat the piece of code 30 times. Add sounds effects to make the game more intense and write code to make the LED screen show how much time is left.



```
repeat 10
    wait 1 seconds
    change time by -1
```



```
play sound step_1
show time until done
```

(3) During the countdown period, anyone who is holding Codey has to give an animal name and then passes Codey to the next person. Animal names cannot be repeated.

(4) When the value of "time" becomes 0, the countdown ends and the bomb explodes. At this moment, the RGB LED indicator lights up in red.

Extension activities:

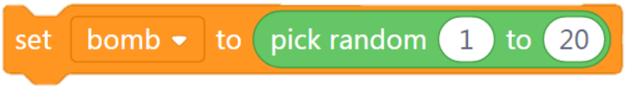
Change the initial value of the variable "time" to make the game time longer or shorter as you would like.

When the button A is pressed, Codey will generate a number randomly. Two players play the game rock-paper-scissors. Anyone who loses the game presses the button B and the number on the LED screen will increase by 1. Take turns pressing the button. When the LED screen shows a number that equals the randomly generated number, Boom! The bomb explodes.

(1) You need to create two variables: bomb and number. The variable "bomb" stands for the randomly generated number. The variable "number" represents the number that keeps changing during the game (starting from 0). The bomb will explode once the two variables output the same value.

(2) When Codey Rocky starts up, both of the initial values are 0.

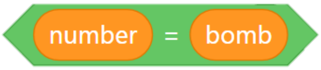
(3) When button A is pressed, the value of "bomb" is set to be a random number ranging from 1 to 20. Meanwhile, Codey puts on its sunglasses and plays the sound "ready".



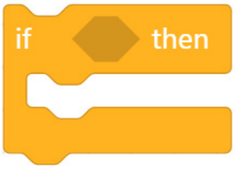
```
set bomb ▾ to pick random 1 to 20
```

(4) Two persons play the game rock-paper-scissors. Anyone who loses the game presses the button B. When the button is pressed, the value on the LED display increases by 1.

(5) If the value of the variable "number" equals the "bomb" value, Codey will display an image of bomb and plays the sound accordingly. To assess whether the two values equal each other, you might need to use the operator block comparison and the if statement block in your code.



```
number = bomb
```



```
if ⬡ then
```

Extension activities:

Redefine the random range of the variable "bomb".

Share

In this lesson, we created a project ____

During the process, we came into an issue___________________________________

In the end, we solved the problem by ______________________________________

Self-review

1. In this lesson, I learned that___________________

2. The part I like most about this lesson is_____________

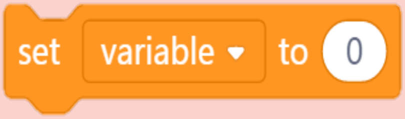
Rock-Paper-Scissors

Learning Objectives

1. Understand the concept of Variables.
2. Make comparisons between variables.

Review

Variable



```
set variable ▾ to 0
```

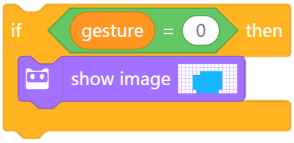
With the Operators blocks, you can use variables to store the calculation results.

Coding Practice

Using Codey Rocky, you can play the game rock-paper-scissors with friends.

(1) When Codey Rocky starts up, the initial values are all set to be 0.

(2) Codey Rocky changes its hand gestures randomly. The numbers 0, 1 and 2 are used to represent Rock, Scissors and Paper respectively. When Codey is shaken, the variable "gesture" will be one of the three numbers randomly. If the value is 0, the LED screen will show an image of a fist. If the value is 1, the LED screen will show an image of scissors. If the value is 2, the LED screen will show an image of paper.



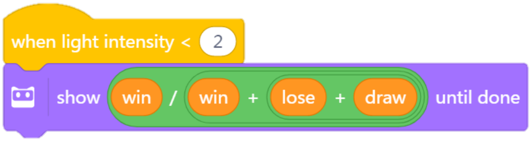
```
if gesture = 0 then
    show image
```

(3) If Codey wins the game, you need to press button A and the variable "win" will change its value by 1. Meanwhile, Codey shows a smiley face and plays the sound "laugh".

(4) If Codey loses the game, you need to press button B and the variable "lose" will change its value by 1. Meanwhile, Codey shows a sad face and plays the sound "sad".

(5) If it is a draw, you need to press button C and the variable "draw" will change its value by 1. Meanwhile, Codey looks calm and gives you a hum sound.

(6) In some cases, Codey might steal a glance at its chance of winning. When the light intensity is lower than 2, the chance of winning will appear on the LED screen in the form of decimals.



```
when light intensity < 2
show win / (win + lose + draw) until done
```

Discussion：

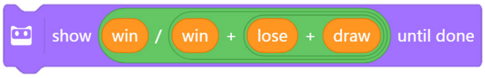
(1) How to display the chance of winning on the LED screen in the form of percentage?

(2) How to display the chance of winning in the form of ratio?

Notes

(1) When coding Codey Rocky, you are supposed to use no more than six Events blocks;

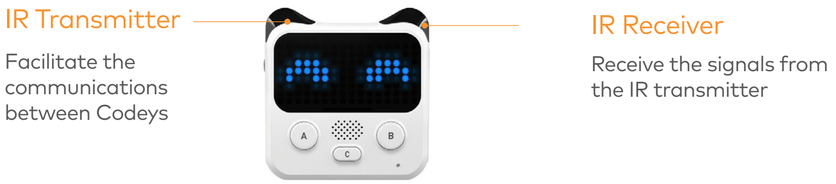
(2) To get the chance of winning, you'll need to add multiple Operators blocks to the code.
Formula:
Winning times / (winning times+ losing times + draw times) = Chance of winning



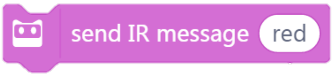
```
show (win / (win + lose + draw)) until done
```

When button A is pressed, Codey transmits an infrared signal; if another Codey receives the signal, its health points go down by 1; the game ends when the health points fall to 0.

(1) The ears of the Codey are preset with IR receivers and IR transmitters, which facilitates the wireless communication between two Codeys.

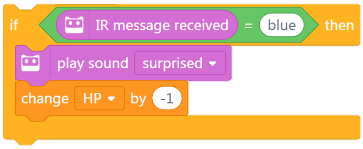

IR Transmitter
Facilitate the communications between Codeys
IR Receiver
Receive the signals from the IR transmitter
A
B
C

(2) Select the block **send IR message** in the category "Infrared". Have two Codeys send each other different messages. In this way, they can identify each other. For instance, one Codey send the message "red" and the other Codey send the message "blue". In this case, the messages serve as the bullets.



```
send IR message (red)
```

(3) Once Codey A receives the message from Codey B, Codey A gets shot and its health points reduce by 1. You can use the if statement block to assess whether Codey receives an IR message (gets shot). Once Codey gets shot, it will make a surprised sound and the health points will fall by 1.



```
if <(IR message received) = (blue)> then
    play sound (surprised ▾)
    change (HP ▾) by (-1)
```

(4) When Codey starts up, the initial value of the health point is 10. The next step is to use the forever block to keep detecting whether Codey gets "bullets" from its "enemy" . If Codey gets shot, the health point change by "-1" and plays the sound "surprised". The game will be over when the health points fall to 0 and Codey will look sad.

(5) If Codey still has health points left (the value is greater than 0) when the button A is pressed, it will then send an IR message and play the sound "laser".

Extension activities:

If it is a battle of three, then how will you rewrite the programs?

Note

When coding Codey Rocky, you are supposed to use no more than six Events blocks.

Discussion：

(1) If there are 3 teams, then how will you rewrite the program?

(2) How to use the repeat until block to create programs?

Share

In this lesson, we created a project ____

During the process, we came into an issue______________________________________

In the end, we solved the problem by ____

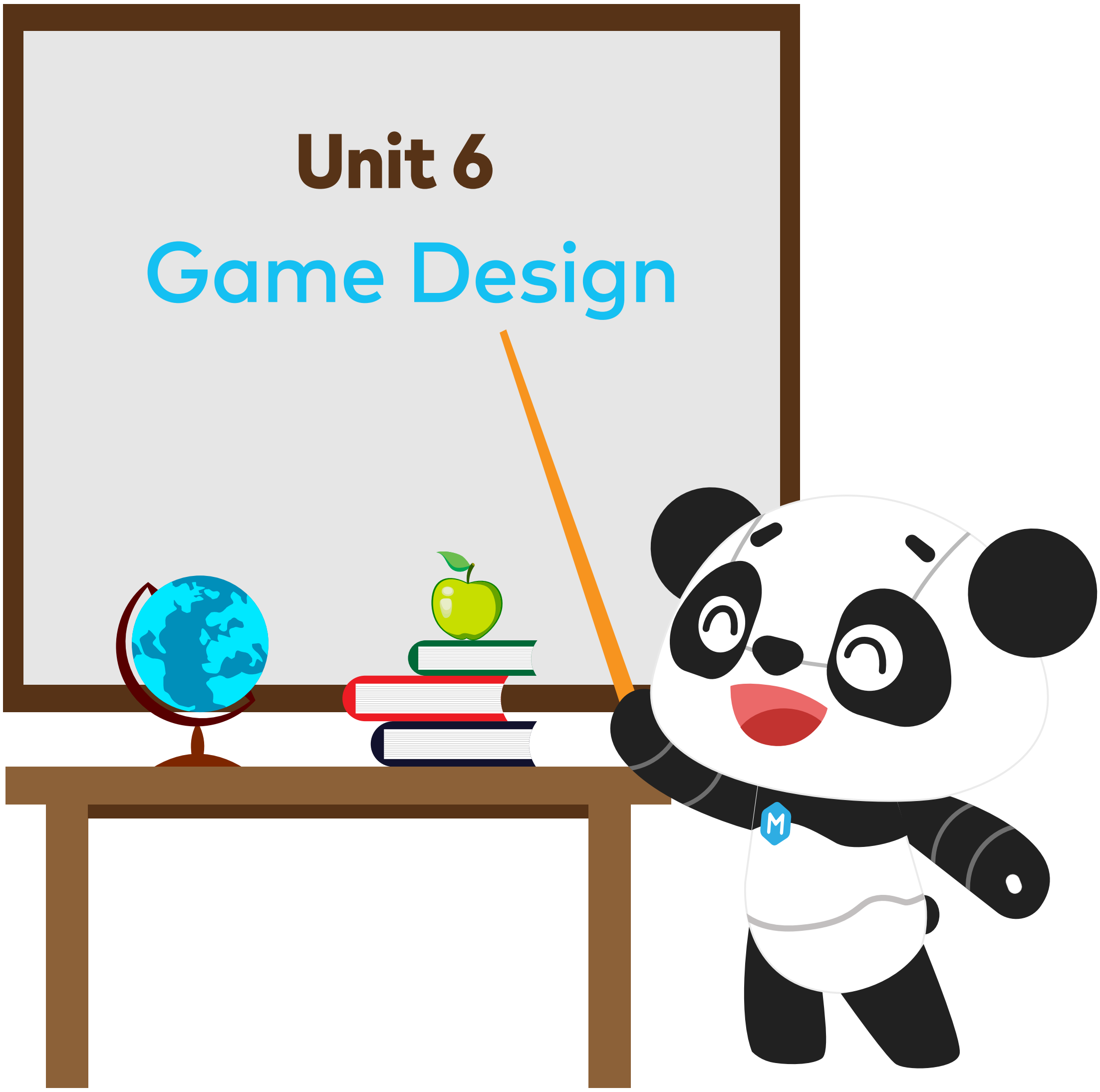
Self-review

1. In this lesson, I learned that_________________________

2. The part I like most about this lesson is_________________

Unit 6

Game Design



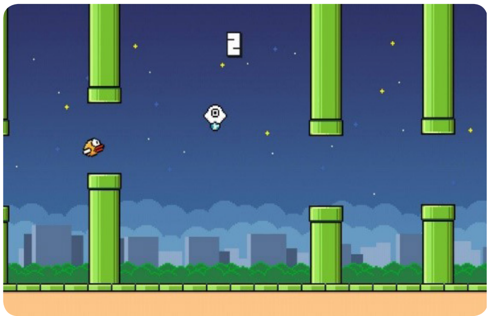
My Speedway

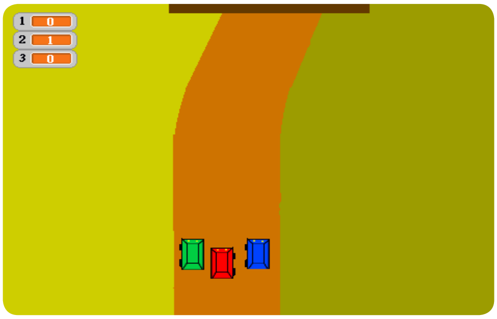
Learning Objectives

1. Know how to use mBlock to design game scenes and characters.
2. Master the ways to achieve dynamic effects.
3. Know how to create simple games.

Lead-in

How players play a game is mostly decided by its characters, scenes and back stories. And these elements work together to determine which group of potential players will find the game attractive. In this sense, we have no reason to ignore the importance of characters and scenes designs when we're talking about a game. When characters and scenes in a game have more intricate details, the game is more visually enjoyable and naturally will keep players engaged.

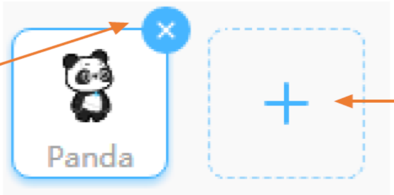




Learn through play

Assignment: Racing Car
About the game: The racing game is coming. We need to design the speedway. Now let's get started.

(1) Delete the sprite "Panda":

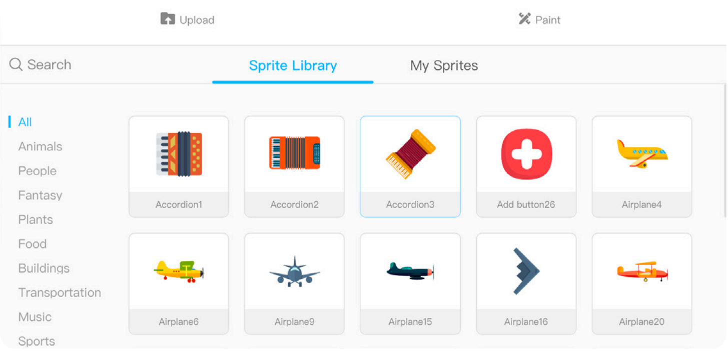


Click here to delete

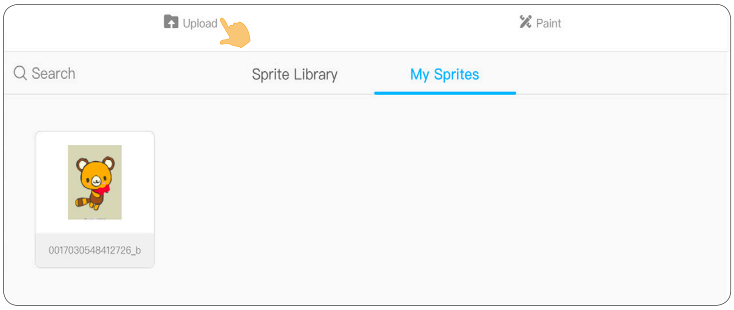
Click here to add a sprite

(2) Add the sprite "racing car". The way to add a sprite:

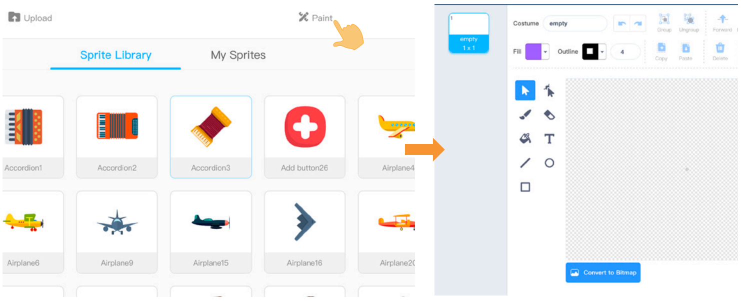
① Pick a sprite directly from the library:



② Search for a picture on the Internet and import the picture:

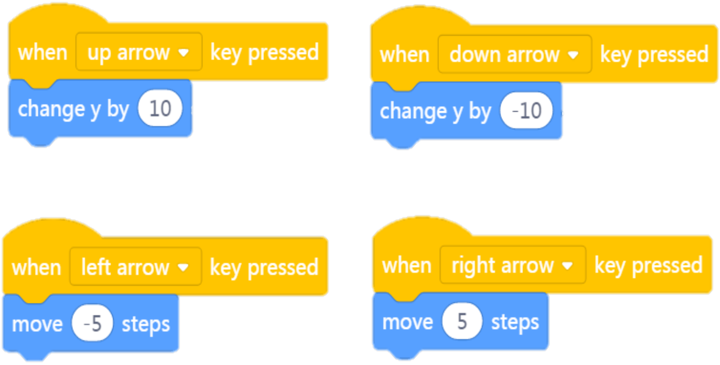


③ Draw a sprite yourself:



(3) Use the arrow keys to move the racing car in a specific direction:

① Press the "Up arrow" to move the car upwards;
②Press the "Down arrow" to move the car downwards;
③Press the "Left arrow" to move the car to the left;
④Press the "Right arrow" to move the car to the right.



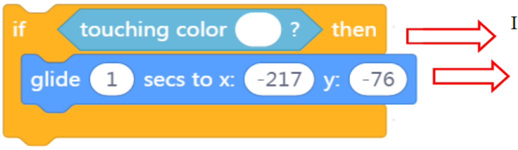
```
when up arrow ▾ key pressed
change y by 10

when down arrow ▾ key pressed
change y by -10

when left arrow ▾ key pressed
move -5 steps

when right arrow ▾ key pressed
move 5 steps
```

(4) Add the background "speedway":

① Pick the background directly from the library;
②Search for a picture on the Internet and import it;
③Draw the background yourself.

(5) Game rules:

①Make sure the car drives inside the race track. Once the racing car runs outside the race track, the car returns to the starting place. Figure out how to achieve this effect;

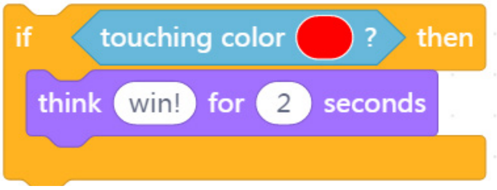


```
if touching color ( ) ? then
    glide 1 secs to x: -217 y: -76
```

If the car detects the color white,

then the car will go back to the starting place.You can relocate the starting point.

②When the car arrives at the red finish line, the car shows the word "win!".



```
if touching color ● ? then
  think win! for 2 seconds
```

Challenge: Try to complete the game design——My Speedway

Extension activities: Add some backward-moving trees in the race track scene. (the car is running forward while the trees are moving backward)

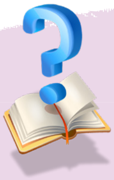
Share

In this lesson, we created a project ____ .

During the process, we came into an issue___________________________________ .

In the end, we solved the problem by ______________________________________ .

Self-review



1.In this lesson, I learned that______________________

2.The part I like most about this lesson is___________________

3.The best design of today is (OR you can draw it out in the blank area below)___________________

Game Control Schemes

Learning Objectives

1. Know how to design an interactive scheme for a game.

Lead-in

There are different types of games platforms and game control schemes.
Common games platforms: PC, Mobile, Consoles, Television, VR
Control schemes: Mouse, Keyboard, Controller, VR headset and more

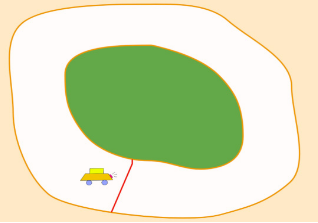
	A controller	A mouse + a keyboard
Good stuff	Holding a controller in hands feels good; Easy to operate; Provide a good immersive experience; No gesture constraints.	Precise; Speedy; Suitable for PC games.
Bad stuff	Poor accuracy; Extra expenditure	The gaming experience might be poor; Need a PC to work with
game genres	Action, Fighting, Driving	Shooter, Real-time strategy

Learn through play

Topic：Racing game

Game rules: The racing game is about to start. Now we are going to set up the car before the race starts. I hope that the car to run in the same direction as I move. When I lean towards the left, the car moves to the left; when I lean towards the right, the car moves to the right.

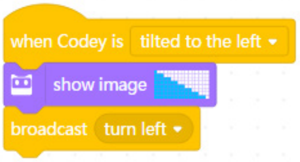
(1) Design the racing car and the speedway:



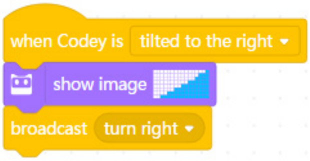
(2) Turn Codey into a game controller:

·Codey has a built-in gyroscope. It is used to tell whether Codey is tilted (front, back, left, right);

·If Codey is tilted, then a message will be broadcast and the sprite on the stage (racing car) will receive the message.



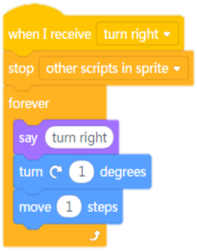
```
when Codey is tilted to the left ▾
show image
broadcast turn left ▾
```



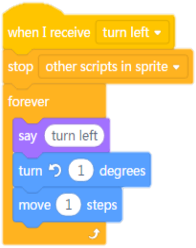
```
when Codey is tilted to the right ▾
show image
broadcast turn right ▾
```

(3) When Codey tilts to the left, the car will turn to the left; when Codey tilts to the right, the car will turn to the right.

(4) If you program the car to change from turning left to turning right, you need to make the car stop moving first.



```
when I receive turn right ▾
stop other scripts in sprite ▾
forever
    say turn right
    turn ↻ 1 degrees
    move 1 steps
```



```
when I receive turn left ▾
stop other scripts in sprite ▾
forever
    say turn left
    turn ↺ 1 degrees
    move 1 steps
```

Thinking:What if we don't stop other scripts in the sprite?

Challenge:Pick an idea and create a game based on the idea. Test the game and refine it.

Extension activities:

1) Using “variable” in the program to replace the “broadcast”;

2) Making the race track more exquisite;

3) Diversifying the characters.

Share

In this lesson, we created a project ____ .

During the process, we came into an issue___________________________________ .

In the end, we solved the problem by _______________________________________ .

Self-review

1.In this lesson, I learned that___________________

2.The part I like most about this lesson is_______________

3. The best game of today is (OR you can draw it out in the blank area below)______________

Game Mechanics I

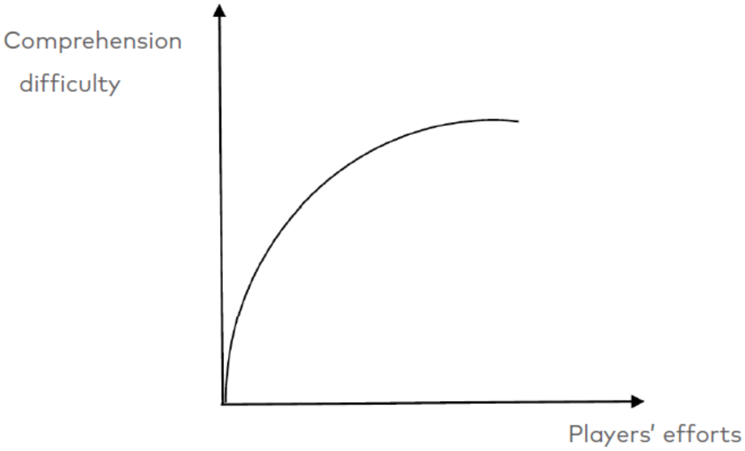
Learning Objectives

1. Know how to design game rules using mBlock 5.
2. Know how to design game mechanics for a game:Design rules for the racing games.
3. Know how to spice up their prototypes by adding game mechanics.

Lead-in

A good game should include well-designed game rules which clearly define what players are going to do in the game. And, the game rules here are called game mechanics.

Through the game mechanics, players understand what they should do and shouldn't do in the game. Game mechanics are an indispensable part of a game because they influence how a game is played and proceeds. Moreover, game mechanics contribute to the gaming experiences of players.



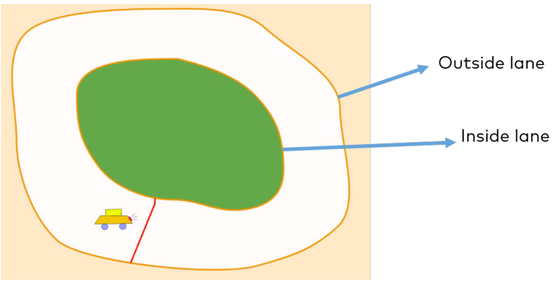
Learn through play

Topic: Game mechanics design——Beginner

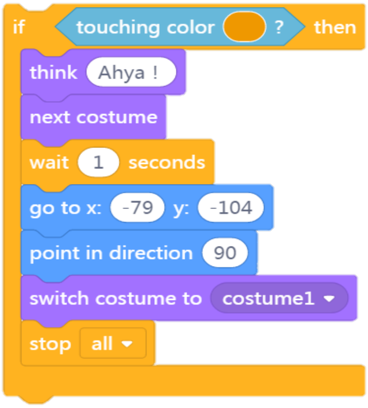
To make our game complete, we need to design rules for the game.

Game rules:The racing car has to run inside the track. Once the car runs outside the track, it has to return to the starting place and start all over again.

(1) About the race tracks: The race tracks consist of two lanes – outside lane and inside lane. The racing car is expected to drive between the outside lane and the inside lane.



(2) Detecting whether the car runs outside the race track or not: it explodes (a new costume) and reappears at the starting place as a racing car, ready to start off.



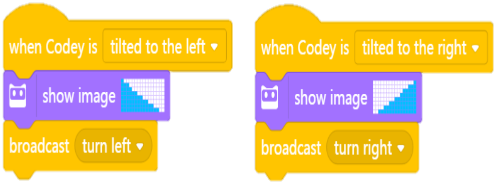
```
if touching color ● ? then
    think Ahya !
    next costume
    wait 1 seconds
    go to x: -79 y: -104
    point in direction 90
    switch costume to costume1 ▾
    stop all ▾
```

(3) Turn Codey into a game controller:

Challenge:Try to complete your game design——My speedway!

①Codey has a built-in gyroscope. It is used to tell whether Codey is tilted (front, back, left, right);

② If Codey is tilted, then a message will be broadcast and the sprite (racing car) on the stage will receive the message.



```
when Codey is [tilted to the left ▾]
show image [ ]
broadcast [turn left ▾]
```

```
when Codey is [tilted to the right ▾]
show image [ ]
broadcast [turn right ▾]
```

(4) Add the piece of program you just created to the previous code.

Challenge:Try to add simple rules to your game and invite players to test the game. Refine the game based on the feedback.

Extension activities: Increase the difficulty of rules:The car has to finish one lap within the required time.

Share

In this lesson, we created a project ____ .

During the process, we came into an issue____________________________________ .

In the end, we solved the problem by _______________________________________ .

Self-review

1.In this lesson, I learned that_____________________

2.The part I like most about this lesson is_________________

3.In our daily life, rules are important as well. Can you think of any rules? (OR you can draw them out in the blank area below)

____

Game Mechanics II

Learning Objectives

1. Know how to design an interactive scheme for a game.
2.Know how to spice up their racing games by increasing the difficulties.

Lead-in

Game Conflicts, By designing challenges for games, we try to make it hard for players to accomplish the goals of games directly. There are three types of game conflicts:
Obstacles: Design obstacles in games. The types of obstacles could be categorized into fixed obstacles and moving obstacles.

Opponents: Add opponents in games (obstacles or other players in multiplayer games).
Dilemmas: Players might come across multiple solutions. But each solution comes with pros and cons.

Learn through play

Topic: Game mechanics design——Game Conflicts

Game rules: To make the racing game more thrilling and engage more players, we need to make the game a little bit more complicated.

We are going to add game conflicts by putting moving obstacles in the game. The obstacles keep coming out of somewhere. The racing car has to avoid those obstacles all the way when driving forwards. Once touching obstacles, the car has to go back to the place where it sets out and restarts the game.

(1) Add a sprite "Bat" :

①Bat 1: A complete bat;

②Bat 2: A bat smashed to pieces by the racing car. Use black spots to represent the smashed bat;

(2) Game conflict: The racing car has to avoid the bats falling from the sky when driving.

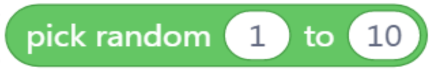
(3) Outcomes:

①The game will be over once the bat hits the car;

②The game continues if the car doesn't hit the bat. But the bat will appear again later.

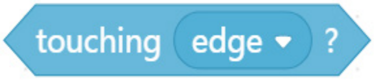
(4) How to achieve this:

①The bat appears randomly:



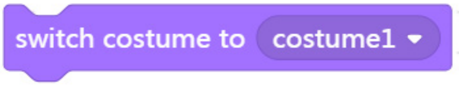
```
pick random 1 to 10
```

②After landing, the bat soon falls from somewhere in the sky:



```
touching edge ▾ ?
```

③Once the car gets hit by the bat, it explodes (switching to a new costume):



```
switch costume to (costume1 ▾)
```

④The car should go back to the starting place once it gets hit by the bat.
Challenge:Try to complete the game conflict design. Invite players to test the game and refine it.

Extension activities:

1） Add a new moving obstacle (gold coin). The racing car should capture the gold coin before the coin falls on the ground and tries to avoid the bats at the same time.
2） Add a new track shortcut to complicate the dilemma-based choices. But make this track narrower.

Share

In this lesson, we created a project ____ .

During the process, we came into an issue_______________________________________ .

In the end, we solved the problem by ____ .

Self-review

1.In this lesson, I learned that_________________________

2.The part I like most about this lesson is___________________

3.The best game of today is (OR you can draw it out in the black area below)_________________

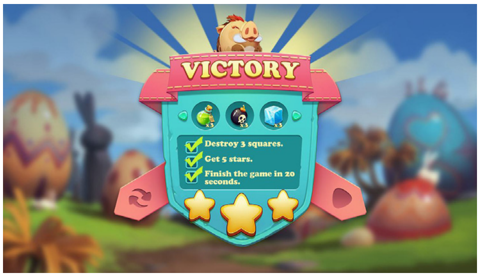
Fast and Furious

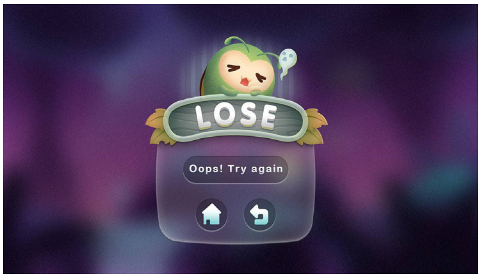
Learning Objectives

1. Develop the skills to design game mechanism and design the outcomes in the racing game.
2. Know how to improve the gaming experience of the games designed by themselves.

Lead-in

Game Outcomes——a game outcome is a winning situation or a losing situation that results from a combination of player's strategies. Players make choices and show different behaviors in games. These choices and behaviors will lead to either a winning outcome or a losing outcome.





Learn through play

Topic: Fast and Furious

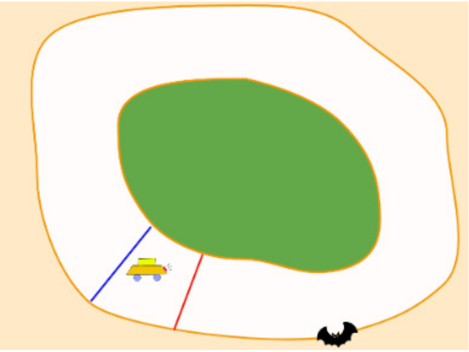
Game rules:To make the game complete, we need to add game outcomes: When the car accomplishes one circle (successfully gets to the blue finishing line), the car will show a sign and make a sound, telling you that you win the game!

(1) Outcomes:

Win: The racing car can accomplish one running lane.

Lose: The car has to go back to the starting point when it runs outside the track or hits obstacles.

(2) Detecting the finish line: Draw a finish line. With this finish line, you can easily program to detect whether the car finishes one lap.



(3) Turn Codey into a game controller:

(4) Add game rules:

①The car explodes once it drives outside the speedway.

②The car avoids bats and make sounds when it gets hit by a bat.

Challenge:Try to complete your game by defining the game rules and outcomes. Invite players to test the game and refine it.

Extension activities:

Complicated rules:

1) the car should accomplish one running lane within a set period of time;

2) should accomplish several lanes;

3)should gets to the red line.

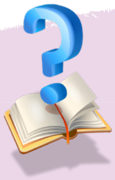
Share

In this lesson, we created a project ____ .

During the process, we came into an issue____ .

In the end, we solved the problem by ____ .

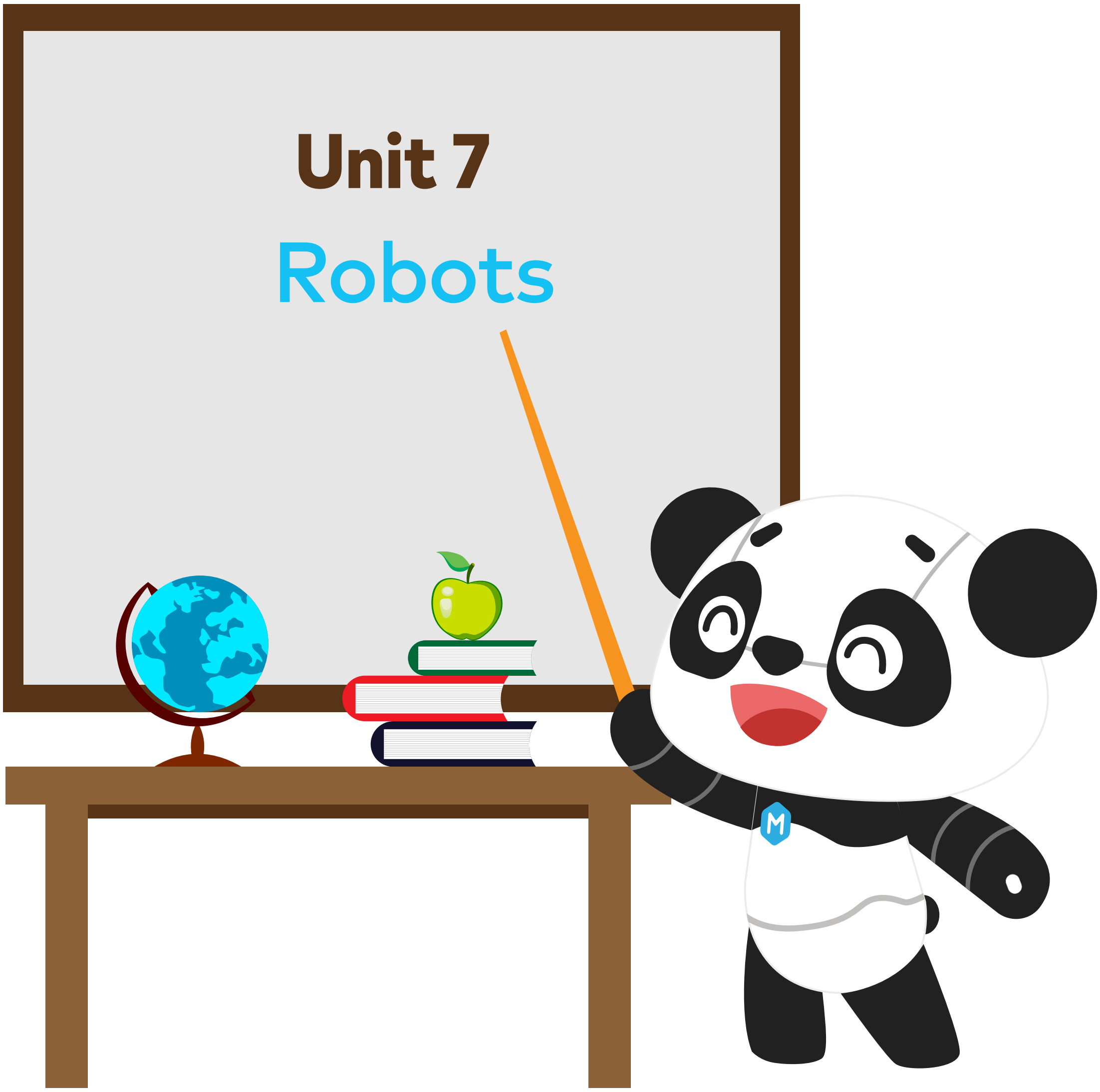
Self-review



1.In this lesson, I learned that______________________

2.The part I like most about this lesson is__________________

3.The best game of today is (OR you can draw it out in the blank area below)________________



Unit 7

Robots

Make a Turn

Learning Objectives

1. Review the basics of robots.
2. Design a route for Codey Rocky to follow.

Lead-in

Robots refer to machines that can perform tasks as instructed. The instructions could be given by human beings or could be preset programs. The most common robots are educational robots, delivery robots and security robots.

educational robots

food-delivery robots

security robots

robots in logistics

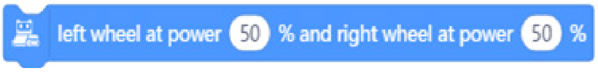
Learn through play

Game rules:
Have two students (Student A & B) act as two wheels. Tell them to simulate the process of how two wheels move
Tell students to complete the following tasks:

(1) Move forward together
(2) Make a circle
(3) Make a semicircle turn

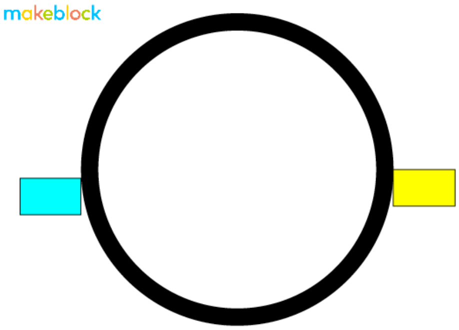
Note: Have two students work in pairs to complete each of the tasks.

(1) The block tells the robot to make both of the wheels move forward at a certain level of power. The power level ranges from 0 to 100.

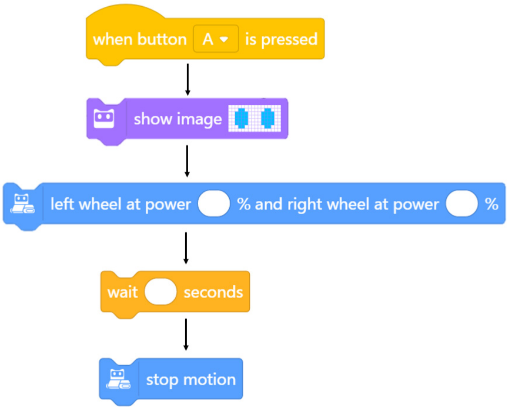


left wheel at power 50 % and right wheel at power 50 %

Task 1: Program Codey Rocky to complete the circle route and the semicircle route.

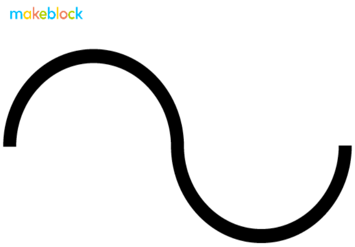


Thinking

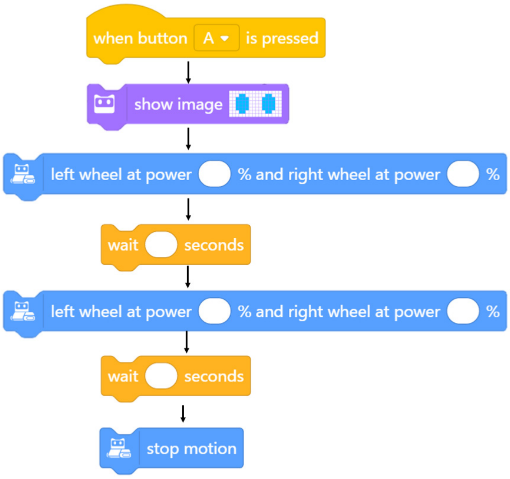


```
when button A ▾ is pressed
show image
left wheel at power ( ) % and right wheel at power ( ) %
wait ( ) seconds
stop motion
```

task2: Show students an S curve route and set a time limit. Students need to make Codey Rocky finish following the route within the given time.



Thinking



```
when button A ▾ is pressed
show image
left wheel at power ( ) % and right wheel at power ( ) %
wait ( ) seconds
left wheel at power ( ) % and right wheel at power ( ) %
wait ( ) seconds
stop motion
```

Challenge: Design curved tracks (consecutive curves or a figure-8 pattern route) Exchange your map with your classmates' maps and get inspirations from others' works.

Share

In this lesson, we created a project ____ .

During the process, we came into an issue__________________________________ .

In the end, we solved the problem by ______________________________________ .

Self-review

1.In this lesson, I learned that______________________

2.The part I like most about this lesson is___________________

3.The most unique route of today is（OR you can draw it out in the blank area below） ___________________

Avoid Obstacles

Learning Objectives

1. Know the application of self-driving car technology.
2. Know how to create a function and call the function.
3. Able to complete the challenge by applying functions and mathematical knowledge.

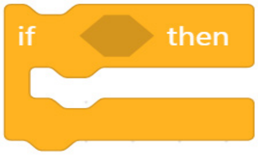
Lead-in

Self-driving cars can decide themselves at what speed they are supposed to drive based on the information gathered by sensors. Moreover, sensors inform self-driving cars of the traffic conditions in real time and help cars identify the best route.

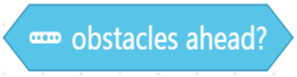
Learn through play

Build a road full of obstacles and have students simulate self-driving cars to avoid obstacles.

(1) Task 1: Turn Codey Rocky into an alarm:

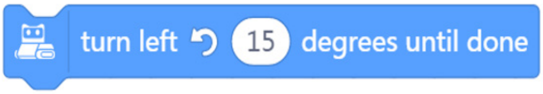


```
if ⬡ then
```

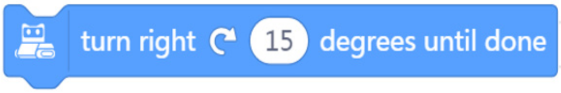


```
obstacles ahead?
```

(2) Task 2: The teacher will give you a map. Use the map as reference and program Codey Rocky to perform tasks as required:



```
turn left ↺ 15 degrees until done
```



```
turn right ↻ 15 degrees until done
```

Challenge: Have each group design a program to ensure Codey Rocky can walk safely on the table without falling.

Thinking: How to detect edges of the desk?

Share

In this lesson, we created a project ____ .

During the process, we came into an issue___________________________________ .

In the end, we solved the problem by _______________________________________ .

Self-review

1.In this lesson, I learned that_________________
2.The part I like most about this lesson is______________
3.Can you think of any real-life situations where we need to avoid obstacles? (OR you can draw it out in the black area below) ______________

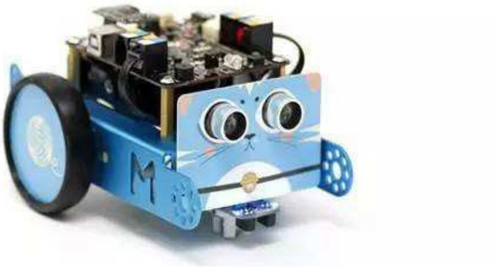
Line-Following Car

Learning Objectives

1. Know the applications of line-following robots in our daily life.
2. Learn about the reflected light intensity.
3. Understand why Codey Rocky can follow lines, complete tasks.

Lead-in

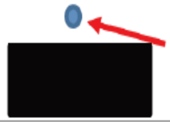
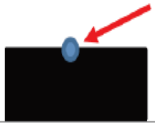
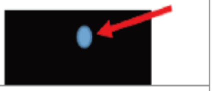
One of the most basic skills of robots is following lines. When we are walking, we often keep our eyes on the roads in case we might tumble. The case is the same with robots. The only difference is that robots "see" things with "sensors". And line-following robots are widely applied in real life, starting from domestic uses to industrial uses.



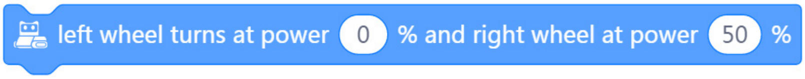
Learn through play

1. Program Rocky to measure the light intensity and display the value on its screen in real time.
2. Measure Light Intensity: Write down each card's light intensity value.

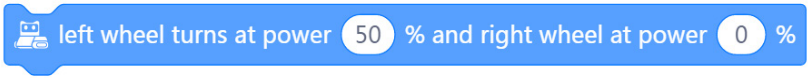
(1) Task1: Measure Black Line's Light Intensity——Measure the reflected light intensity when the RGB LED light spot is on the edge of the black line.

	In white area	On the edge	In black area
Position of light spot			
Reflected light intensity			
Motion			

(2) Task2: Simulation Game rules:
When the fist is in the black area, take one step forward on your right foot, turn your body 45 degrees to the left and keep the left foot still; when the fist is in the gray area, take one step forward with your left foot, turn your body 45 degrees to the right and keep the right foot still.

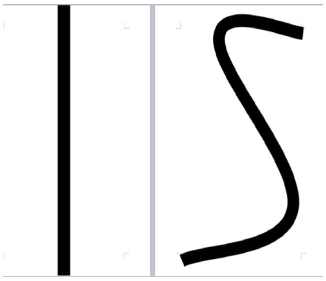


left wheel turns at power (0) % and right wheel at power (50) %



left wheel turns at power (50) % and right wheel at power (0) %

Challenge: Figure out how to make Codey Rocky follow these paths.



Share

In this lesson, we created a project ____ .

During the process, we came into an issue____ .

In the end, we solved the problem by ____ .

Self-review

1.In this lesson, I learned that__________________

2.The part I like most about this lesson is____________________

3.Can you think of any real-life situation where we need to avoid obstacles? (OR you can draw it out in the blank area below)

____