



SKH LAM WOO MEMORIAL SECONDARY SCHOOL

S2 COMPUTER LITERACY

THUNKABLE AND TINKERCAD

LET'S CODE!

Thunkable (1): Introduction And Basic Interface Design

Thunkable is a platform where anyone can build their own mobile apps. It supports both Android and iOS, so it is known as “cross-platform” (跨平台).



thunkable



Operating Systems

- **Android** and **iOS** are the most popular operating systems (OS) in mobile phone.
- Operating system is a software which provides an interface between users and computer hardware (such as processor [CPU] and memory).
- Name some examples of operating systems that are commonly found in desktop / laptop computer:

○ **First mobile app**

1. Go to <http://thunkable.com/>, and click “GET STARTED”.

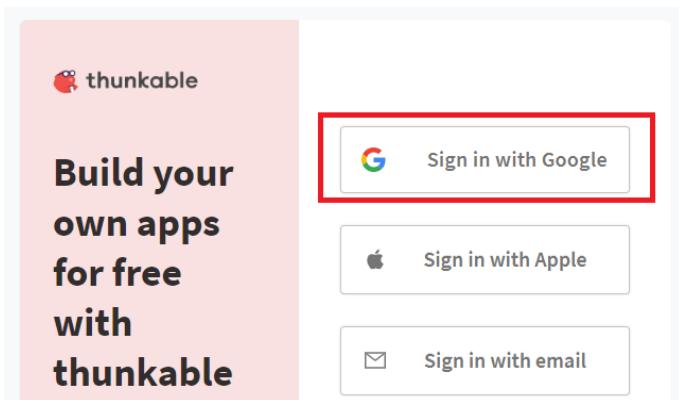
Community Docs Blog Pricing Log In Sign Up

Fast apps, no coding

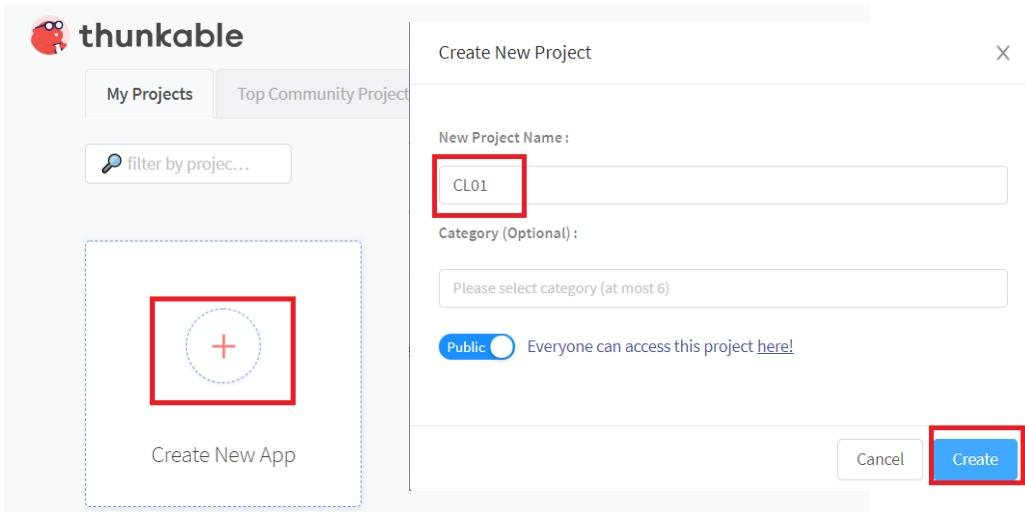
One project, three cross-platform apps. No Coding.

GET STARTED

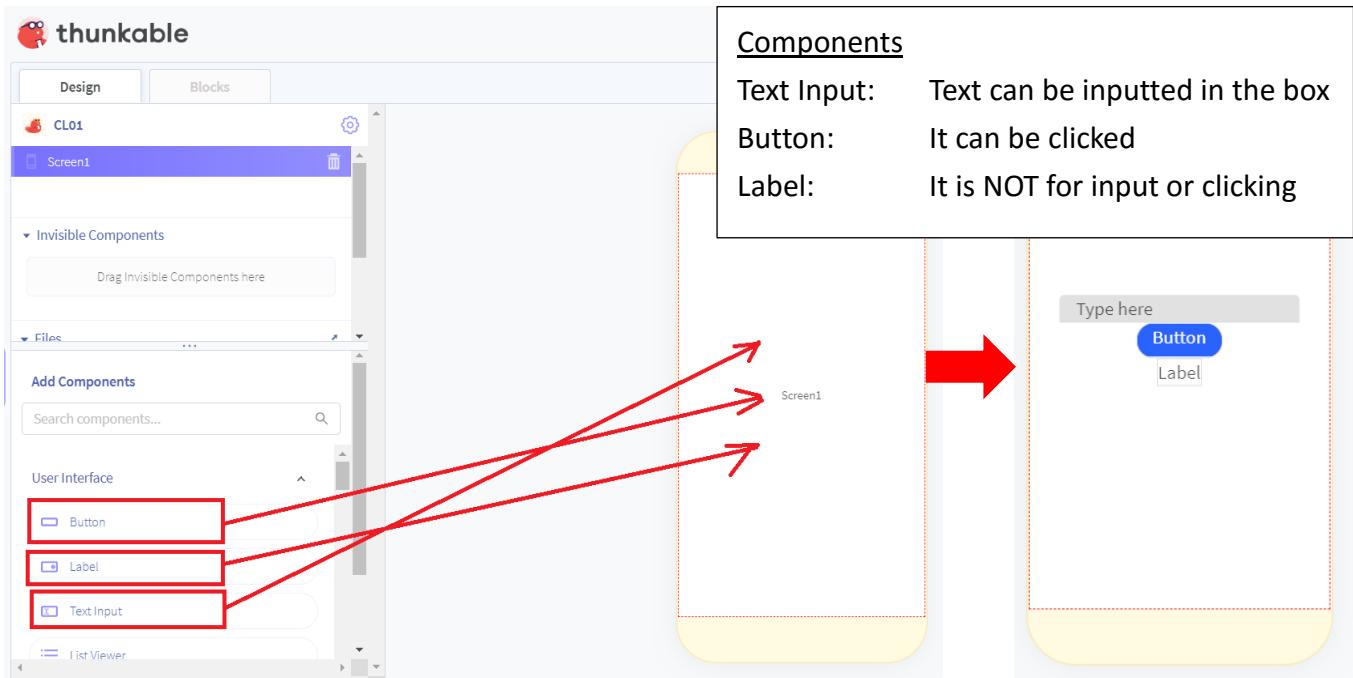
- Sign in with Google account.



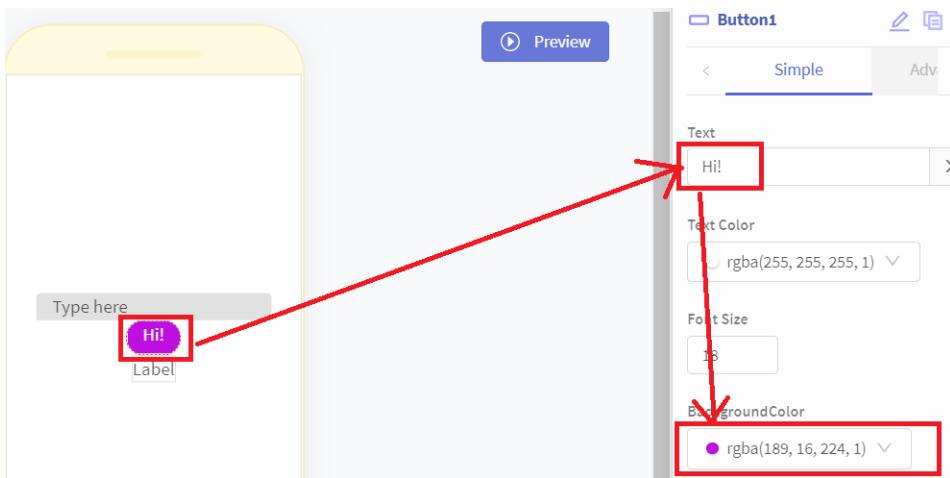
- Click “+ Create New App”. Then input the project name as “CL01”, and click “Create”.



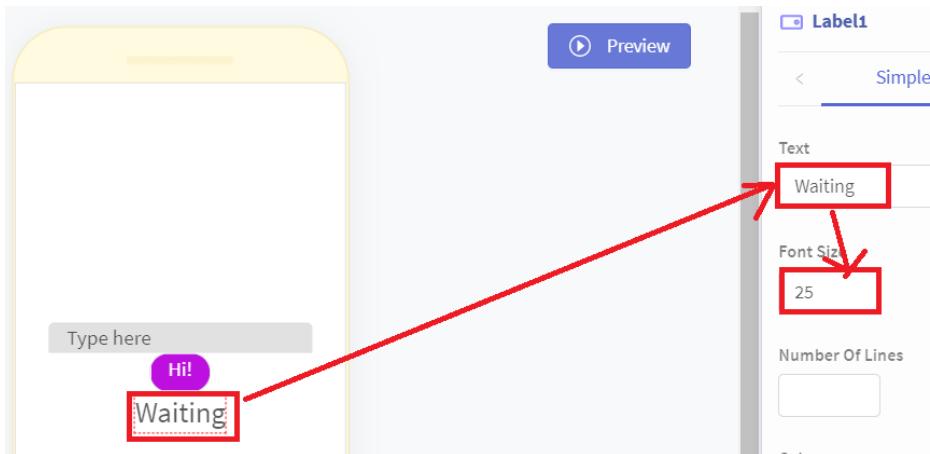
- Put “Text Input”, “Button” and “Label” into “Screen 1”.



- Select the “Button”. Change its text to “Hi!” and BackgroundColor into purple.



6. Select the “Label”. Change its text to “Waiting” and Font Size to “25”.



7. Change the name of each component by clicking the icon shown below.

Original Name	New Name
Text_Input1	Your Name
Button1	Hi Button
Label1	Display Result

A red arrow points from the 'Edit Name' icon in the top right of the Thunkable interface to the 'New Name' column of the table.

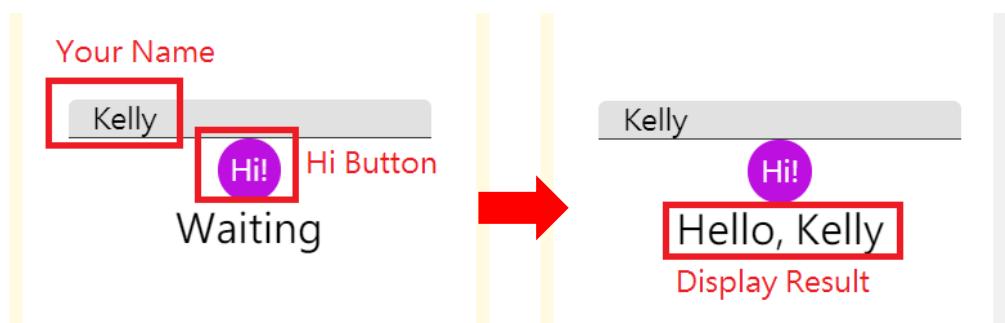
8. The interface (界面) is done. We will move to “Blocks” to do programming for actual functions.



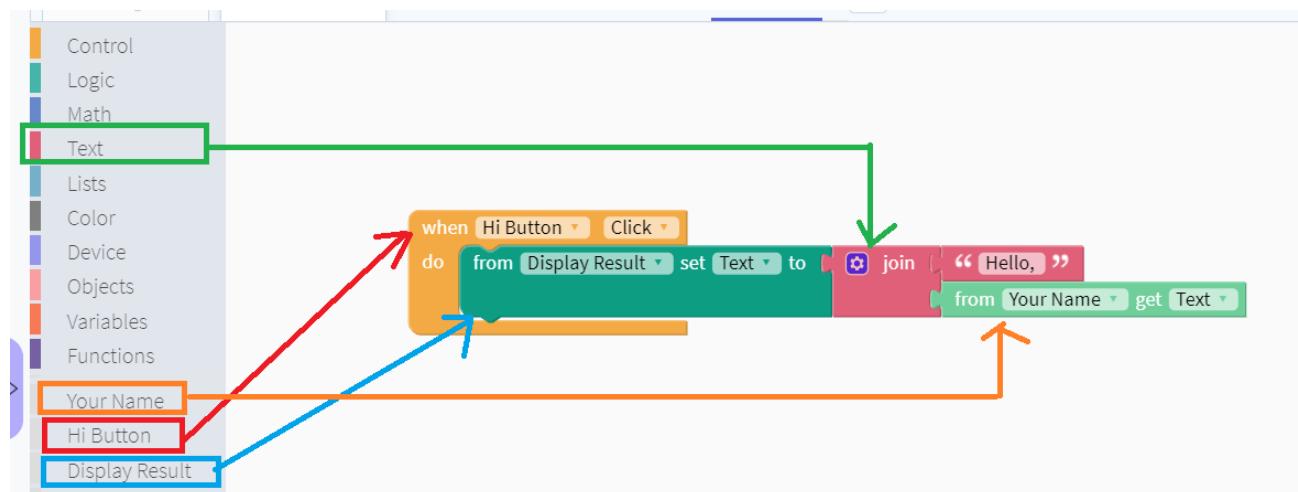
9. Before writing the app, you should understand what the function of your app is.

Click “Hi Button”

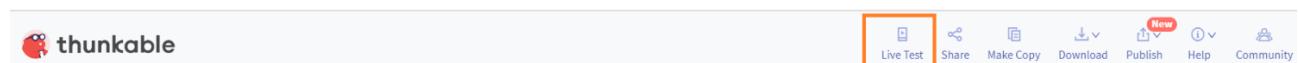
→ The text in “Display Result” would become “Hello, [text in “Your Name”]” (e.g. Hello, Kelly)



10. Drag the 4 blocks as shown below:



11. You can test your app by clicking “Live Test”.

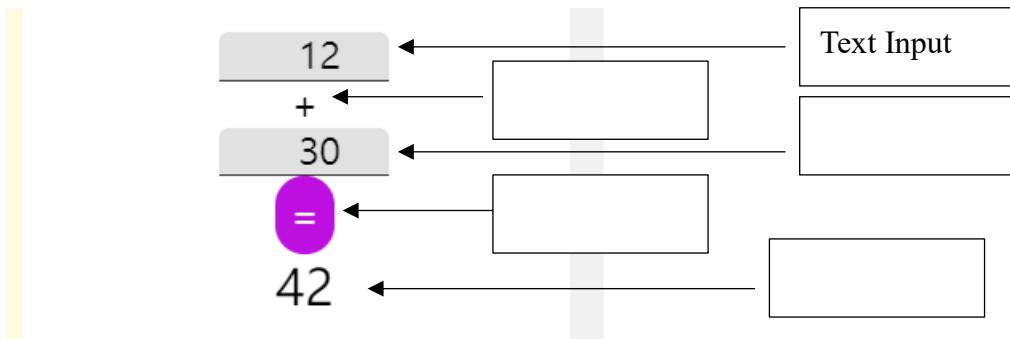


12. Your first app is then completed! 😊



○ Simple Calculator (1)

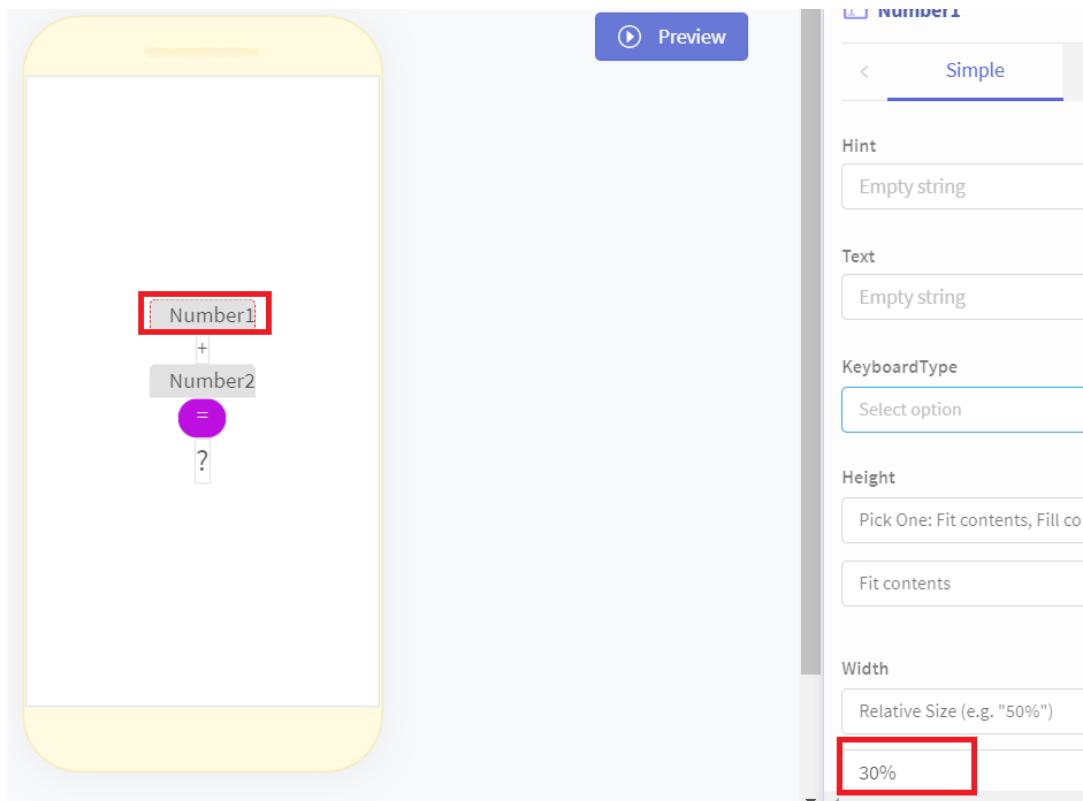
In the exercise, you have to create a simple calculator for adding 2 numbers.



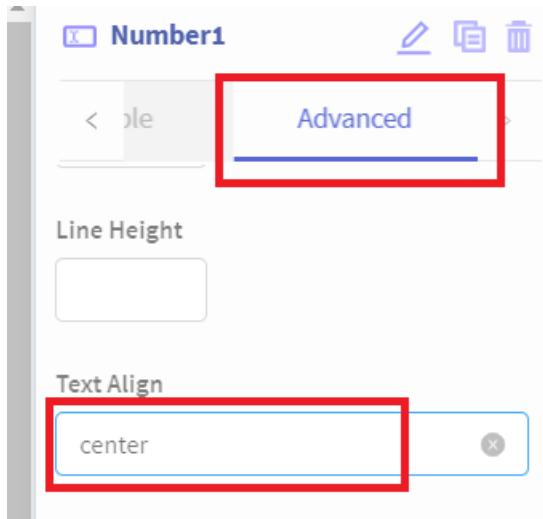
1. Open a new project called “CL02”.
2. Fill in the boxes above to indicate the type of component. Drag the components accordingly.
3. Change the name and text of each components.

Name	Text
Number1	
Plus	+
Number2	
= Button	=
Display Result	?

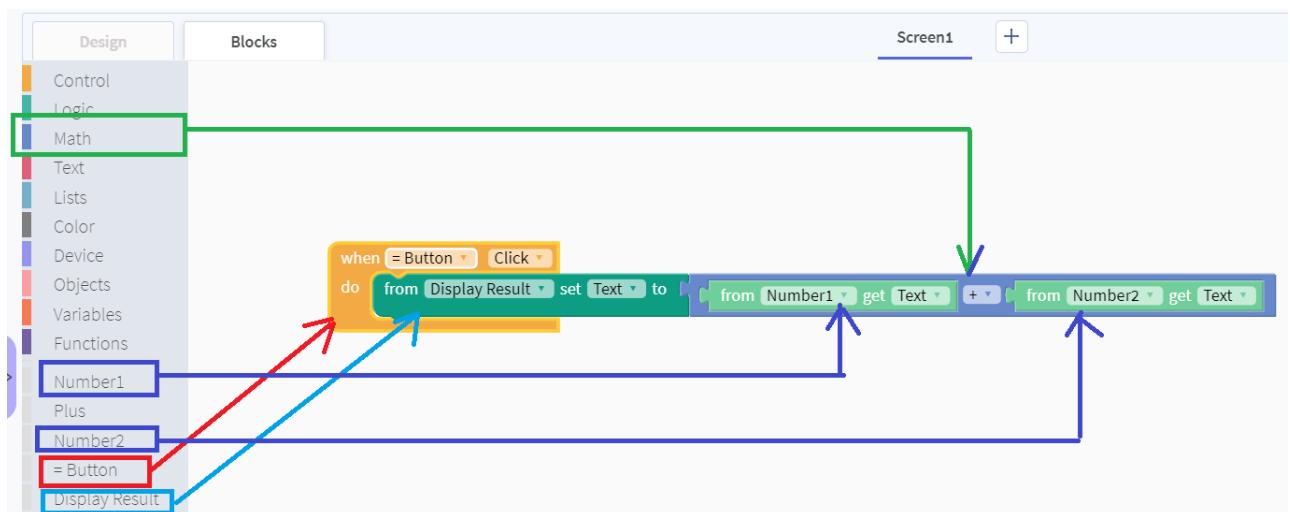
4. Select “Number1”, and change its Width to “30%”.



5. Select “Advanced”, and change Text Align to “center”.



6. Move to “Blocks”, and drag the following five blocks accordingly:

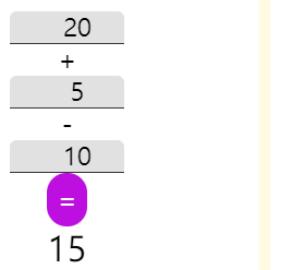


7. Test your app.

- **Simple Calculator (2)**

Modify Simple Calculator (1) to do the following app.

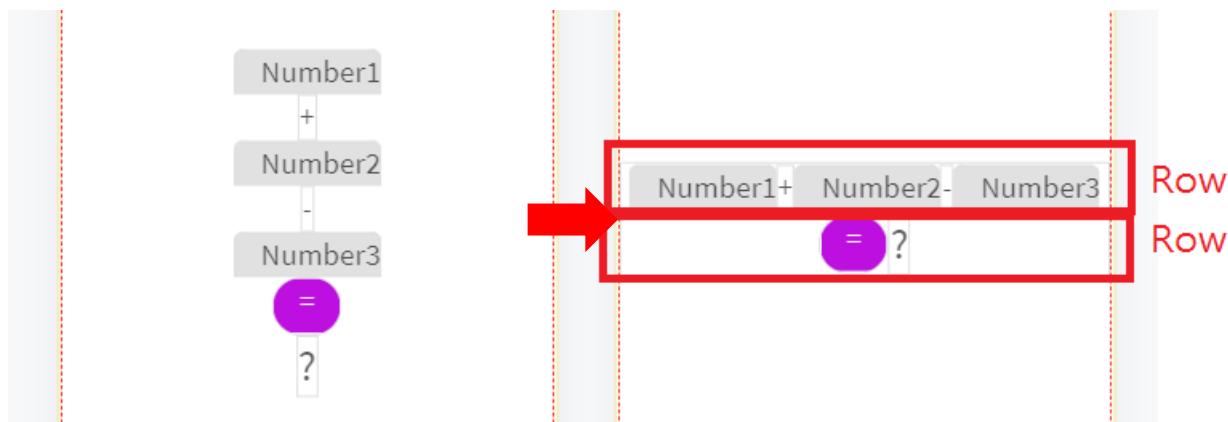
Save as “CL02”.



2

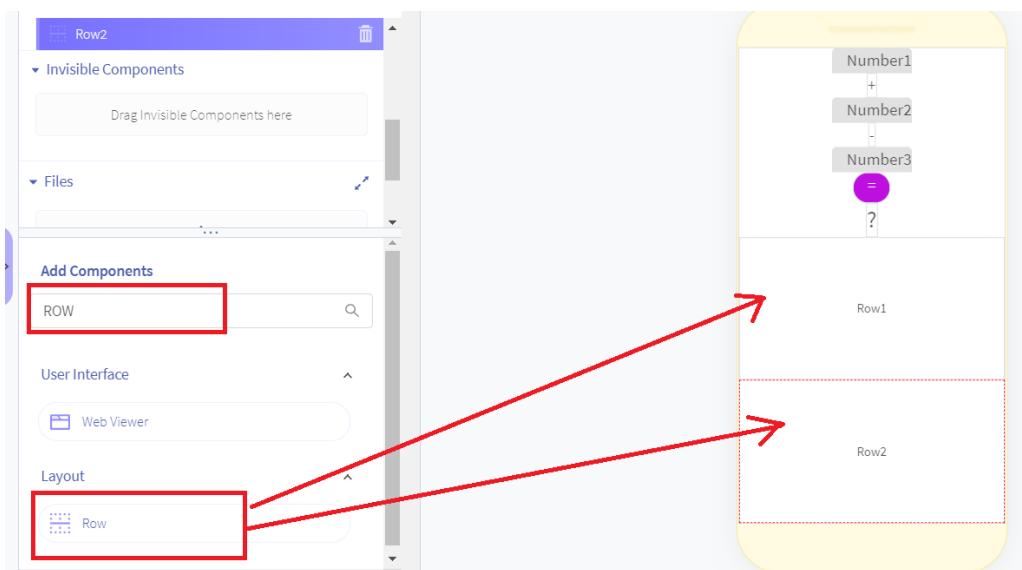
Thunkable (2): Conditional Structure

In the previous lesson, we have learnt the basic components of interface, which are “text input”, “button” and “label”. However, you can only arrange them vertically. Therefore, you will learn how to arrange them in horizontal manner by using another component called “Row”.



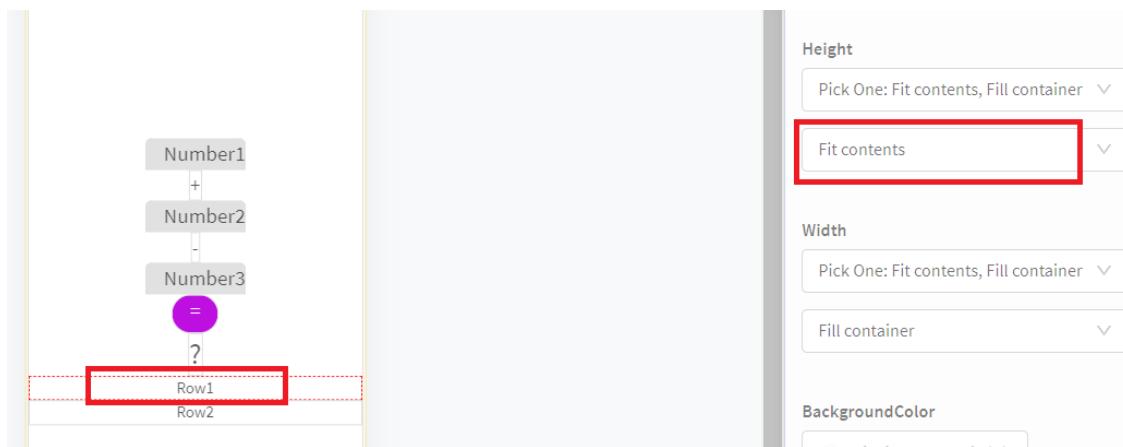
○ Simple Calculator (3)

1. Open “CL02”.
2. Search “Row” in components, and add 2 “Row” into the screen.



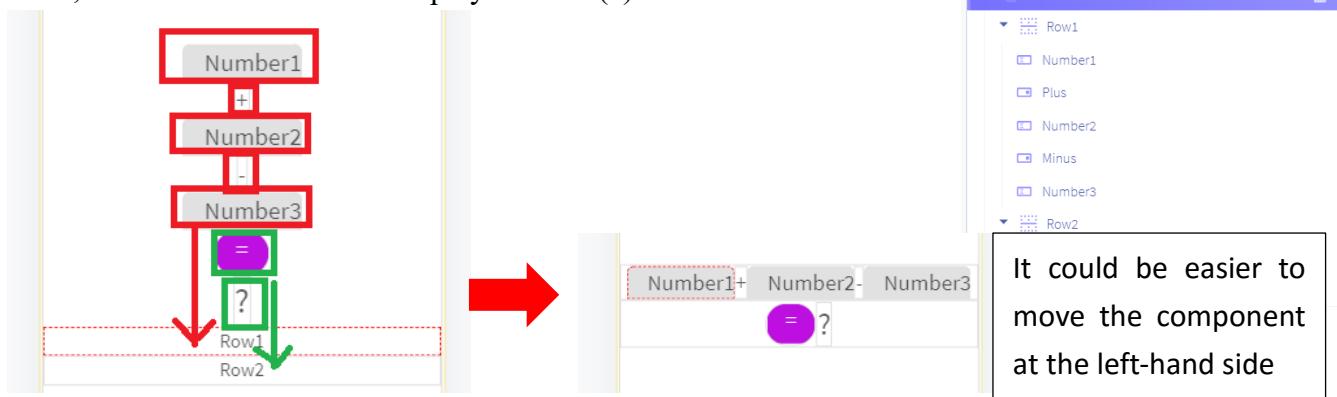
3. Select Row1 and change its Height to “Fit contents”.

Do the same to Row2.



4. Move “Number1”, “Plus” (+), “Number2”, “Minus” (-) and “Number3” to Row1.

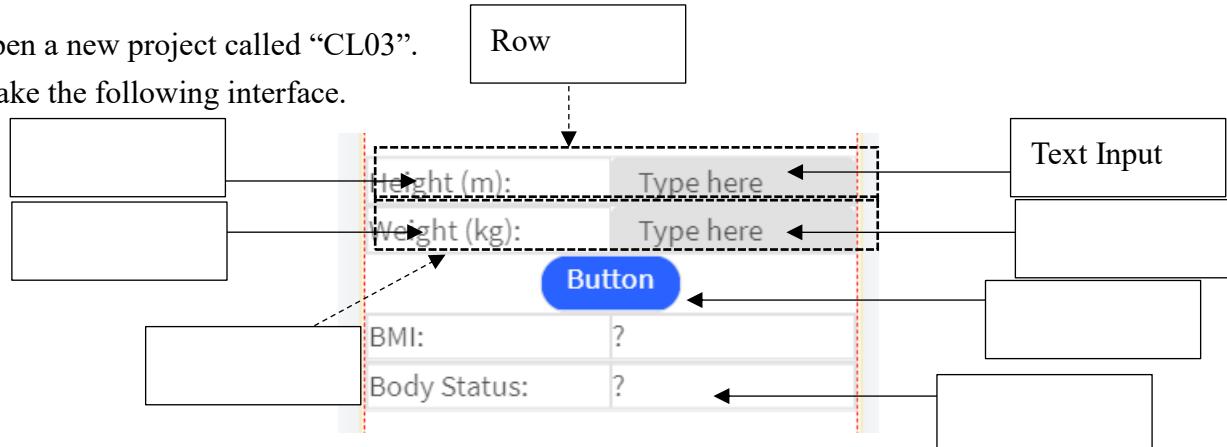
Then, move “= Button” and “Display Result” (?) to Row2.



○ BMI Calculator (1)

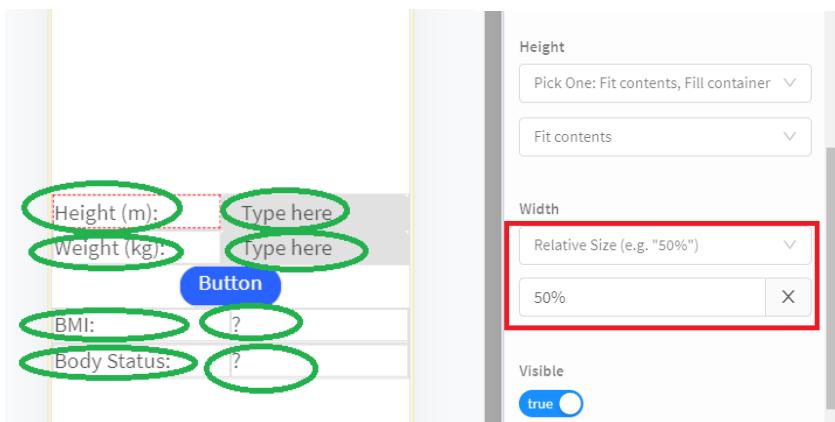
1. Open a new project called “CL03”.

2. Make the following interface.



Fill in the boxes above to indicate the type of component. Drag the components accordingly.

3. Select the Labels and Text Inputs, and change their Width to “Relative Size” and “50%”.



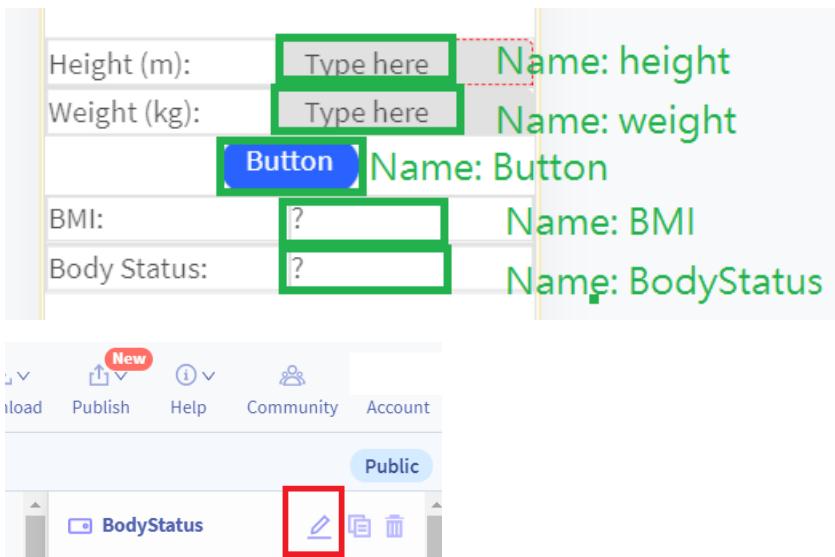
Apart from relative size and absolute size, there are two more types of options applicable to height and width.

Fit content: The height / width depends on the content (e.g. text) inside.

Fill container: The height / width depends on the container (e.g. screen) outside.



4. Change the name of the following components, for easier programming.



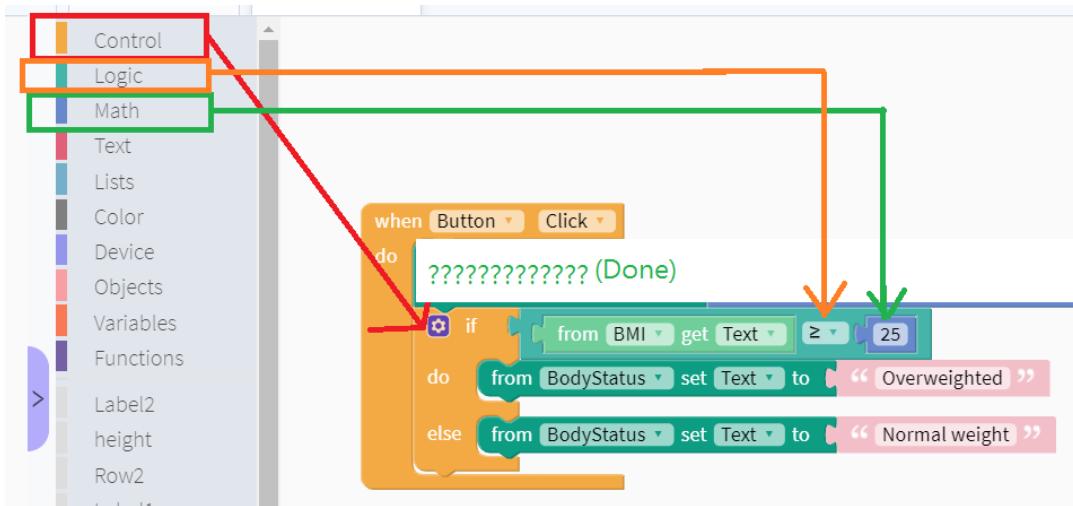
5. Click “Blocks” to start programming.

When the button is clicked, calculate BMI ($BMI = \text{Weight} \div (\text{Height} \times \text{Height})$).



6. Determine BodyStatus by the following table:

BMI	Body Status
$BMI \geq 25$	Overweight
$BMI < 25$	Normal weight



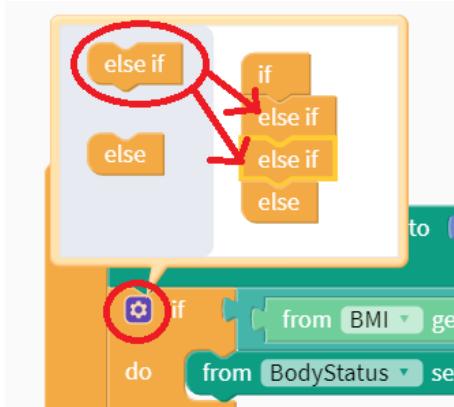
7. Test your program.

o BMI Calculator (2)

1. Modify the project “CL03”.
2. Determine BodyStatus by the following table:

BMI	Body Status
$BMI \geq 30.0$	Obese
$25.0 \leq BMI < 30.0$	Overweight
$18.5 \leq BMI < 25.0$	Normal weight
$BMI < 18.5$	Underweight

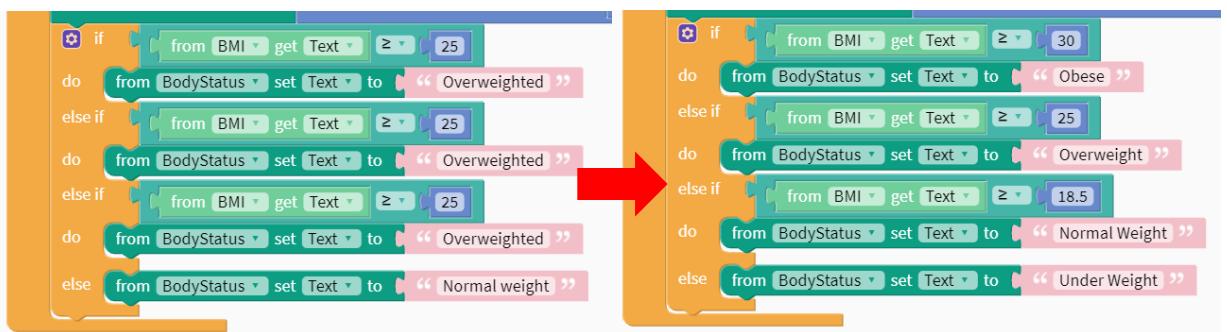
- (a) Click the blue button next to “if”, and drag 2 more “else if” blocks between if and else.



- (b) To save time, right-click the block “from BMI get Text ≥ 25” and select “Duplicate”.



(c) Change the content of the duplicated blocks.



3. Test the program.

Height (m):	<input type="text" value="1.65"/>
Weight (kg):	<input type="text" value="41"/>
<input type="button" value="Button"/>	
BMI:	15.05968778696
Body Status:	Under Weight

○ Exam Marks Calculator

Open a new project called “CL04”. The app would do the following.

- Accept input of marks of 3 subjects: Chinese, English and Mathematics
- When button is clicked, the average mark is calculated. The average mark is rounded off to the nearest integer.



- The grade is shown too.

Average mark	Grade
Mark \geq 85	A
$70 \leq$ Mark $<$ 85	B
$60 \leq$ Mark $<$ 70	C
$50 \leq$ Mark $<$ 60	D
BMI $<$ 50	E

Chinese:	<input type="text" value="90"/>
English:	<input type="text" value="90"/>
Mathematics:	<input type="text" value="80"/>
<input type="button" value="Button"/>	
Average Marks:	87
Grade:	A

3

Thunkable (3): Random number generation and use of pictures

In the previous lessons, we have learnt how to make an app with conditional statements.

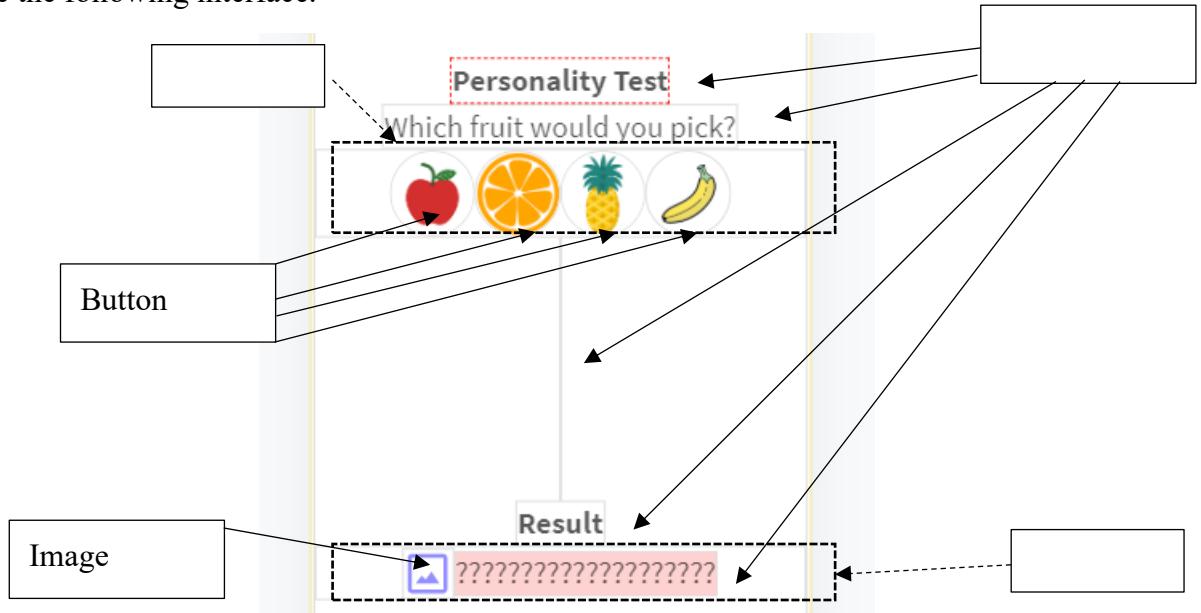
However, the app looks so plain with text only. Therefore, you will learn how to import pictures to your app in this lesson. Also, you will also learn how to draw random number.

○ Personality Test

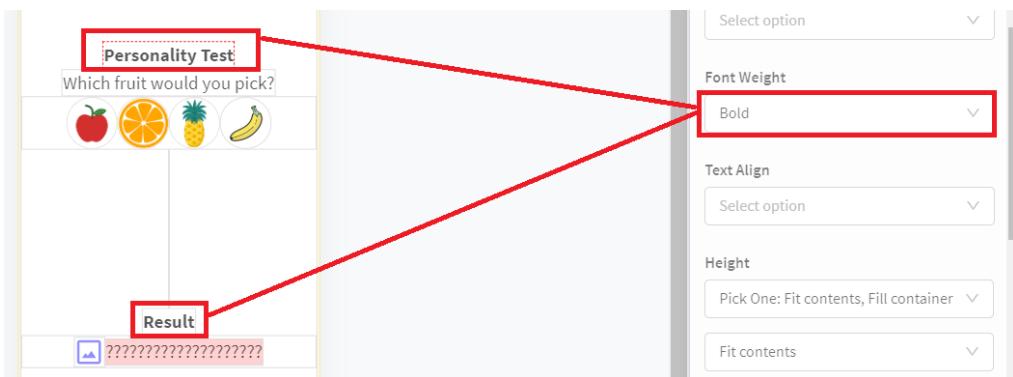
1. Open a new project called “CL05”.
2. Click “Choose a File”, and select “apple.png”, “orange.png”, “pineapple.png” and “banana.png”.



3. Create the following interface.



(a) Change the Font Weight of “Personality Test” and “Result” into Bold.

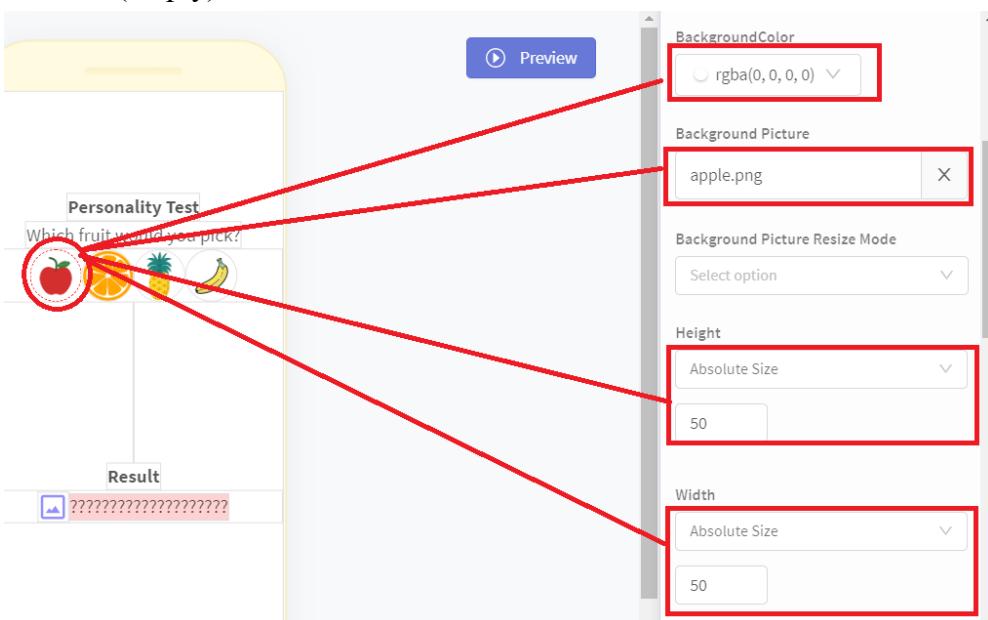


(b) Change the Height of the empty label into “Relative Size” “30%”.

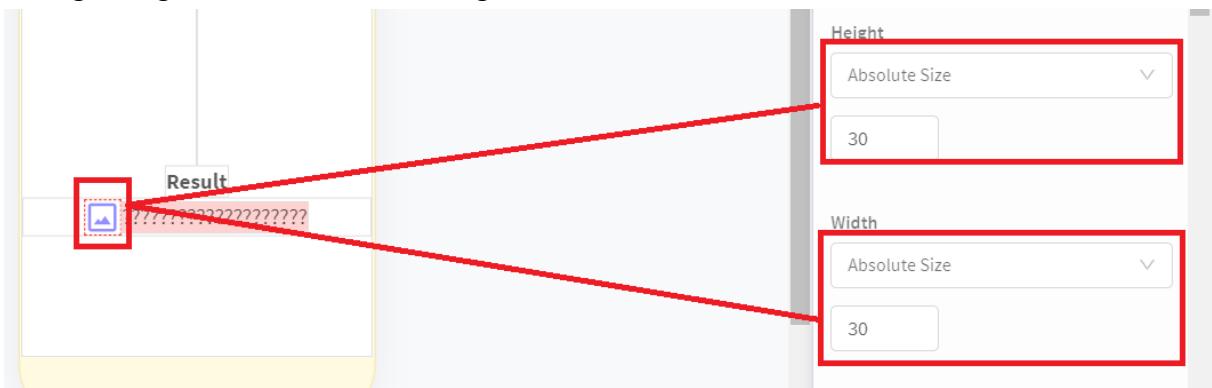


(c) Change the following attributes of the four buttons:

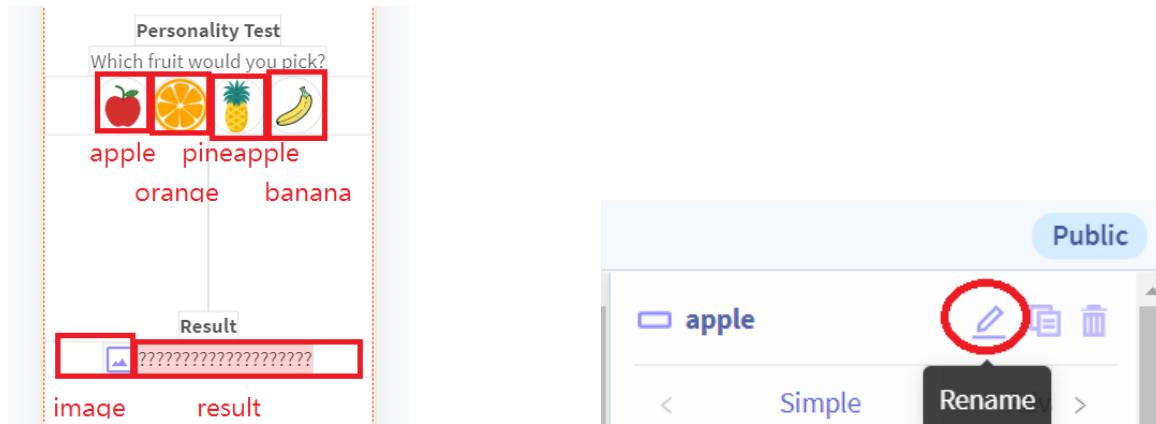
- Background Color (into transparent
- Background Picture (“apple.png”, “orange.png”, “pineapple.png”, “banana.png”)
- Height and Width (into “Absolute Size” “50”)
- Text (empty)



(d) Change Height and Width of the image into “Absolute Size” “30”.



4. Change the name of the following components, for easier programming.



5. Click “Blocks” to start programming.

When “apple” is clicked, “result” and “image” will change accordingly.

```
when apple [Click]
do [from result set Text to "You are extravagant, impulsive and outspoken, often with a bit of a temper."
  from image set Picture to apple.png]
```

6. Also, complete the blocks for orange, pineapple and banana. (You can change the result as you like)

Apple	You are extravagant, impulsive and outspoken, often with a bit of a temper.
Orange	You have enduring patience and willpower. You like to do things slowly; but very thoroughly and are completely undaunted by hard work.
Pineapple	You are quick to decide and even quicker to act. You are brave in making career changes; if that is what is to your advantage.
Banana	You are a softie! Loving, gentle, warm and sympathetic by nature.

7. Test the program.

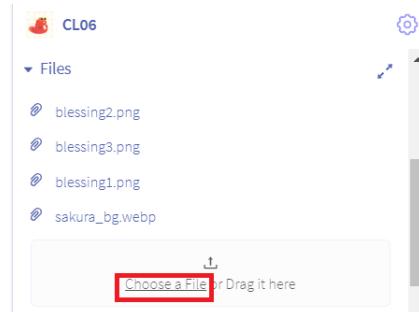
○ Omikuji

Omikuji is a kind of Japanese culture. It refers to paper strip that can be found at shrines and temples in Japan. It includes a general blessing which can be:

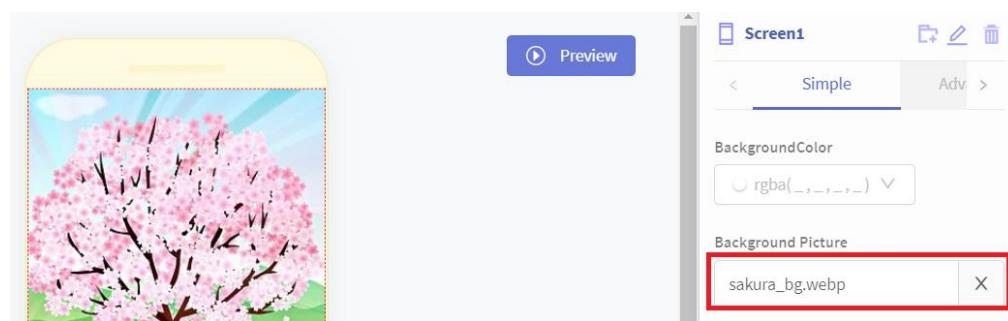
- (a) Great blessing (大吉)
- (b) Middle blessing (中吉)
- (c) Small blessing (小吉)



1. Open a new project called “CL06”.
2. Click “Choose a File”, and select “blessing1.png”, “blessing2.png”, “blessing3.png” and “sakura_bg.webp”.



3. Change Screen1 Background Picture into “sakura_bg.webp”.



4. Create the following interface by inserting an image and a button.

Change the Height and Width of the image into Absolute Size “144” and “70” respectively.



5. Click “Blocks” to start programming.

When “draw” is clicked, the image will randomly change to either “blessing1.png”, “blessing2.png” or “blessing3.png”.



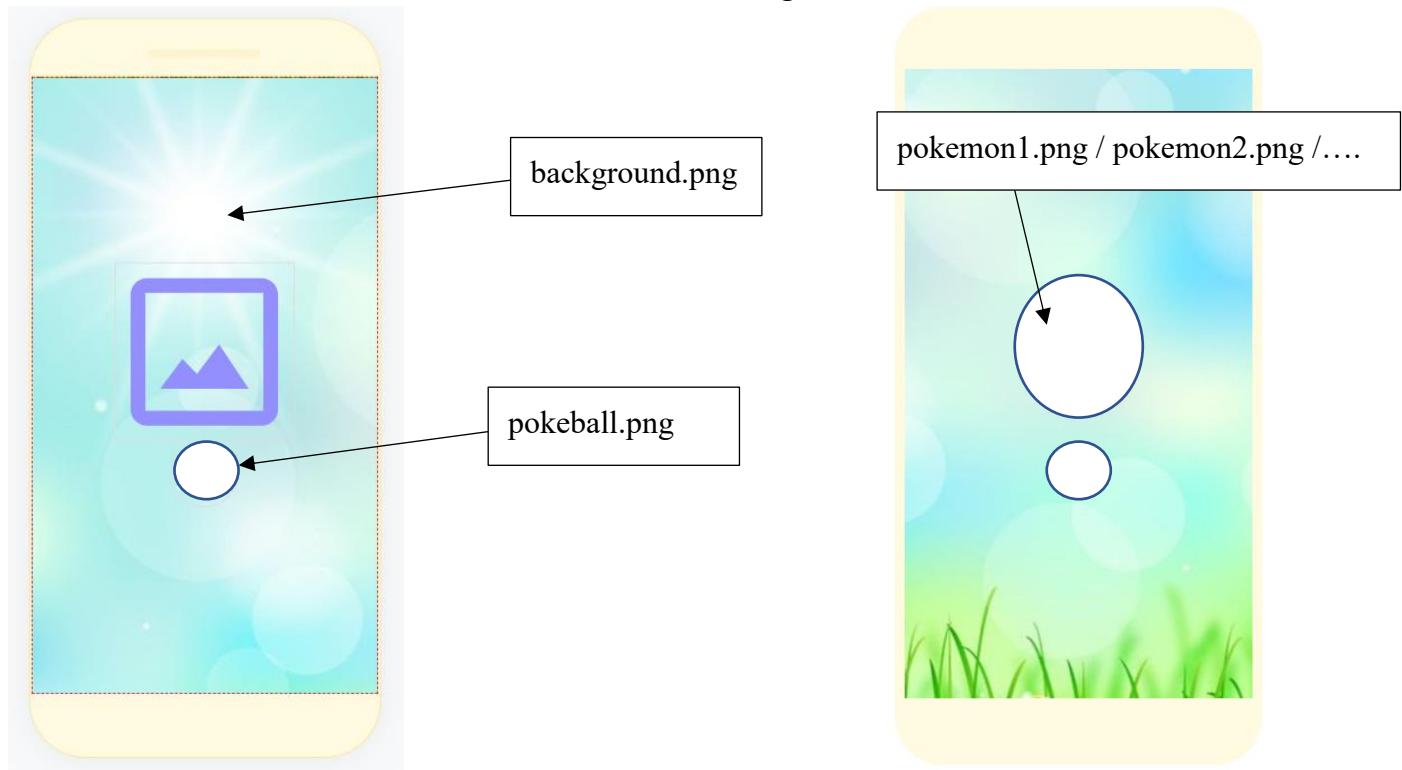
6. Test your program.



○ Gotta Catch Them All!

Open a new project called “CL07”. The app would do the following:

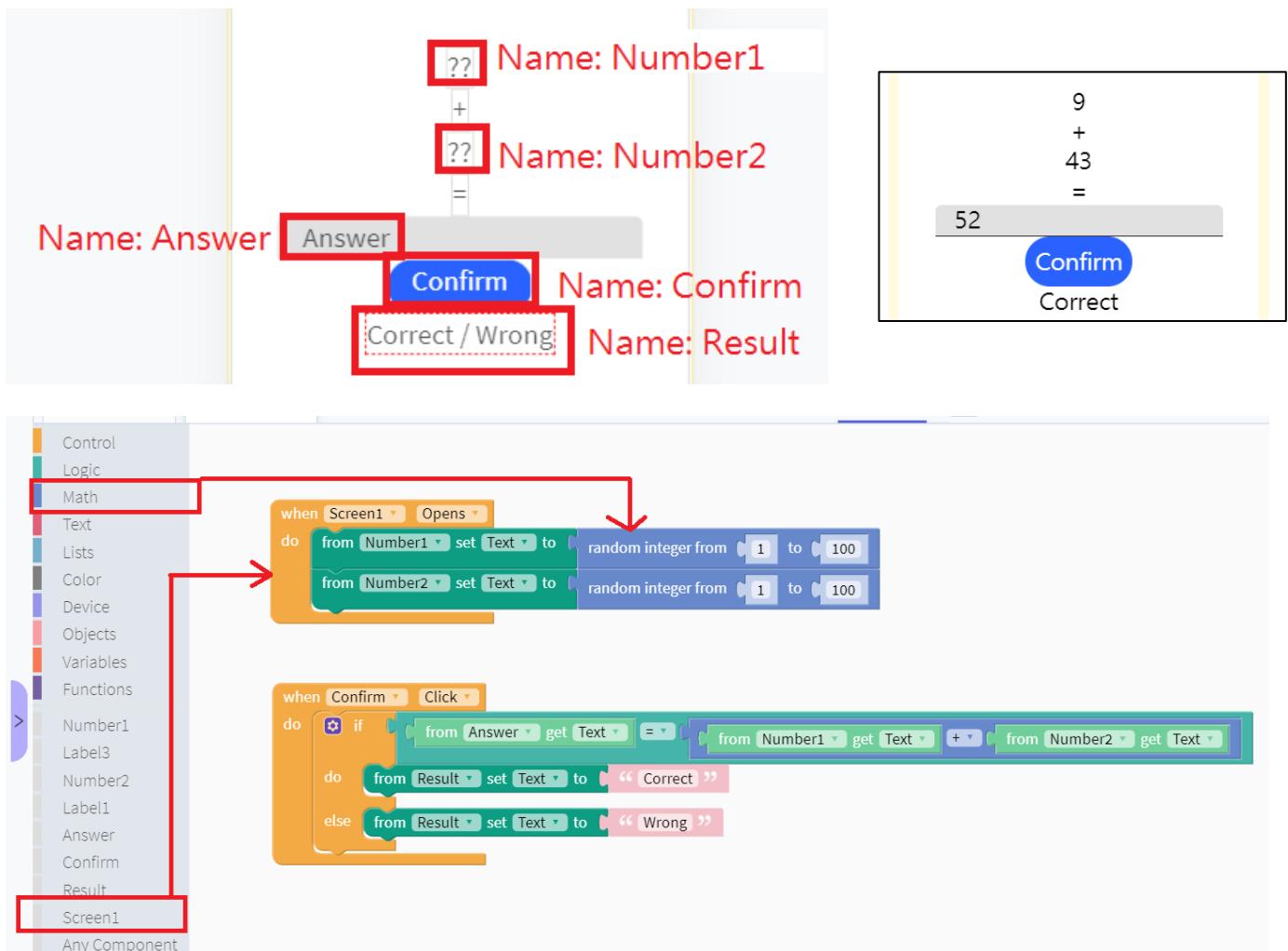
When the “ball” button is clicked, a random “monster” image is shown.



o Math Quiz (1)

Open a new project called “CL08”. The app would do the following:

- At the beginning, Text of Number1 and Number2 are set to random integer between 1 and 100.
- Accept input of Answer
- When Confirm button is clicked, it checks whether $\text{Answer} = \text{Number1} + \text{Number2}$:
- (a) If true, Result shows “Correct”.
- (b) If false, Result shows “Wrong”.



o Math Quiz (2) **

Modify the project “CL08” so that it would count the score, and able to generate more questions.

- (if Result = “Correct / Wrong”)

When “Confirm” is clicked, Text of button changes to “Next Question”

Text of Result changes to either “Correct” or “Wrong”

Text of Mark increases or decreases by 1

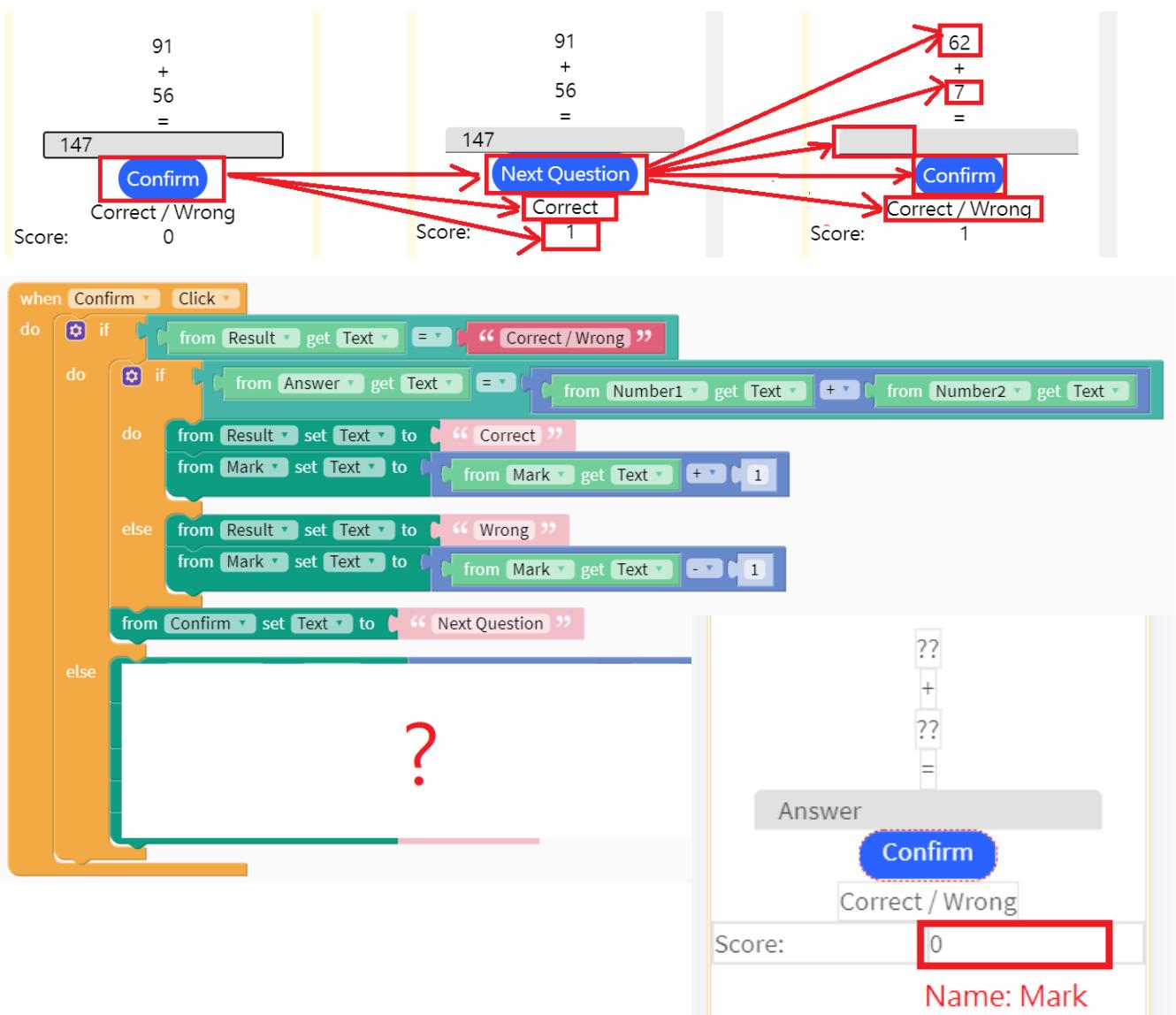
- (else)

When “Next Question” is clicked, Text of button changes to “Confirm”

Text of Result changes to “Correct / Wrong”

Text of Answer becomes empty

Text of Number1 and Number2 set to random integer (1 – 100)



4

Thunkable (4): Time and Location

So far we have learnt about the basic interface and programming in Thunkable.

If we would like to make a more complicated app, we should be able to get the user's current location and time, so as to provide timely and useful information.

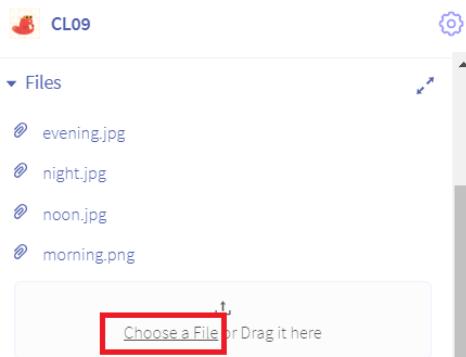
For example, *OpenRice* app would suggest restaurants nearby (need to get user's location)

Alarm app would ring at a particular time (need to get the current time)



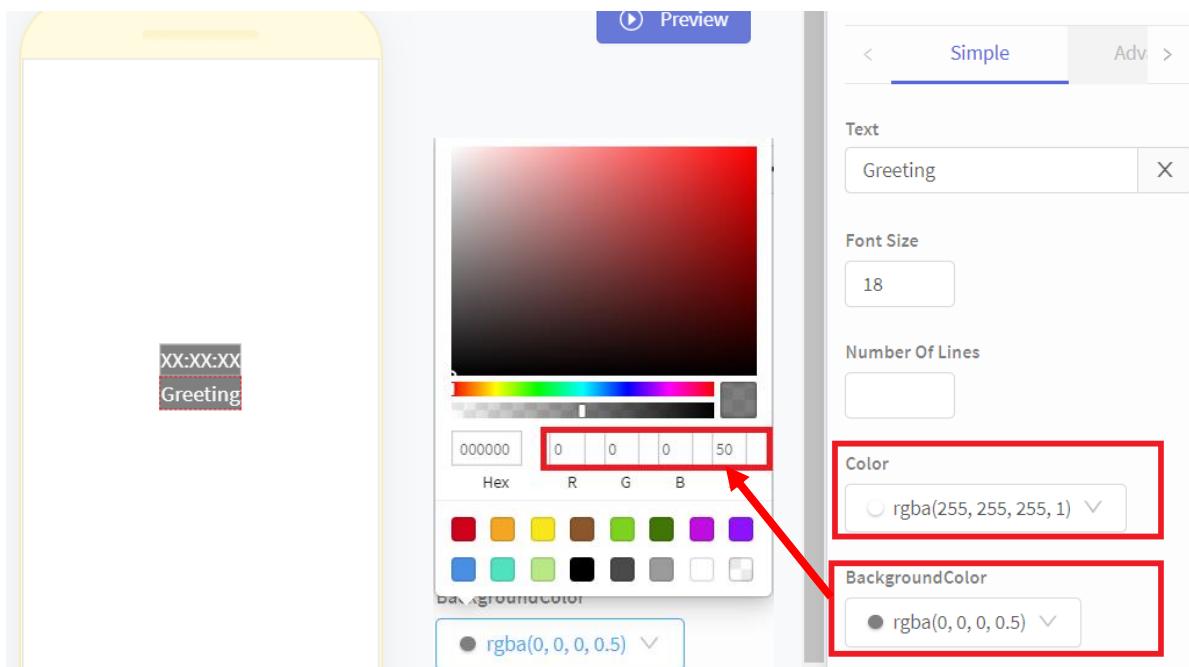
○ Greeting App

1. Open a new project called “CL09”.
2. Click “Choose a File”, and select “evening.jpg”, “night.jpg”, “noon.jpg” and “morning.png”.



3. Create the following interface by inserting two labels.

In the labels, change the Color (R 255, G 255, B 255, A 100) and
BackgroundColor (R 0, G 0, B 0, A 50).



RGBA Colour Model

In RGBA Model, each colour is represented by three values ranging from 0 to 255.

R	G	B
0 ~ 255	, 0 ~ 255	, 0 ~ 255

For example, red colour is (255, 0, 0)

Blue colour is (0, 0, 255)

Purple color is (255, 0, 255) (i.e. red + blue)

In addition, there is an Alpha channel ranging from 0 (0%) to 1 (100%).

0 (0%) means fully transparent (透明)

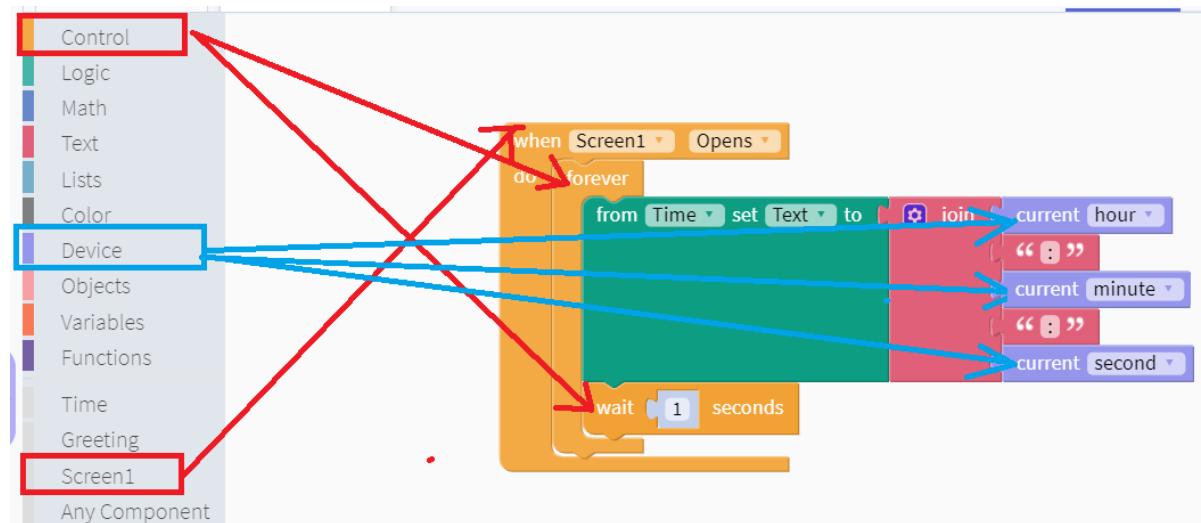
1 (100%) means fully opaque (不透光)

4. Change the name of the labels.



5. Click “Blocks” to start programming.

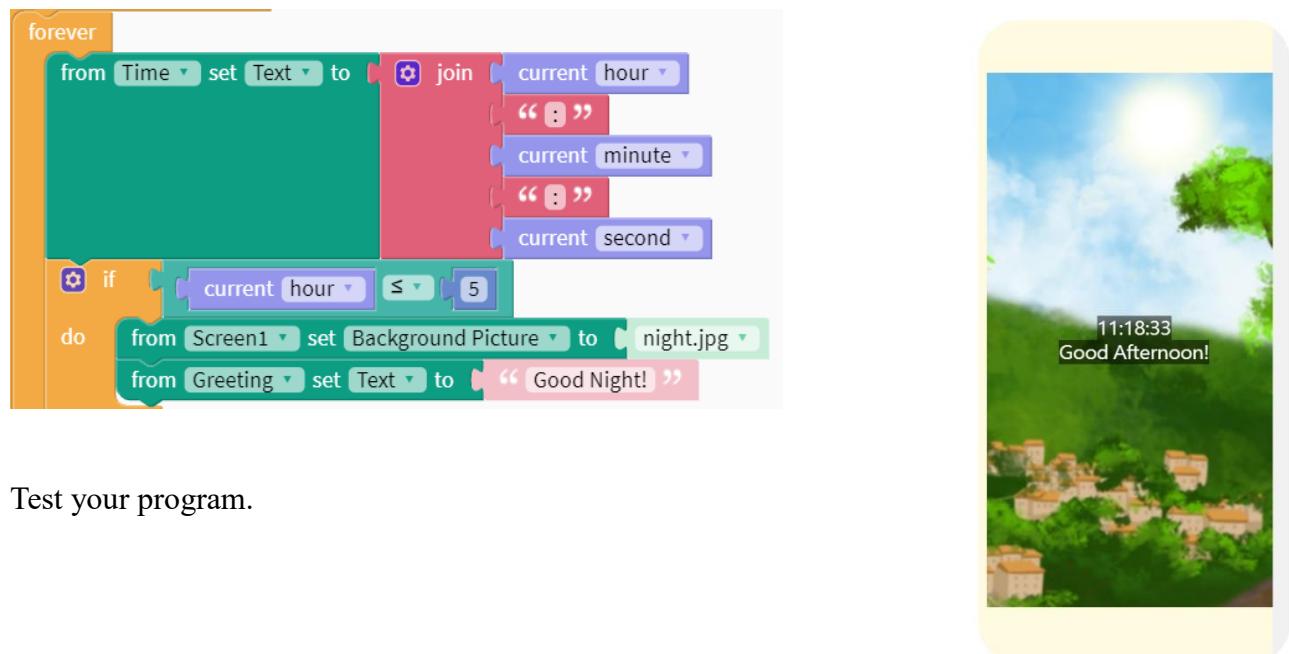
*When the screen opens, the time (hour:minute:second) is **repeatedly updated** (i.e. “forever”) in **every 1 second** (i.e. “wait 1 seconds”).*



6. Also, prepares the following programming blocks:

When current hour changes, the Background Picture of Screen and Text of Greeting would change accordingly:

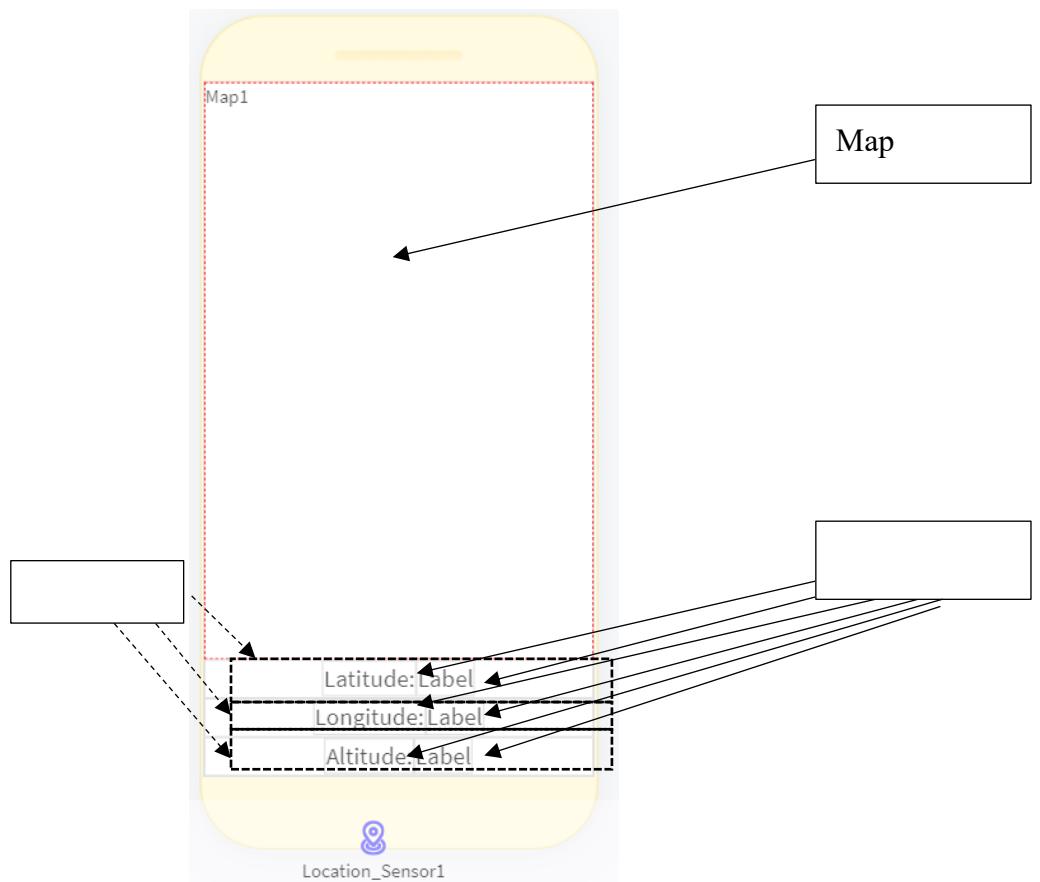
Current hour	Background Picture (Screen)	Text (Greeting)
Hour \leq 5	Night.jpg	Good Night!
$5 < \text{Hour} \leq 10$	Morning.png	Good Morning!
$10 < \text{Hour} \leq 16$	Noon.jpg	Good Afternoon!
$16 < \text{Hour} \leq 19$	Evening.jpg	Good Evening!
Hour $>$ 19	Night.jpg	Good Night!



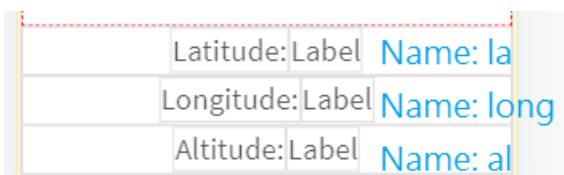
7. Test your program.

- **Where Am I? (1)** (*requires a physical device to test*)

1. Open a new project called “CL10”.
2. Drag **Map**, **Location Sensor** and other components into the interface.



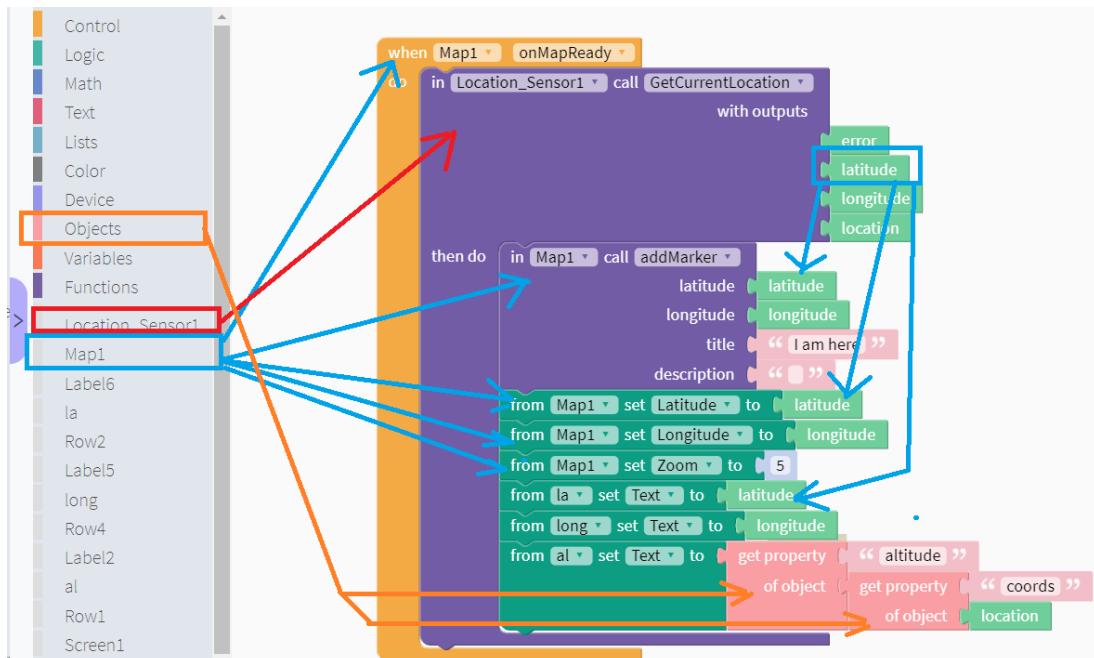
3. Change the name of the labels.



4. Click “Blocks” and start programming.

When Map is Ready (onMapReady):

- Get the current location from Location Sensor
- Add a Marker (addMarker) on the map
- set the latitude and longitude of Map, as well as its Zoom (放大) to 5
- set the text of labels la, long and al respectively.



Zoom of Map

The greater is the zoom, the smaller is the area being shown.



Zoom = 0



Zoom = 5

Location of GetCurrentLocation in Location Sensor

Apart from “coords >> altitudes”, there are “coords >> speed” and “coords >> heading” (direction).

5. Test the program.

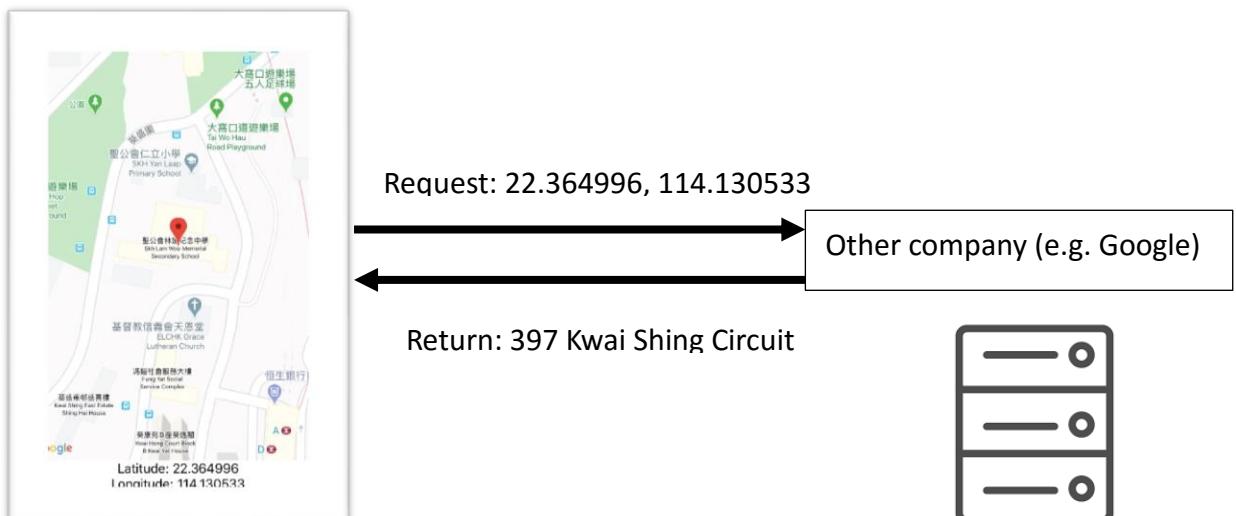
- Where Am I? (2) (*requires a physical device to test*) ** (*optional*)

Is it possible to know the address (e.g. Kwai Shing Circuit, Kwai Hing)?

Yes, but we need to use the API provided by other companies (e.g. Google, Positionstack).

API (Application Programming Interface)

API defines how a request to be made (e.g. latitude and longitude), and the data to be returned upon request (e.g. address of corresponding location).



Our mobile app

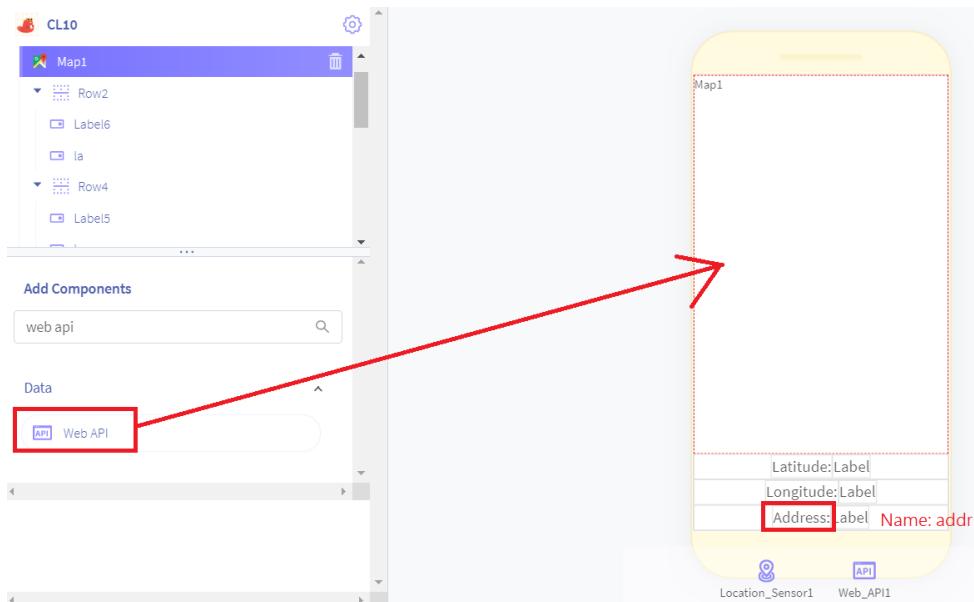
In later chapters, we may discuss how we could get the latest weather and finance information by using APIs.

In this example, we will use Positionstack as it is free of charge.

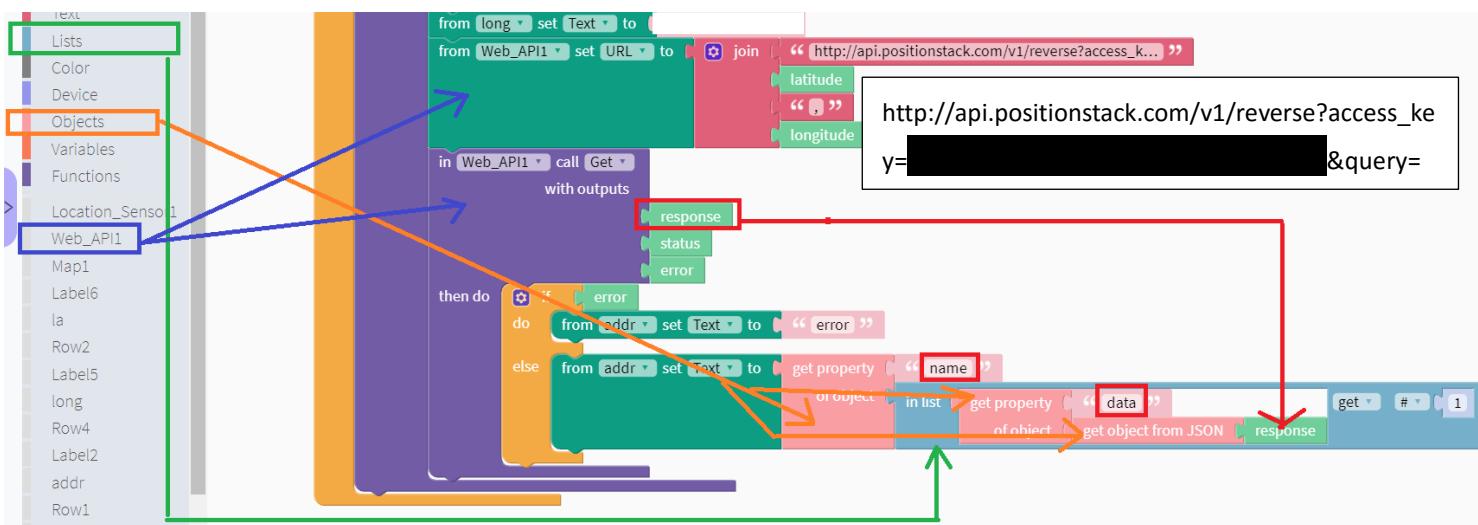
1. Go to http://api.positionstack.com/v1/reverse?access_key=XXXXXXXXXX&query=22.364996,114.130533

```
← → C ⓘ 不安全 | api.positionstack.com/v1/reverse?access_key=[REDACTED] &query=22.364996,114.130533 ⓘ ⭐ SK
{
  "data": [
    {
      "latitude": 22.364905,
      "longitude": 114.130607,
      "type": "venue",
      "distance": 0.013,
      "name": "\u0b05\u0516\u0670\u0679\u08b7\u07d0\u05ff\u0e2d\u05b7 Sheng Kung Hui Lam Woo Memorial Secondary School",
      "number": "397",
      "postal_code": null,
      "street": "Kwai Shing Circuit",
      "confidence": 0.8,
      "region": "Kwai Tsing",
      "region_code": "KI",
      "county": null,
      "locality": null,
      "administrative_area": null,
      "neighbourhood": null,
      "country": "Hong Kong S.A.R.",
      "country_code": "HK",
      "continent": "Asia",
      "label": "\u0b05\u0516\u0670\u0679\u08b7\u07d0\u05ff\u0e2d\u05b7 Sheng Kung Hui Lam Woo Memorial Secondary School, Hong Kong S.A.R."
    },
    {
      "latitude": 22.364905,
      "longitude": 114.130607,
      "type": "address",
      "distance": 0.013,
      "name": "397 Kwai Shing Circuit",
      "number": "397",
      "postal_code": null,
      "street": "Kwai Shing Circuit",
      "confidence": 0.8,
      "region": "Kwai Tsing",
      "region_code": "KI",
      "county": null,
      "locality": null,
      "administrative_area": null,
      "neighbourhood": null,
      "country": "Hong Kong"
    }
  ]
}
```

2. Modify the project “CL10” by adding “Web API”, and change the label “Altitude:” to “Address:”, and change the name of the next label from “al” to “addr”.



3. Click “Blocks” to start programming, so as to get the name of the 1st data from JSON response of Web API.



4. Test the program.



Thunkable (5): Use of AI Applications

In this chapter, we will learn how to use the following components to build AI (Artificial Intelligence) applications, e.g.

- Text To Speech
- Speech To Text (Speech Recognizer)
- Image Recognizer
- Translator between different languages

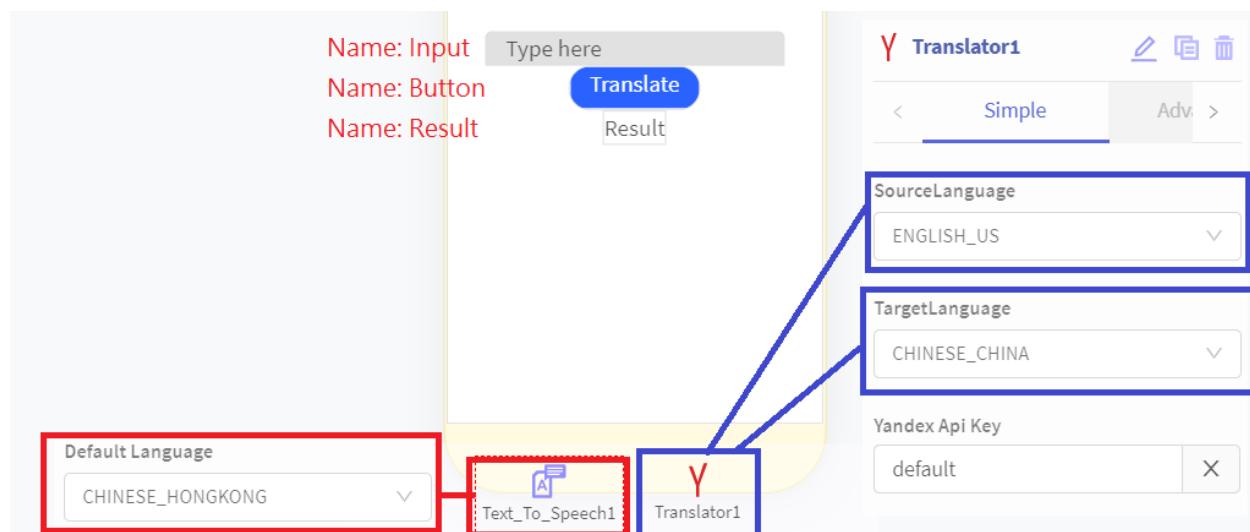
■ English-to-Chinese Translator (1)

1. Open a new project called “CL11”.
2. Drag **Text To Speech**, **Translator** and other components into the interface.

Change the name of each component accordingly.

Change the Default Language of Text To Speech into “CHINESE_HONGKONG”.

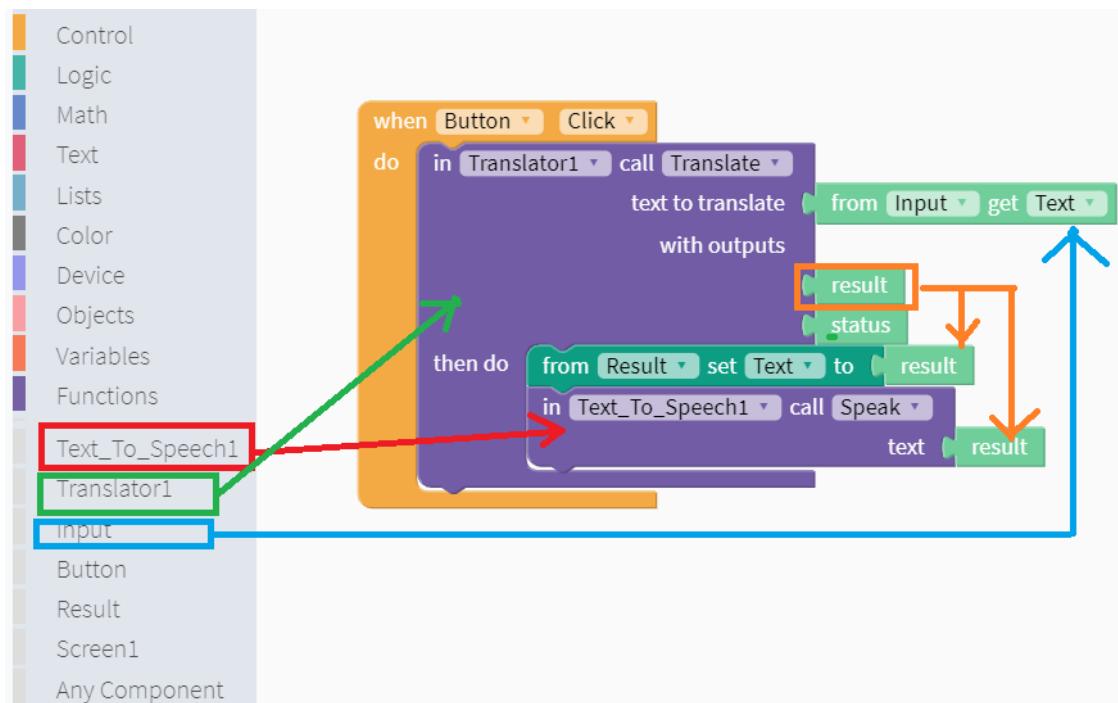
Change the SourceLanguage and TargetLanguage of Translator into “ENGLISH_US” and “CHINESE_CHINA” respectively.



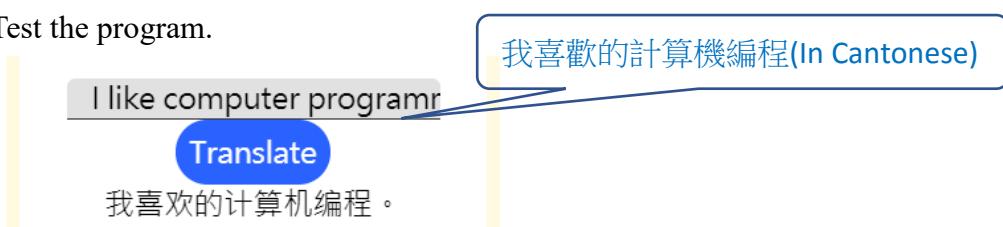
3. Click “Blocks” to start programming:

When the Button is clicked:

- Translator would translate the text from Input
- Text of Result would change to Translator’s output (result)
- Text To Speech would speak the Translator’s output (result)

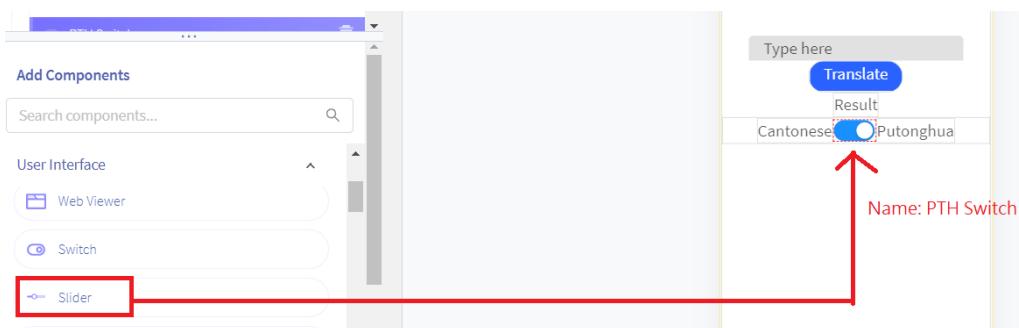


4. Test the program.



5. Modify the program so that it can speak either Cantonese or Putonghua.

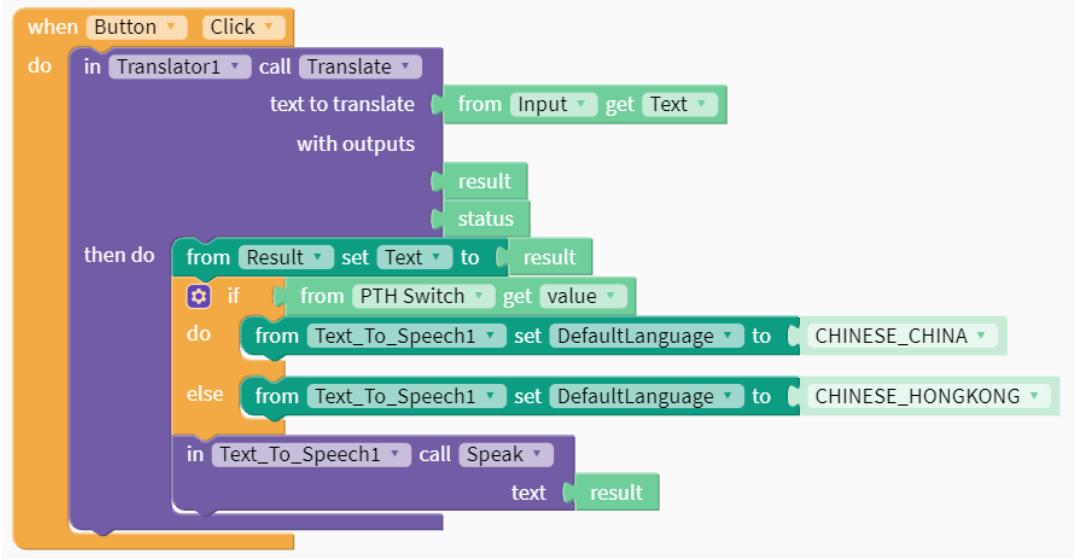
- (a) Create the interface by adding Row, Slider and two Labels



(b) Click “Blocks” to start programming:

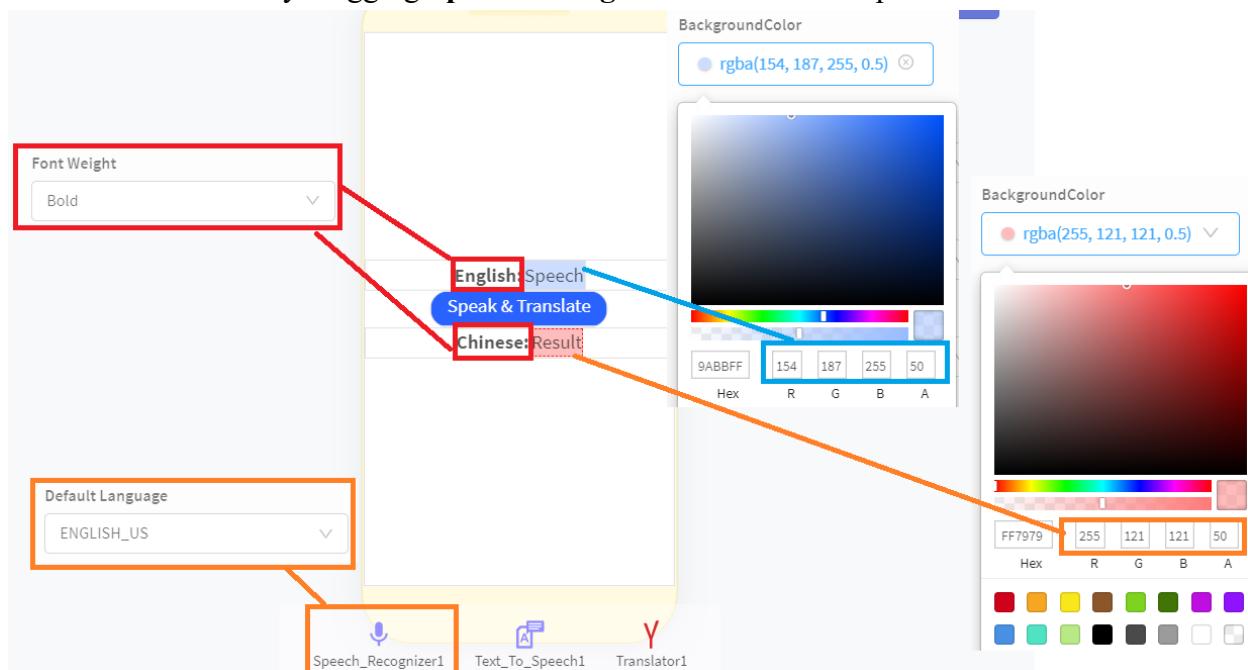
If “PTH Switch” = true (on)

→ then DefaultLanguage of Text To Speech becomes CHINESE_CHINA (Putonghua)
Otherwise, DefaultLanguage of Text To Speech becomes CHINESE_HONGKONG (Cantonese)



■ English-to-Chinese Translator (2)

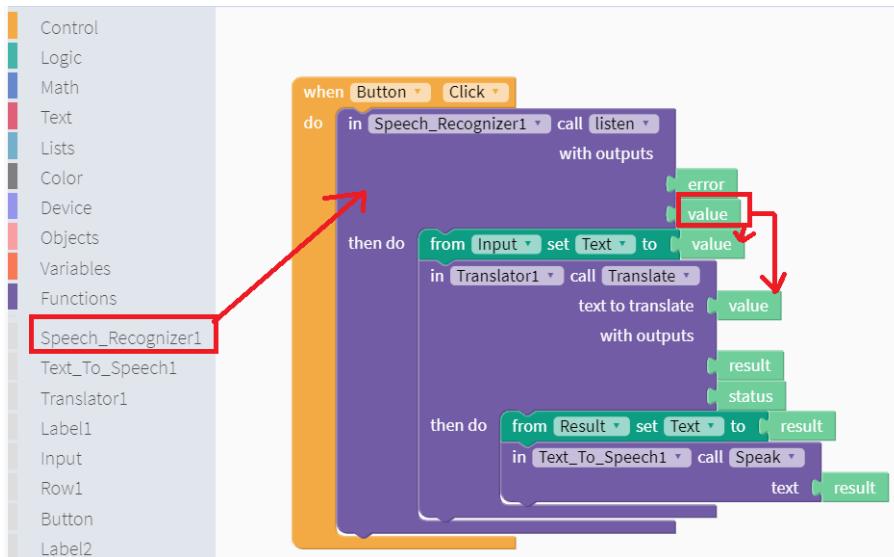
1. Create a new project “CL12”.
2. Create the interface by dragging **Speech Recognizer** and other components.



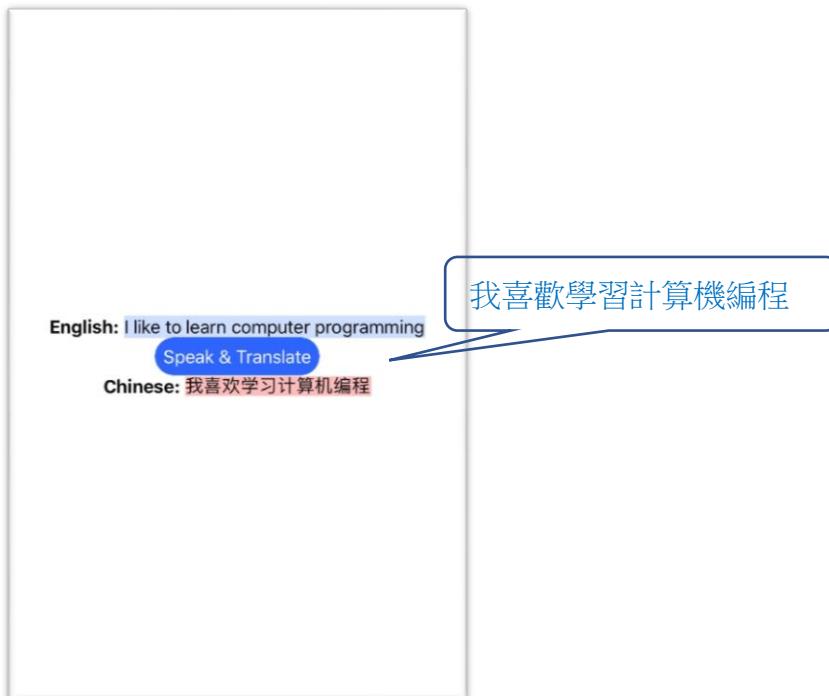
3. Click “Blocks” to start programming:

When the Button is clicked:

- Speech Recognizer will listen and return value
- Text of Input will become the return from Speech Recognizer (value)
- Translator would translate the text from the return from Speech Recognizer (value)
- Text of Result would change to Translator's output (result)
- Text To Speech would speak the Translator's output (result)

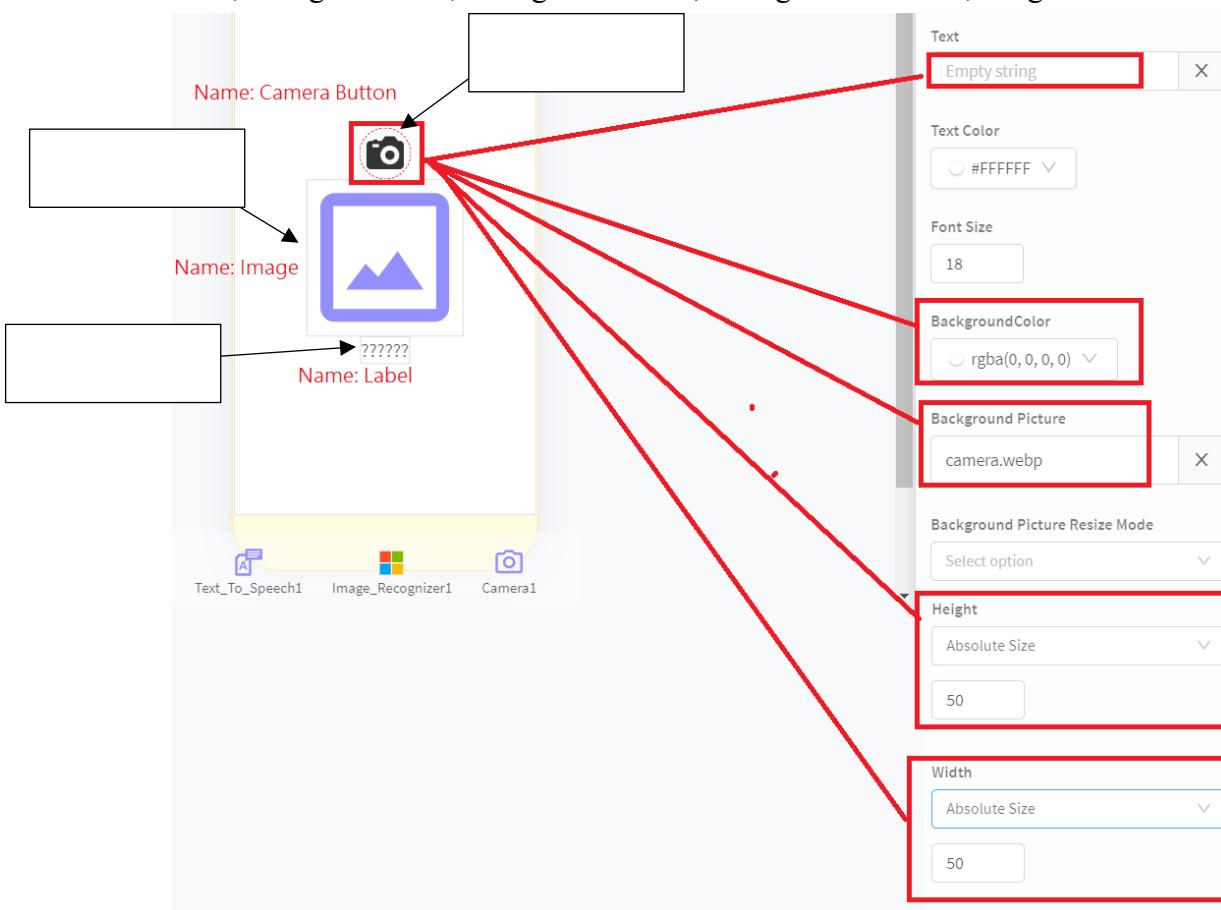


4. Test the program.



■ Tell me what it is! (1)

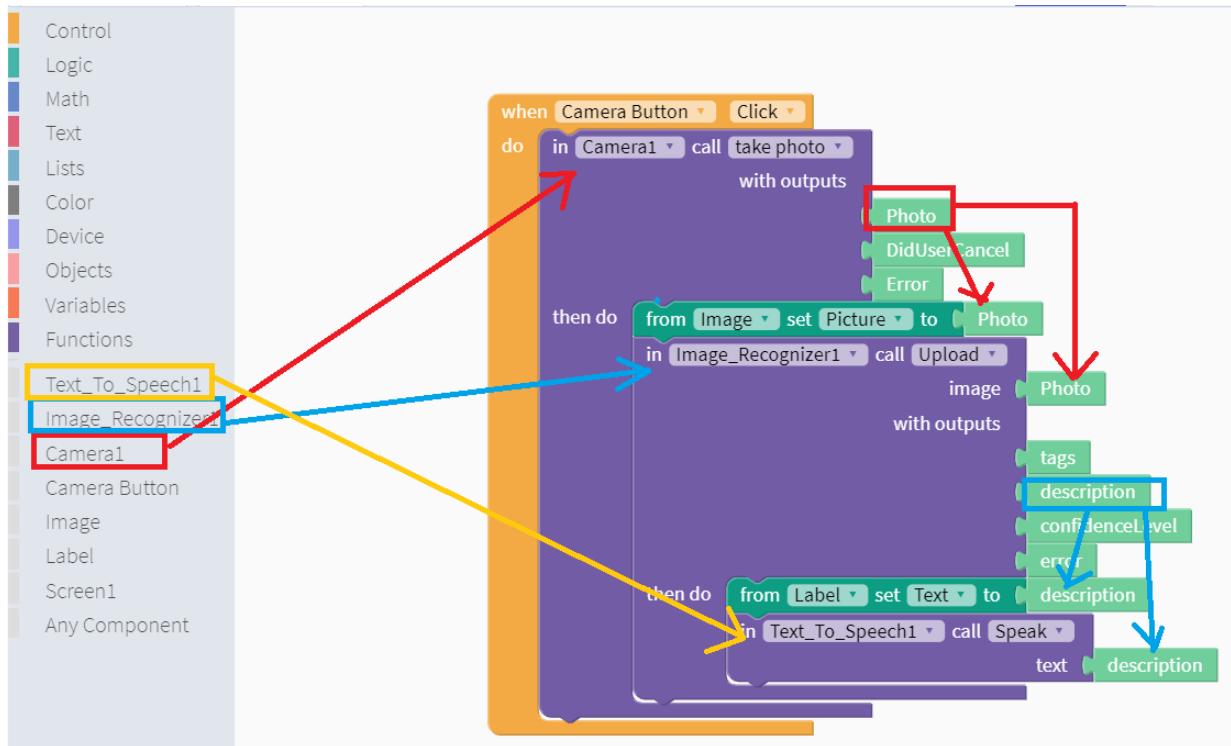
1. Create a new project “CL13”.
2. Upload a File (camera.webp) into the app.
3. Create the following interface by dragging **Text to Speech**, **Image Recognizer**, **Camera**, and other components. Fill in the boxes below. Change the name of the components.
Also, change the Text, BackgroundColor, Background Picture, Height and Width of Camera Button.



4. Click “Blocks” to start programming:

When the Camera Button is clicked:

- Camera takes photo, and return output photo
- Picture of image becomes the output photo from Camera
- Image Recognizer upload the output from Camera, and output descriptions of the photo
- Text of Label becomes the description output by Image Recognizer
- Text to Speech speaks the description



5. Test the program.



■ **Tell me what it is! (2) ****

1. Modify the project “CL13”.

2. If the photo you took contains orange, apple, banana or pineapple, it would show the corresponding image (orange.png, apple.png, banana.png, pineapple.png):



a glass of orange juice



a banana sitting on top of a wooden table



3. Otherwise, it would show a cross (cross.png).



a plate of food with broccoli



4. The only hint is, you need to check if the description contains the keyword (e.g. orange).

The screenshot shows a Scratch script editor. On the left, there's a sidebar with categories: Text (highlighted in red), Lists, Color, Device, Objects, and Variables. The main workspace contains the following script:

```
when green flag clicked
  [if   
 [does [description]   
 contain   
 [orange?]]  
 then  
   [join   
 [hello]  
 ,  
 [world]  
 v  
 end  
 ]  
 end]
```

6

Thunkable (6): Use of canvas for game making

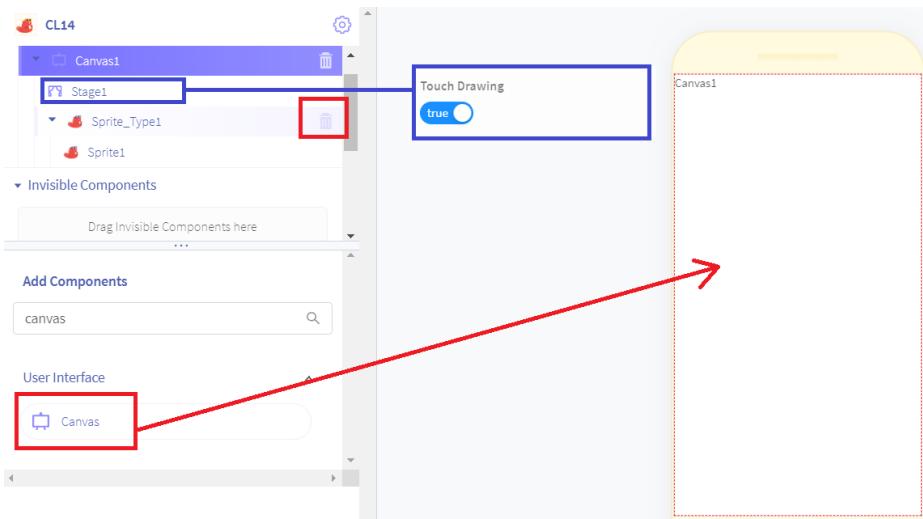
In this chapter, we will start making some games by canvas, where different sprites (e.g. characters) can be made on it.

■ Drawing app

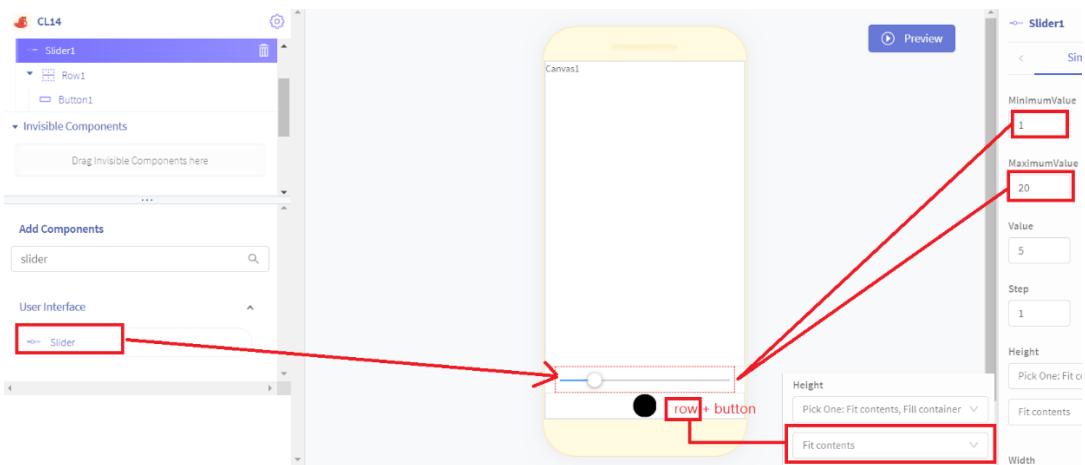
1. Create a new project “CL14”.
2. Create the following interface by dragging Canvas.

Then, delete Spirite_Type1

Also, change Touch Drawing of Stage1 to true.

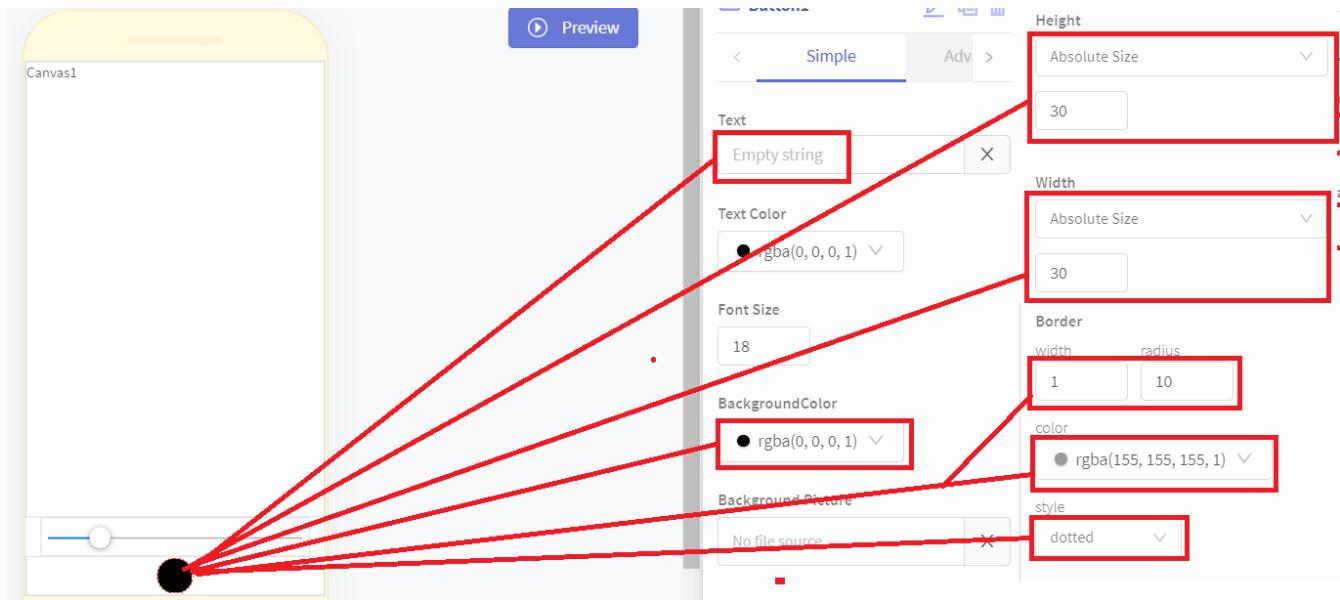


3. Add Slider into the interface, and change its MinimumValue and MaximumValue into 1 and 20 respectively. Also, add row and button, and change Height of row into “Fit contents”.



4. Change the setting of the button:

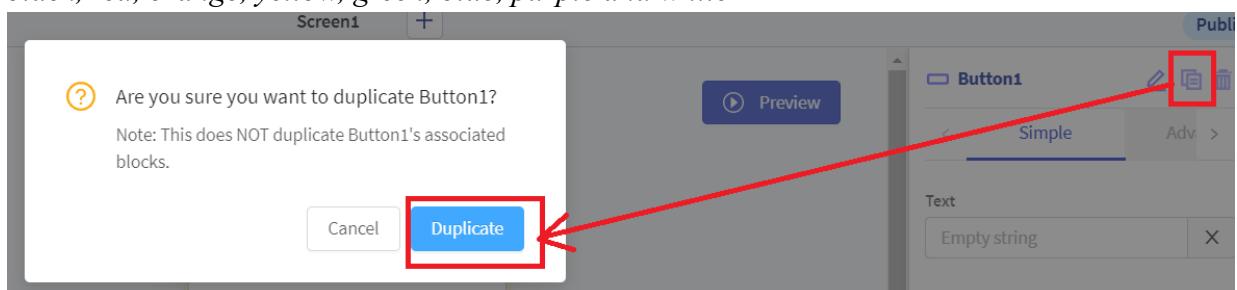
- Text becomes empty
- BackgroundColor becomes black
- Height and Width become “Absolute Size” “30”
- Border: width becomes 1, radius becomes 10, color becomes grey, and style becomes dotted



5. Duplicate the Button so that there are 8 buttons.

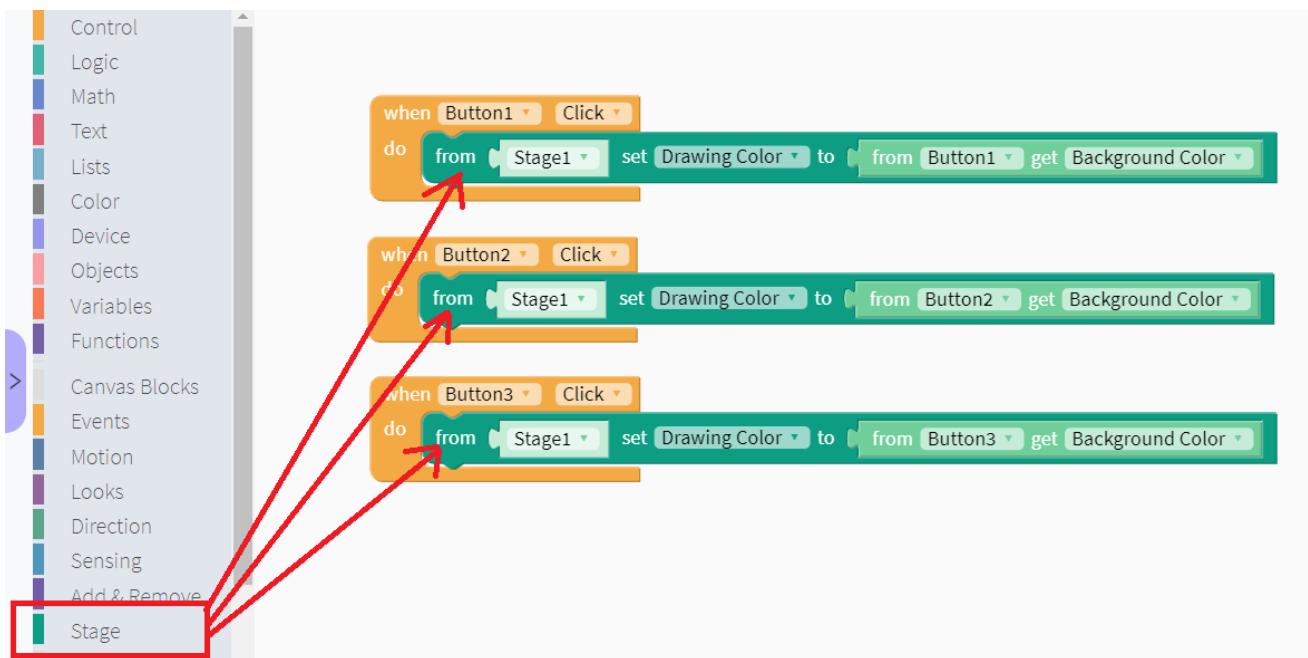
Then, change the BackgroundColor of each Button:

black, red, orange, yellow, green, blue, purple and white.

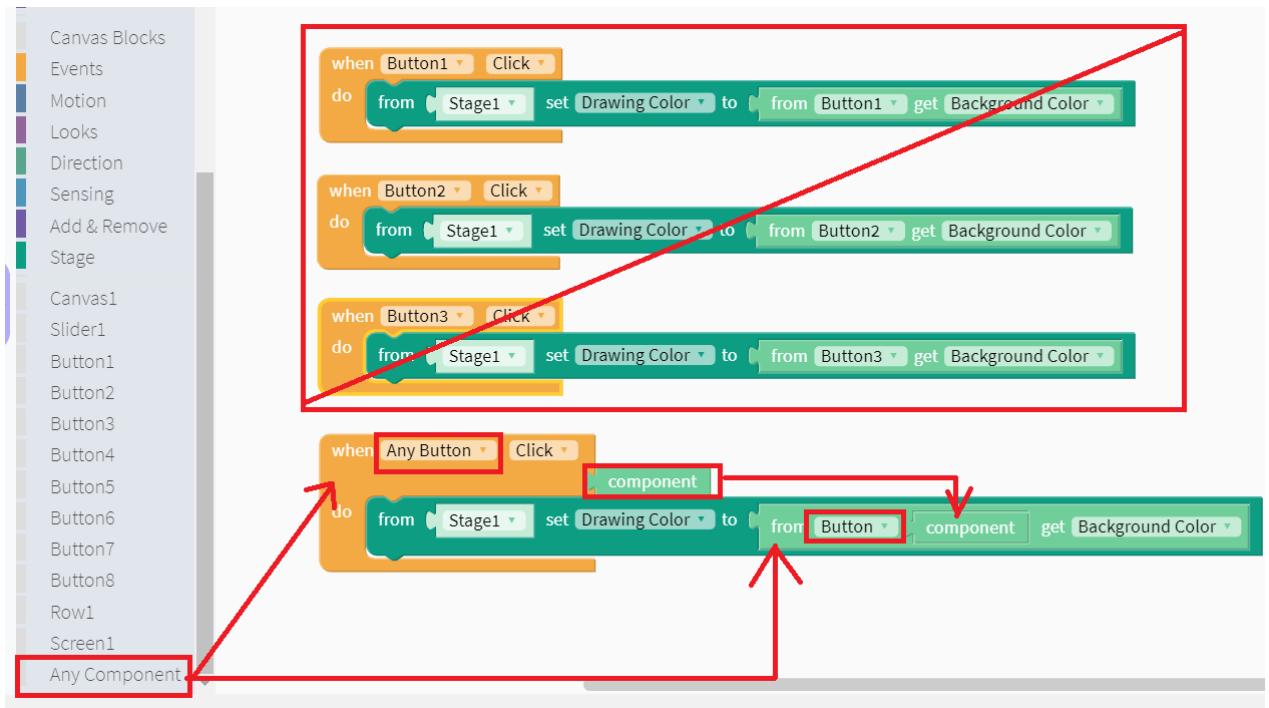


6. Click “Blocks” to start programming, so that:

When each colour button is clicked, the Drawing Color would change into the Background Colour of the colour button (e.g. when click the red button, the drawing color would change into red)

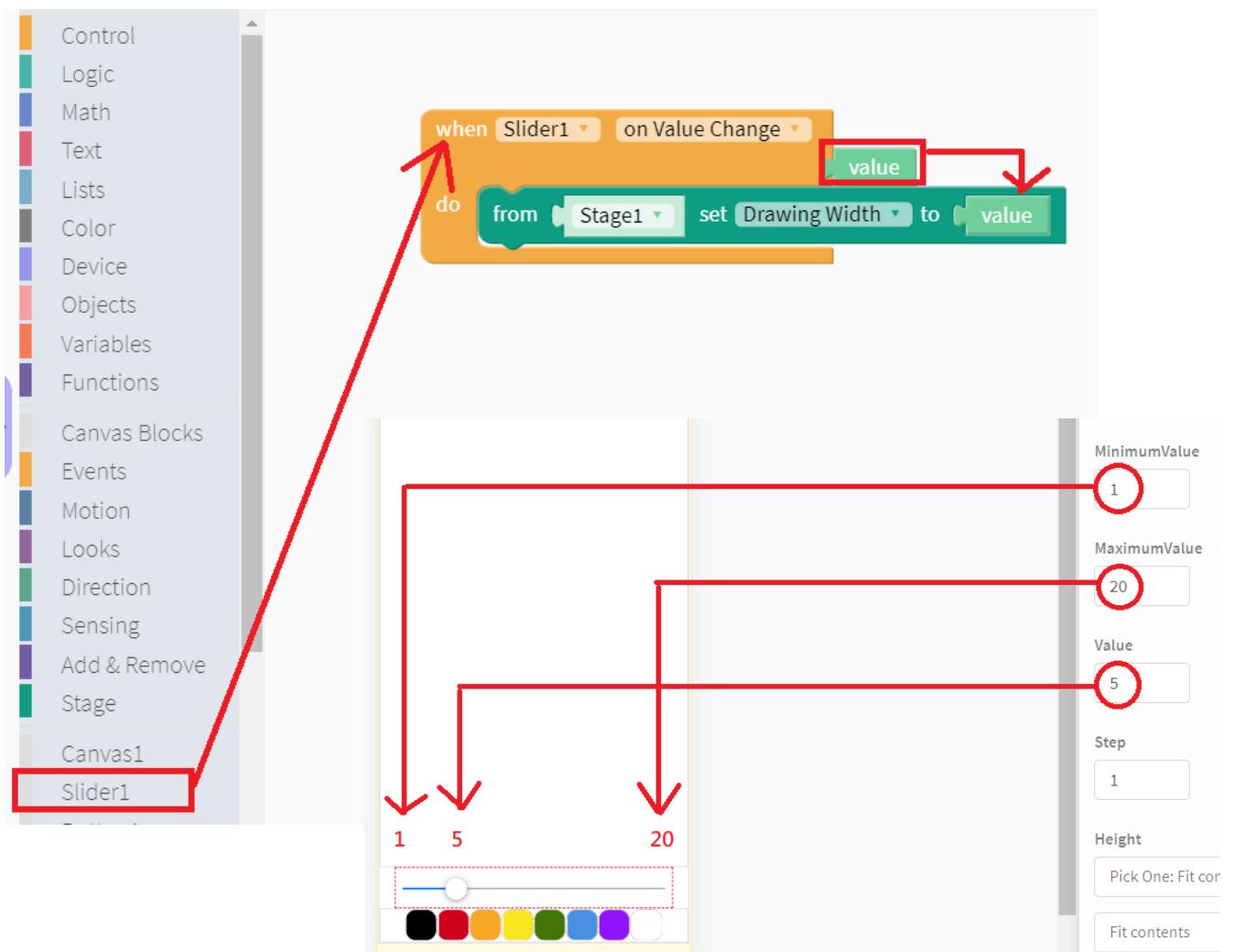


(Challenging) So, you need to make similar blocks by 8 times. In fact, you can use the following “Any Button” block to replace those 8 similar blocks.

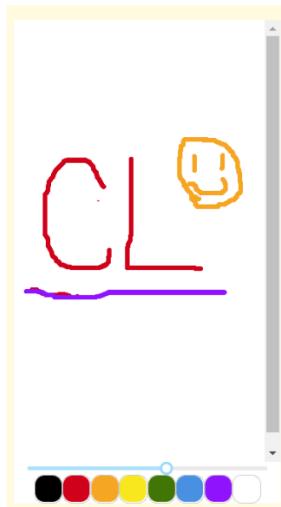


7. Moreover, for the slider, it would change the Drawing Width.

As you have changed before, the minimum value and maximum value of the slider are 1 and 20 respectively. So, when the slider is at its maximum, the value is 20, and thus the drawing width is 20.

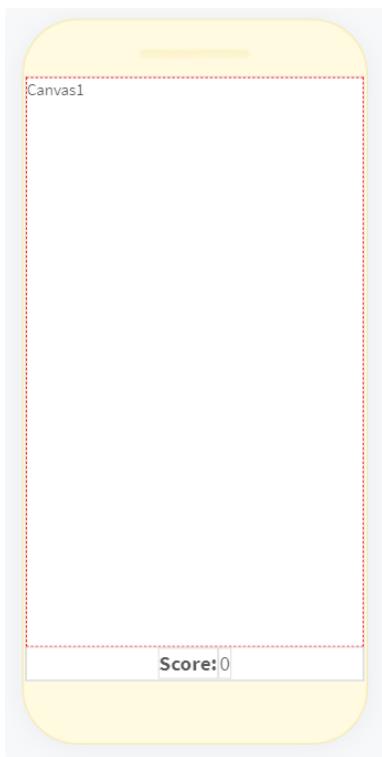


8. Test the program.



■ Wrack a mole (打地鼠) (1)

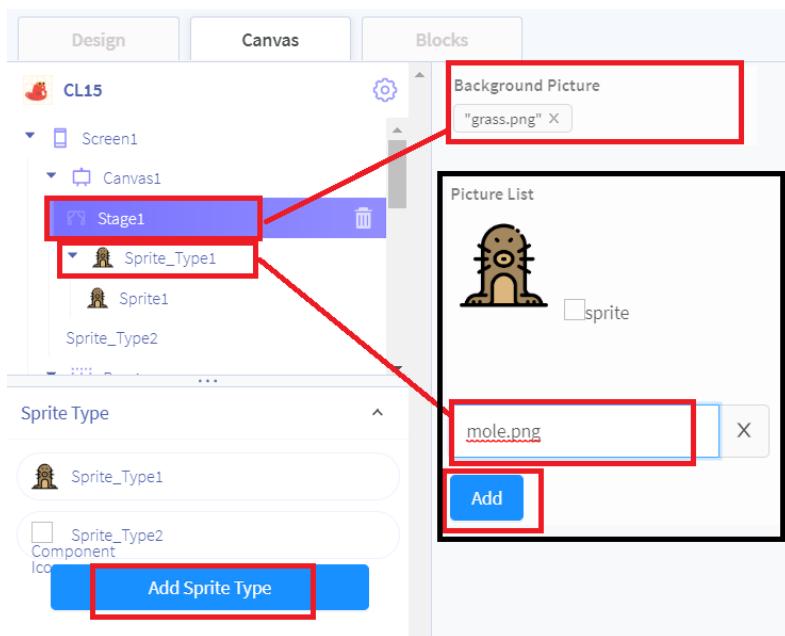
1. Create a new project “CL15”.
2. Create the following interface by dragging Canvas and other components



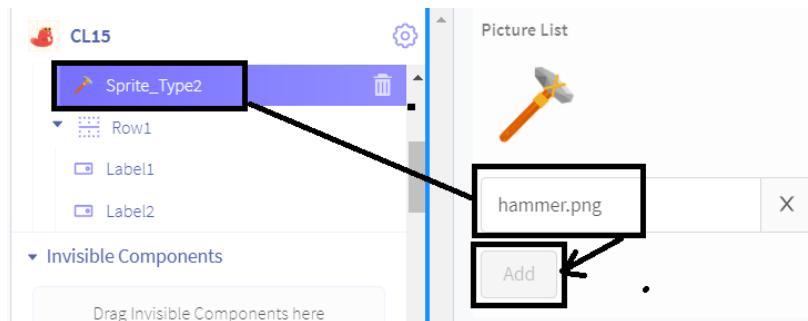
3. Upload the files into the app, including grass.png, mole.png and hammer.png.



4. Select Stage1, and choose “grass.png” as Background Picture.
Select Sprite_Type1, and choose “mole.png” in Picture List and click “Add”.
Select “Add Sprite Type”.

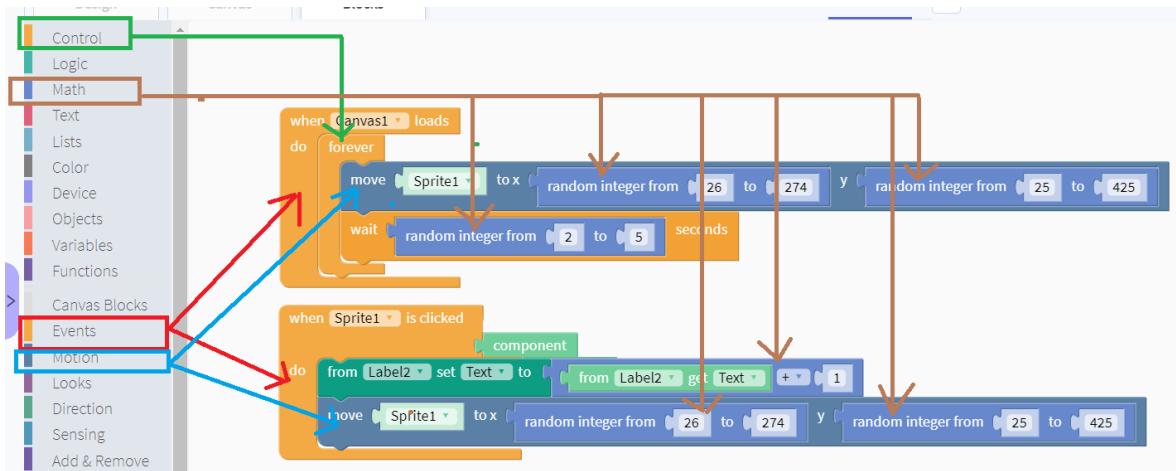


5. Select Sprite_Type2, and choose “hammer.png” in Picture List and click “Add”.

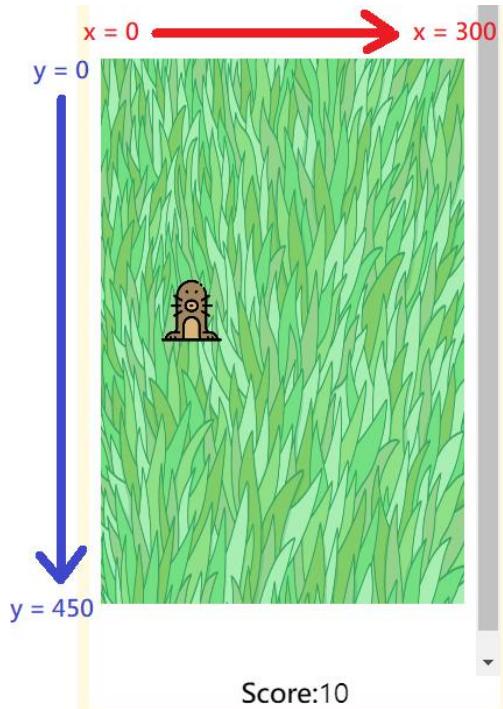


6. Click “Blocks” and start programming, so that:

- When the canvas is loaded, the location of Sprite1 (the mole) is repeatedly updated (i.e. “forever”) in every 2 to 5 seconds randomly.
- When the Sprite1 (the mole) is clicked:
 - (a) the location of Sprite1 (the mole) is updated randomly.
 - (b) The text in label2 (score) is increased by 1



7. Test the program.

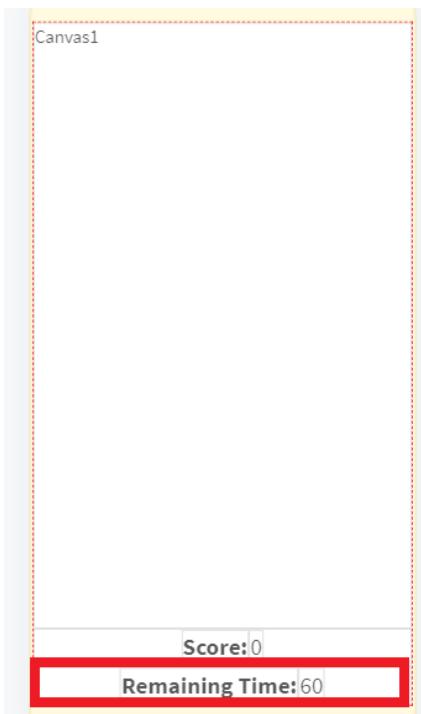


■ Wrack a mole (打地鼠) (2)

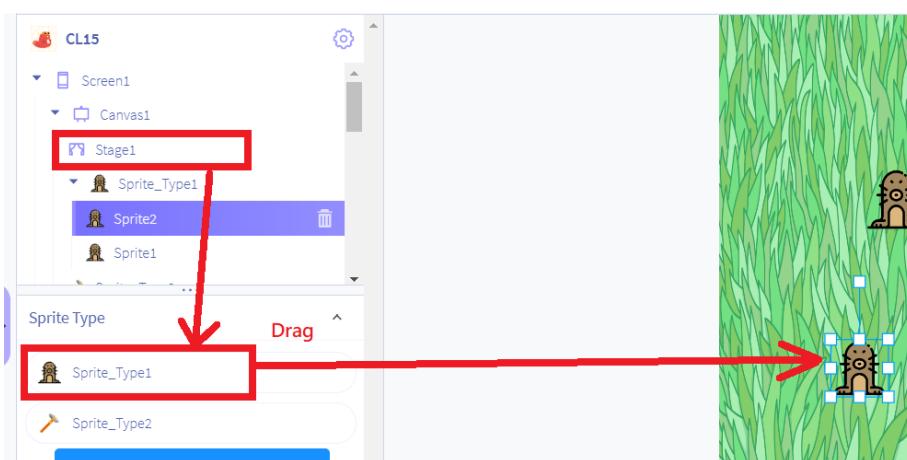
In this example, we will add more features, including:

- (a) Count the time (e.g. 60 seconds).
- (b) Add one more sprite (mole)
- (c) When mouse click, there will be a hammer.

1. Modify the project “CL15”.
2. Modify the interface by adding some components.



3. Click “Stage1”, and Drag “Sprite_Type1” to the interface. It will become Sprite2, i.e. the second mole.



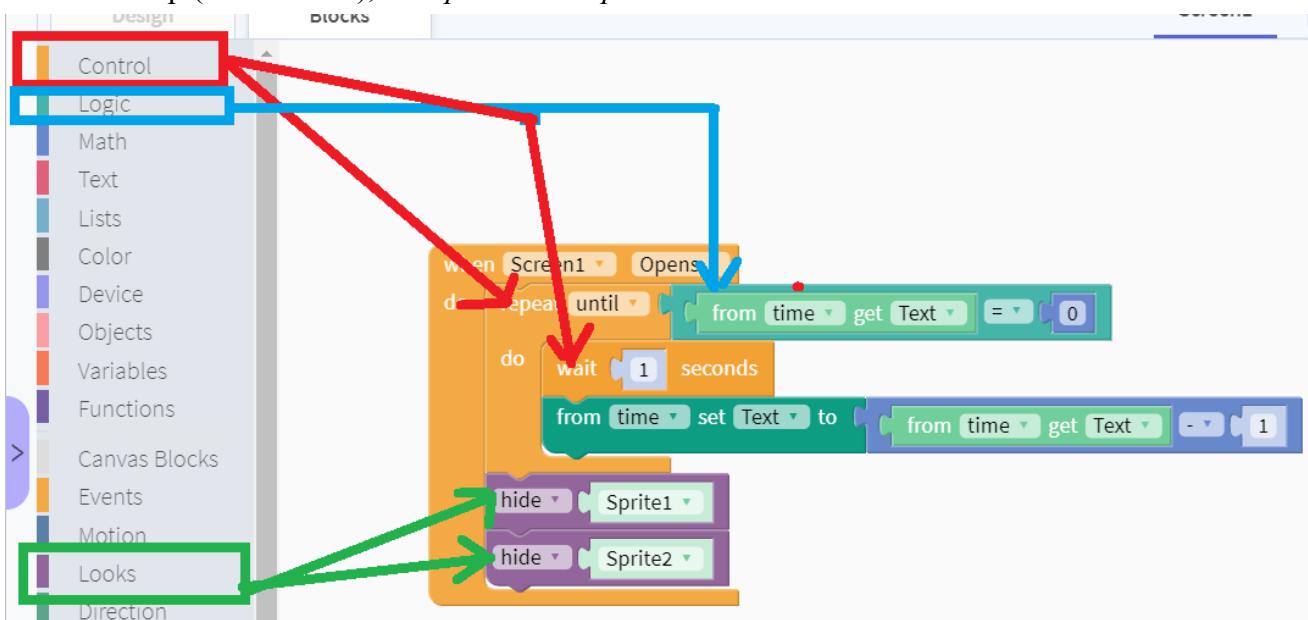
4. Add the following blocks to move Sprite2, and make the event when Sprite2 is clicked.



5. For the time counting function, the following would repeat until time (text) = 0:

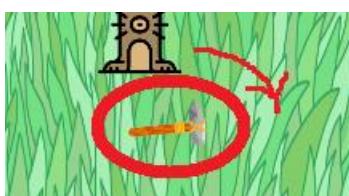
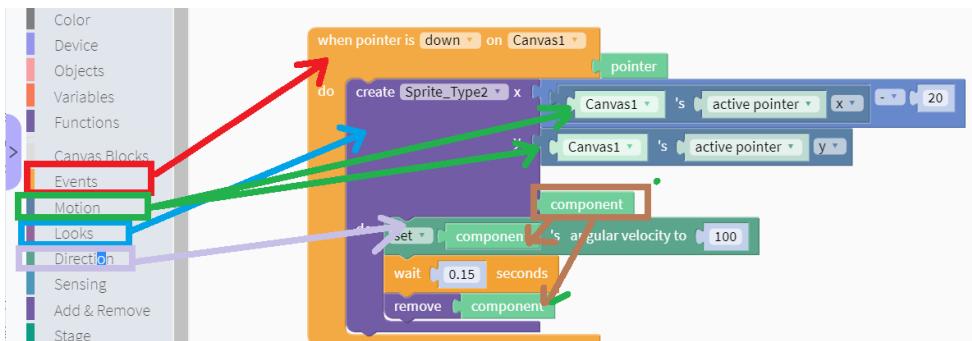
After 1 second, the time would decrease by 1.

After the loop (i.e. time = 0), the Sprite1 and Sprite2 would hide.



6. The last part would be about the hammer. When pointer (mouse) is clicked on Canvas1:

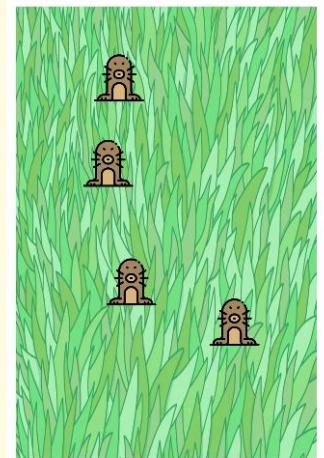
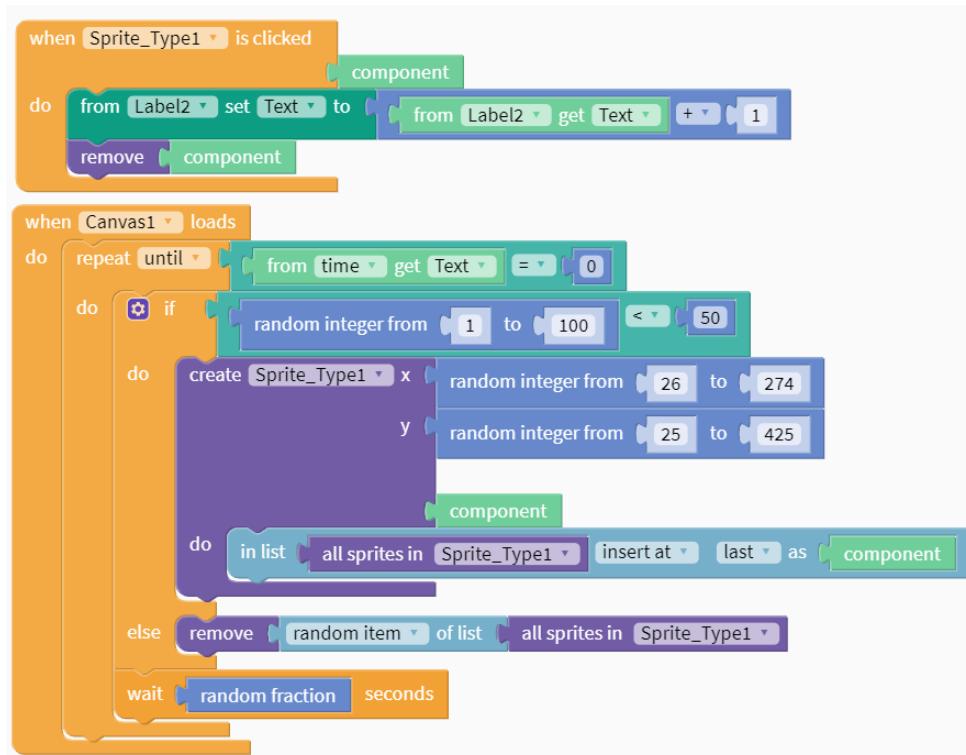
- Create *Sprite_Type2 (hammer)* at position of active pointer (mouse click), and the Sprite (hammer) becomes “component”
- the angular velocity of the hammer is set to be 100 (clockwise)
- after 0.15 seconds, the hammer is removed.



■ Wrack a mole (打地鼠) (3) ***

In this example, the mole appears and disappears randomly, and there is no limited number of moles.

In this example, there will be **no more “Sprite1” and “Sprite2”**. The Sprite will be created and removed from time to time randomly. The Sprite is stored in a **list called “all sprites in Sprite_Type1”**.



Thunkable (7): Integrated use of sensor, screen and canvas

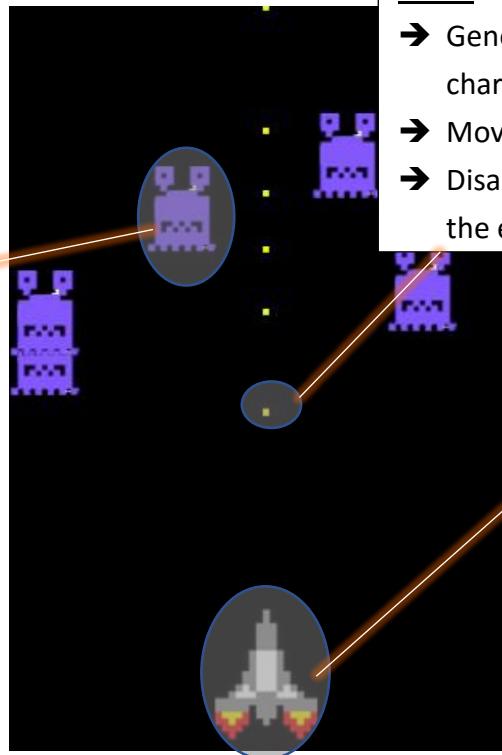
In this chapter, we will try to write some more exciting games, with the use of sensor and screen.

■ Shooting Game (1) (*requires a physical device to test*)

In this example, we would write an app that has a main character who is shooting bullet towards enemies.

Enemy

- ➔ Generate at the upper edge randomly
- ➔ Moving downwards
- ➔ Disappear at the lower edge, or hit by bullet



Bullet

- ➔ Generate at the tip of main character when the screen is clicked
- ➔ Moving upwards
- ➔ Disappear at the upper edge, or hit the enemy

Main Character

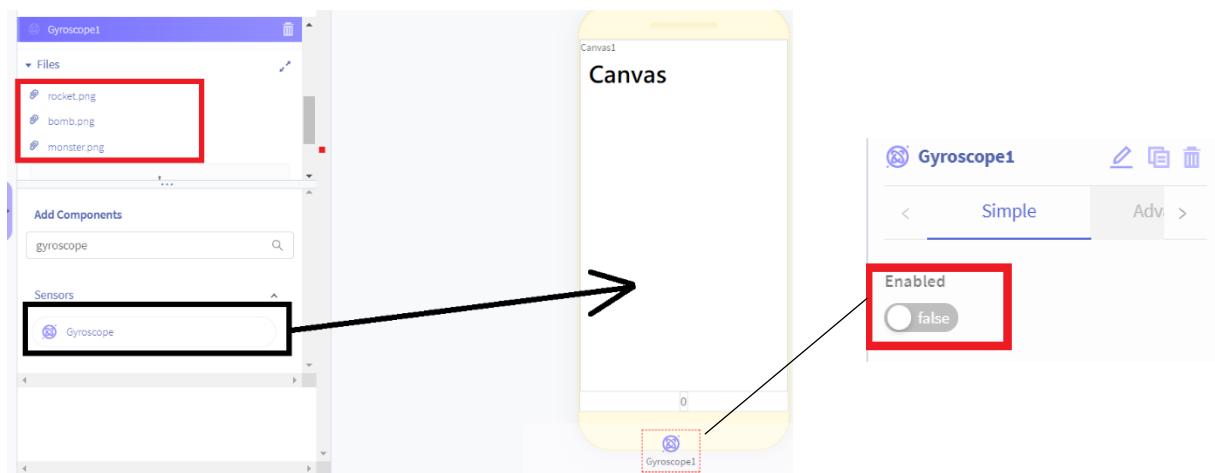
- ➔ When the phone is tilted, it moves to the left or right
- ➔ When the screen is clicked, it shoots a bullet

1. Create a new project “CL16”.

2. Create the following interface by dragging **Canvas**, **Gyroscope** and Label

In Gyroscope, turn the Enabled in Gyroscope to “false”.

Also, upload the 3 files (rocket.png, bomb.png and monster.png).

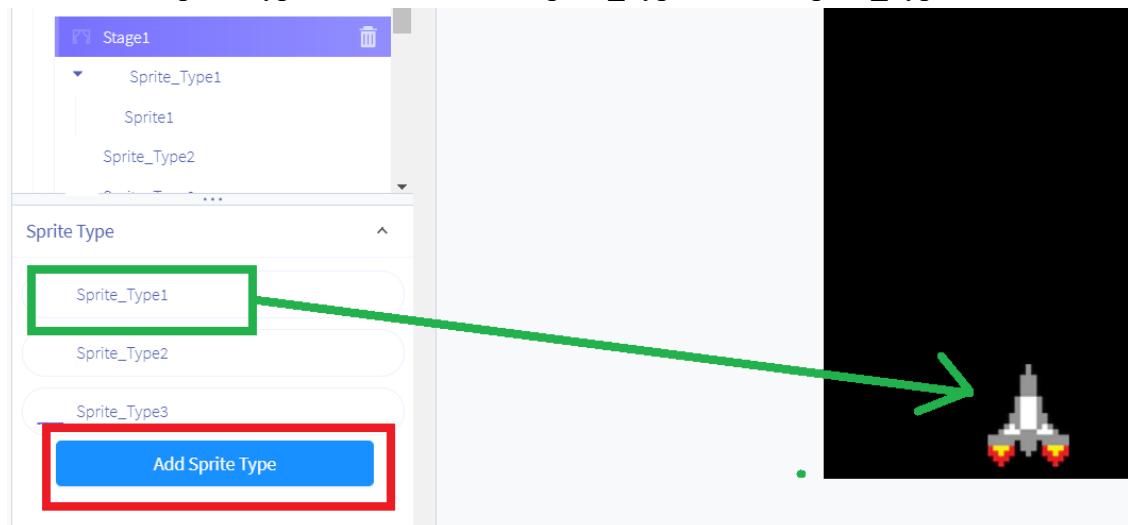


3. In Stage1, Change Background Color to black (#000000).



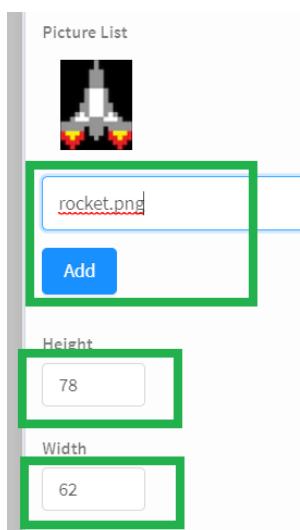
4. Drag Sprite_Type1 to the Screen to add “Sprite 1”.

Click “Add Sprite Type” so that it adds “Sprire_Type2” and “Sprite_Type3”.

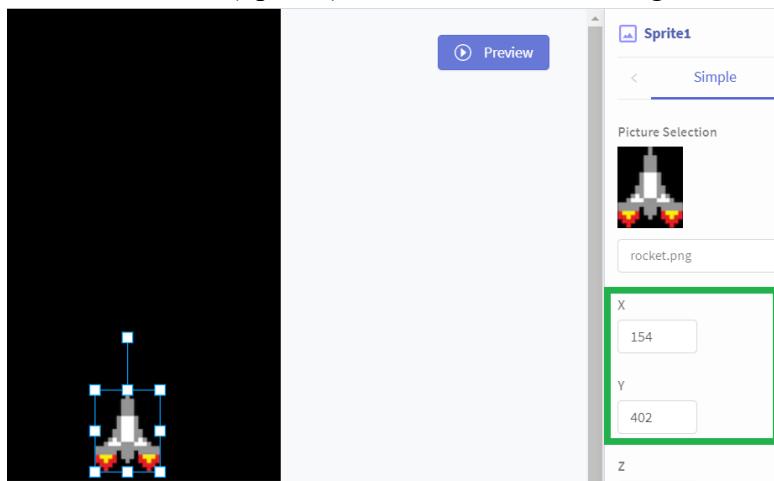


5. Change the Picture List, Height and Width of Sprite_Type1, Sprite_Type2 and Sprite_Type3.
Remember to click “Add” under Picture List.

	Sprite_Type1	Sprite_Type2	Sprite_Type3
Picture List	Rocket.png	Bomb.png	Monster.png
Height	78	4	51
Width	62	4	42

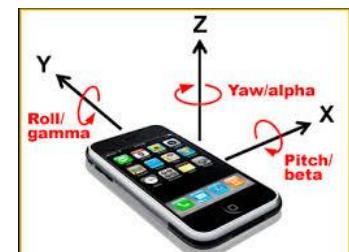
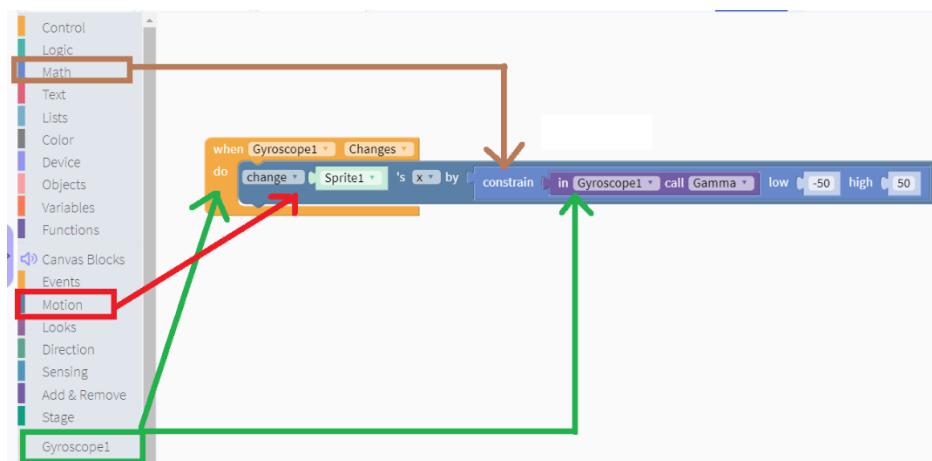


6. Select the rocket (Sprite1) on the screen, and change its X and Y into 154 and 402 respectively.

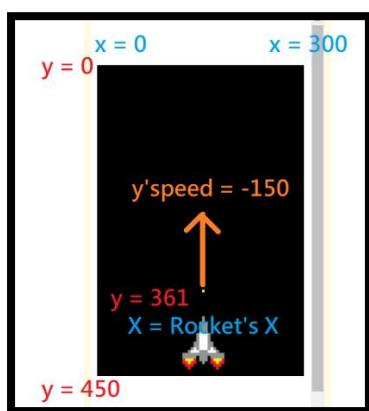
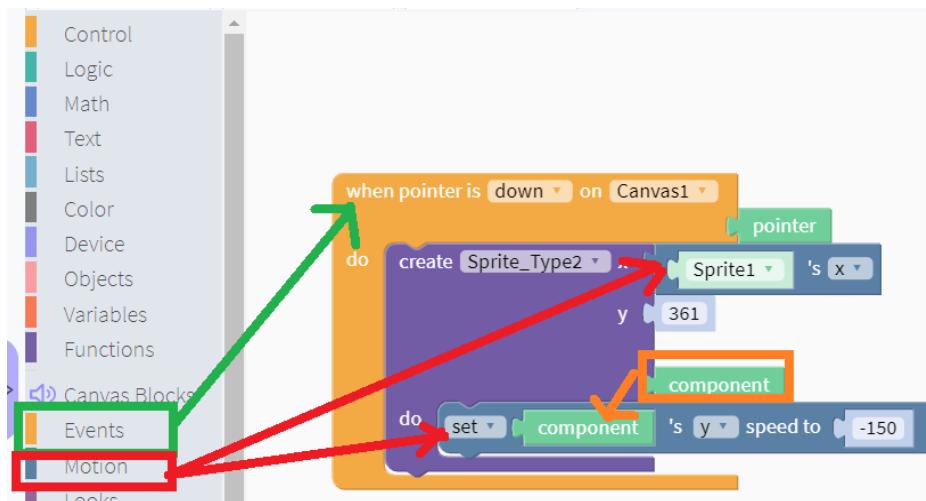


7. Click “Blocks” to start programming:

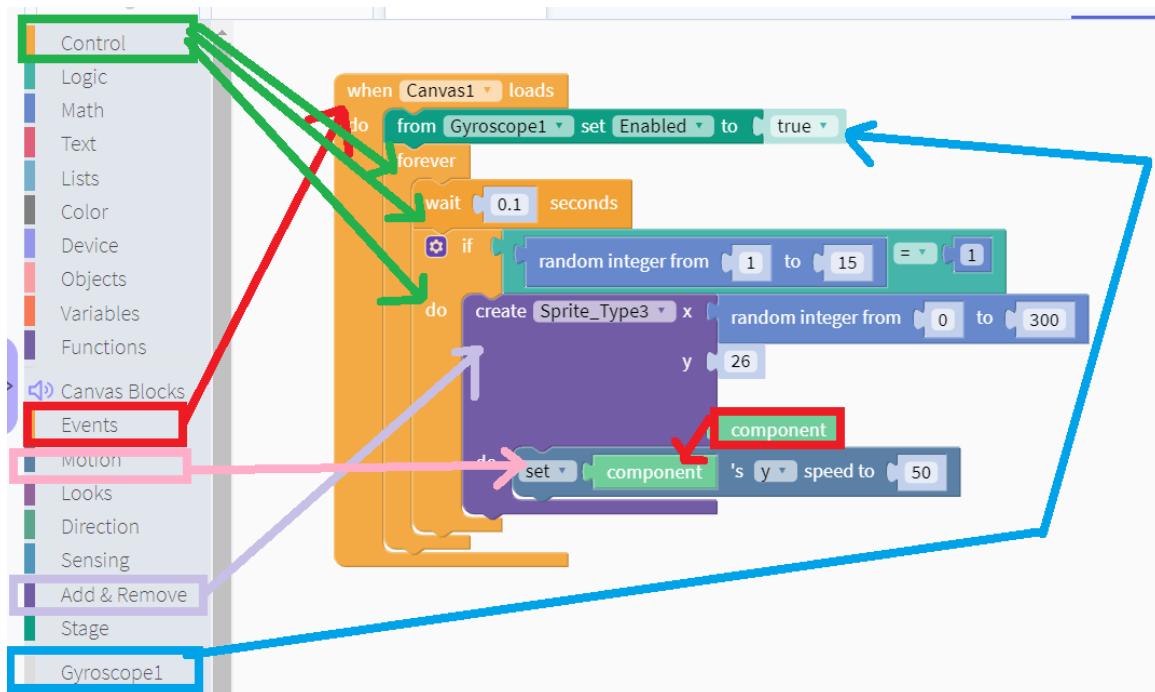
When Gyroscope (陀螺儀) detects the phone is tilted, “X” of Spritel (rocket) is changed according to the gamma reading from Gyroscope. To avoid changing too much, it sets a constrain from -50 to 50.



8. When the mouse cursor (pointer) is down, create Sprite_Type2 (bullet) at (Rocket's X, 361) and set its Y's speed to -150.

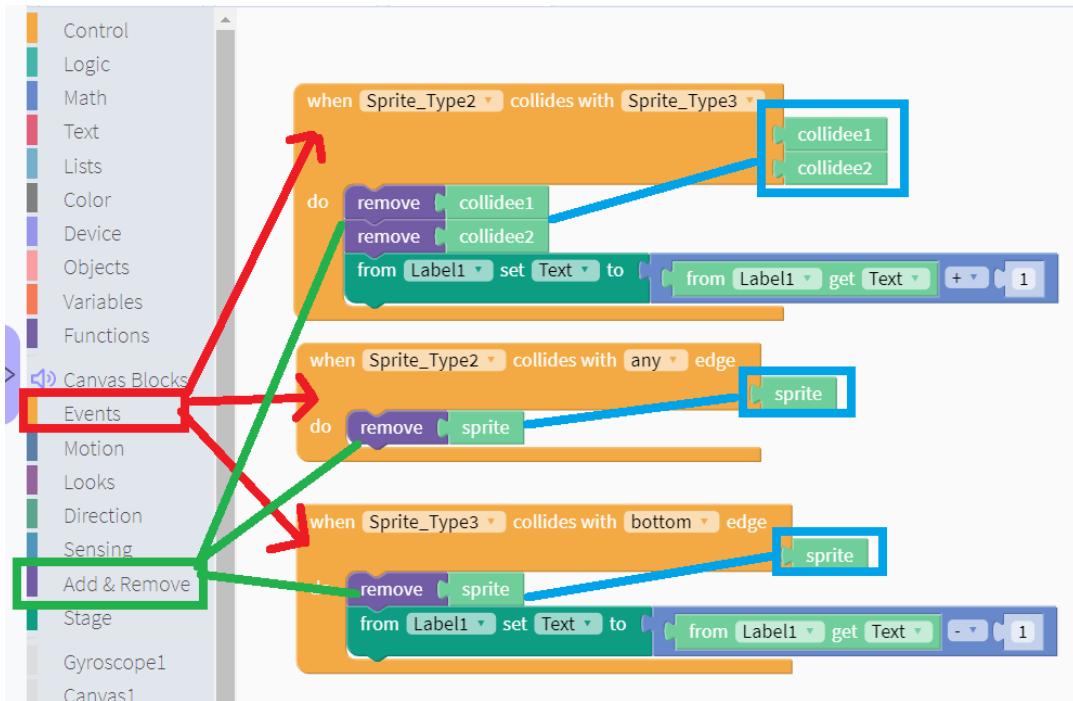


9. It enables Gyroscope1 first, then for every 0.1 second, it MAY create a monster (Sprite_Type3) (in 1/15 chance) at upper edge, and its Y's speed is set to 50.

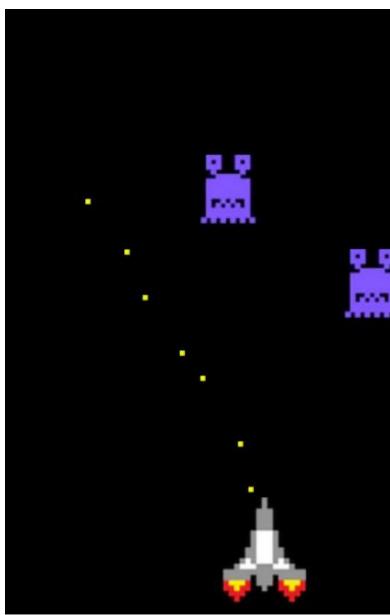


10. Set 3 more collision events:

- When bullet (Sprite_Type2) collides with monster (Sprite_Type3), they both disappear, and the label text changes by 1 (score increases by 1).
- When bullet (Sprite_Type2) collides with edge of screen, the bullet disappears
- When monster (Sprite_Type3) collides with bottom edge of screen, the monster disappears, and label text changes by -1 (score decreases by 1)



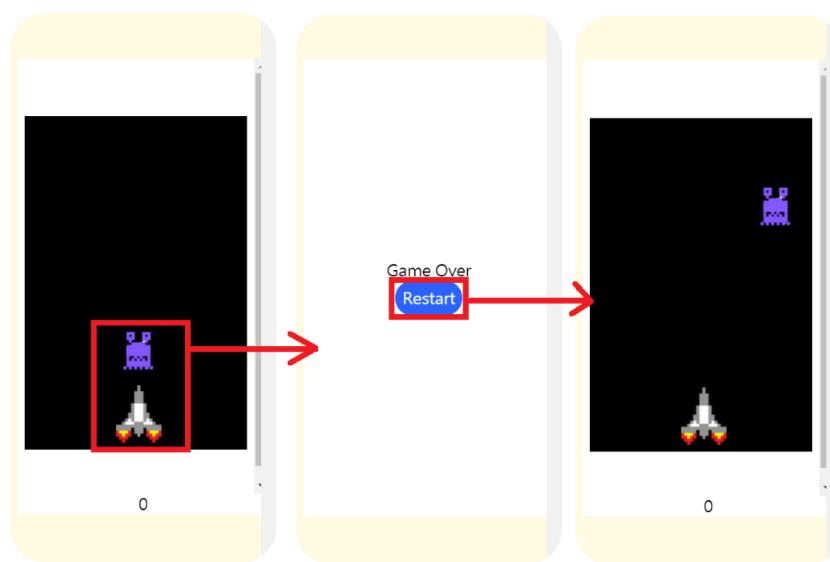
11. Test the program.



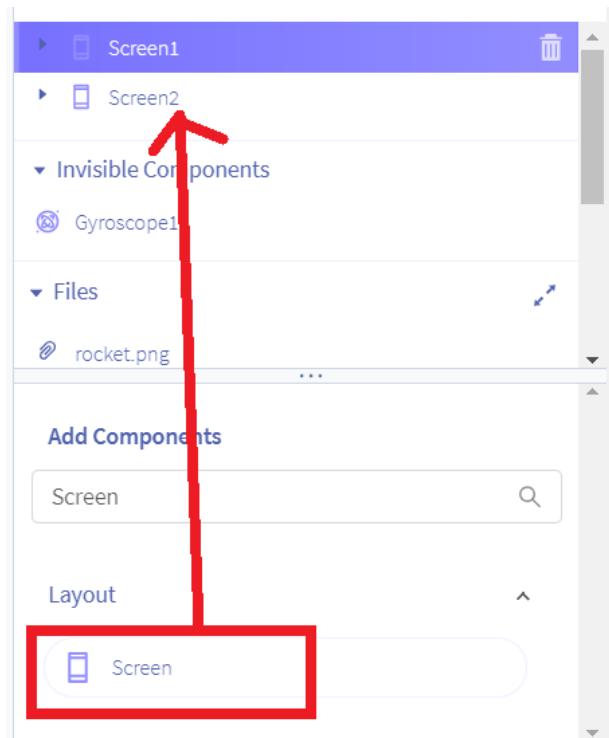
8

■ Shooting Game (2) (requires a physical device to test)

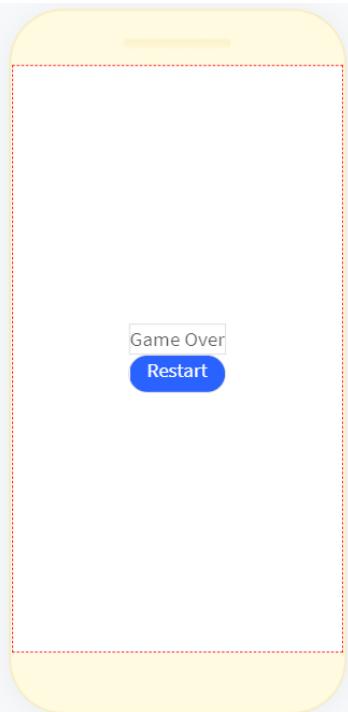
In this exercise, when the monster hits the rocket, it will change to another screen with a button called “Restart”. When “Restart” button is clicked, the game is restarted.



1. Open the project “CL16”
2. Drag a screen towards “Visible Components”, so that “Screen2” is added under “Screen1”.

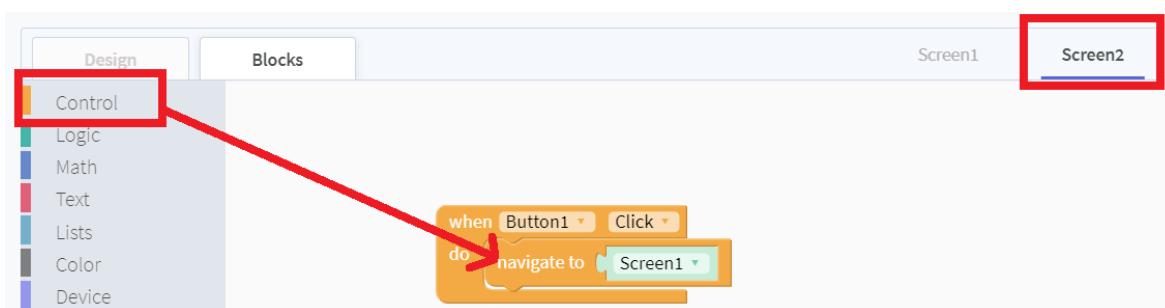


3. On Screen2, drag a button and a label on the screen. Remember to change the text on them.

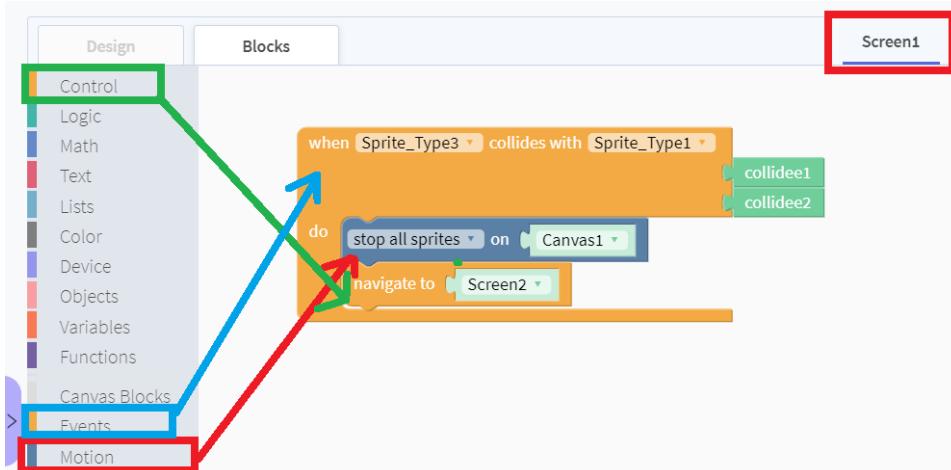


4. Select “Blocks” to start programming:

(On Screen2) When button is clicked, it will navigate to Screen1.



5. (On Screen1) When Sprite_Type3 (monster) collides with Sprite_Type1 (rocket), it will navigate to Screen2, and all elements (sprites) on this Canvas stops.

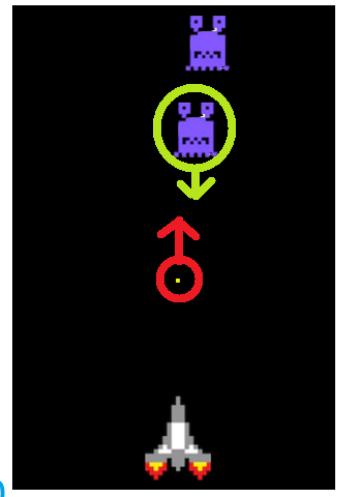


Thunkable (8): Physical Simulation in Game Design

In the previous chapters, we have discussed how we can design a game. In the previous games, the characters (sprites) are either at rest or moving at uniform (unchanged) speed.

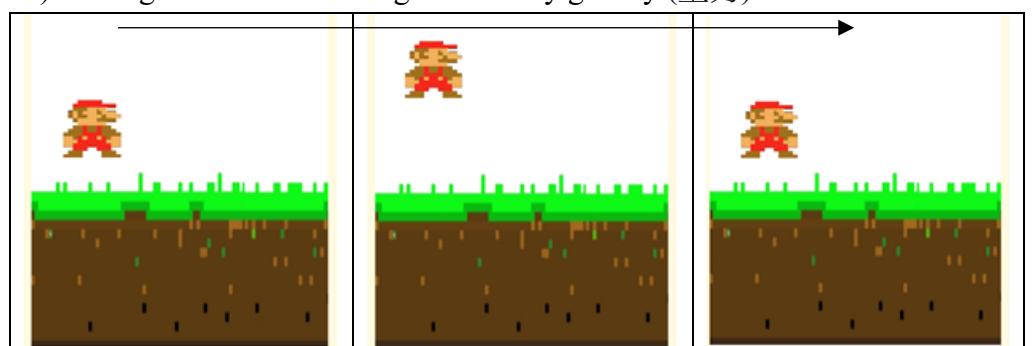
	Monster	Bullet
Direction of Y's speed (upward / downward)	downward	Upwards
Sign of Y's speed (+ / -)	+	-
Y's Gravity (change of speed / acceleration)	0	0

Y = 0



Y = 450

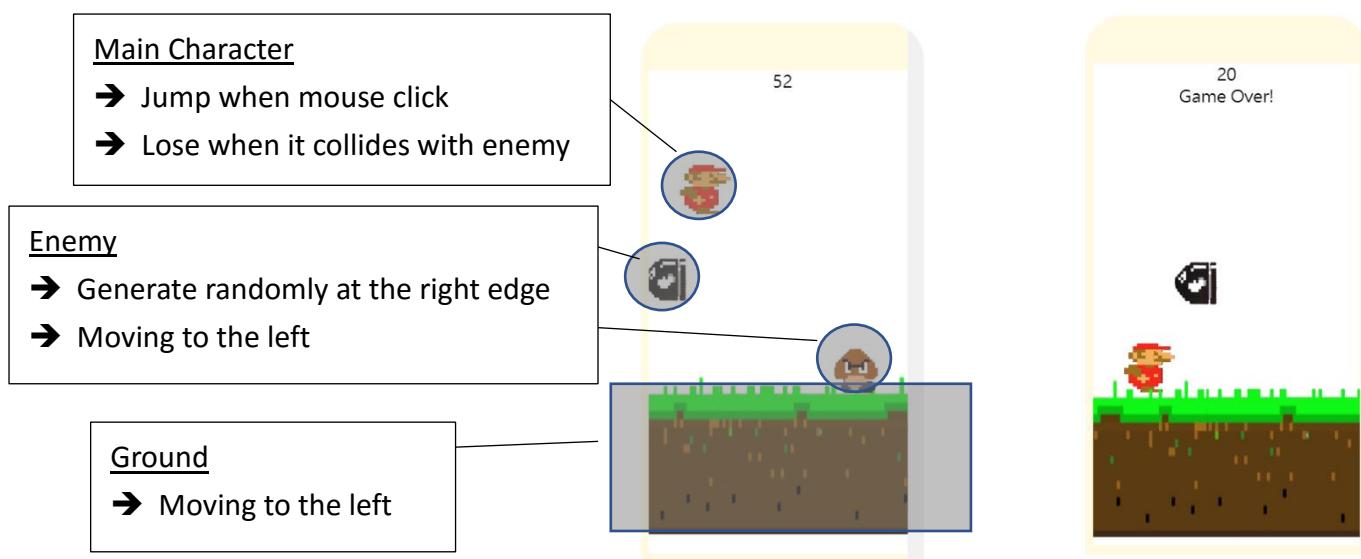
Sometimes, the character (sprite) in the game is like us being affected by gravity (重力).



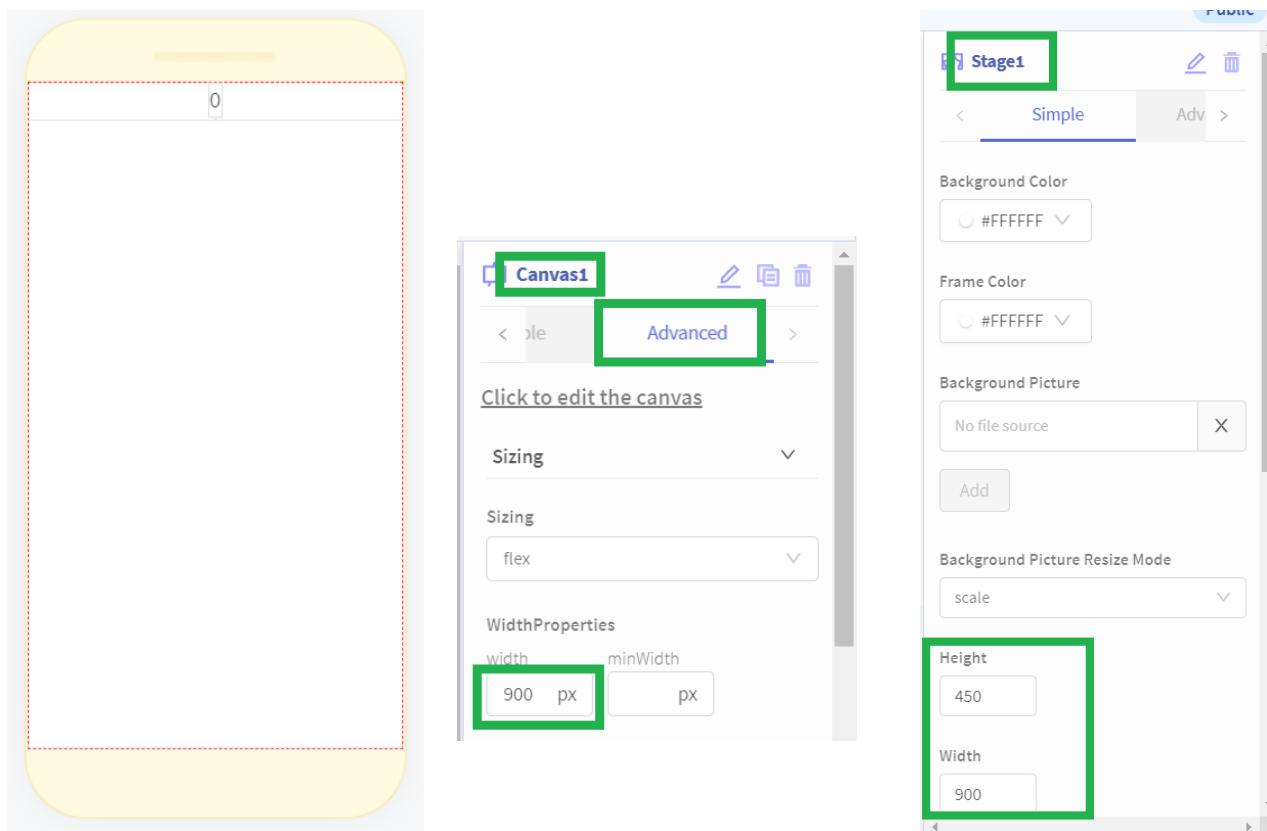
Direction of Y's speed (upward / downward)	upward	/	Downward
Sign of Y's speed (+ / -)	-	0	+
Y's Gravity (change of speed / acceleration)	Downward (+)		

■ Running with Obstacles

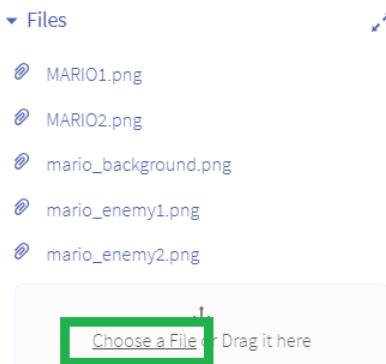
In this example, we would write an app that has a main character who is running and jumping over the enemies (obstacles).



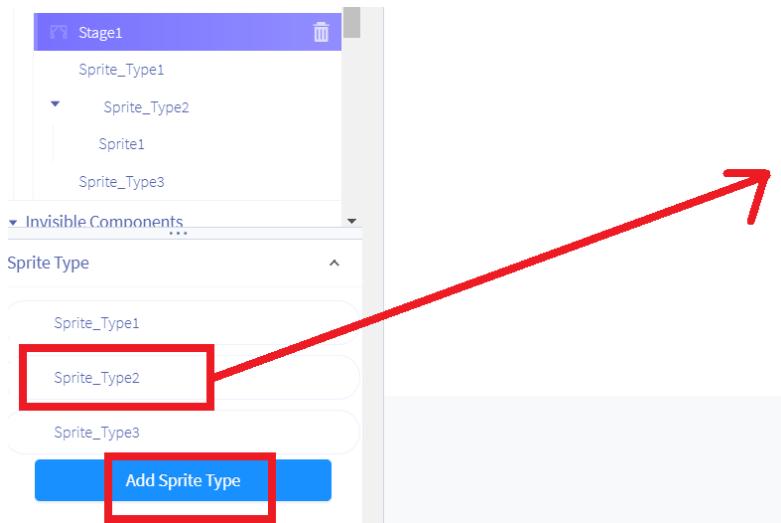
1. Create a new project “CL17”.
2. Create the following interface by dragging Canvas and 2 labels
Select “Canvas1” and “Advanced”, and change WidthProperties “width” to 900.
Select “Stage1” and “Simple”, and change its Height and Width to 450 and 900 respectively.



3. Upload the files into the app, including MARIO1.png, MARIO2.png, mario_background.png, mario_enemy1.png, mario_enemy2.png.

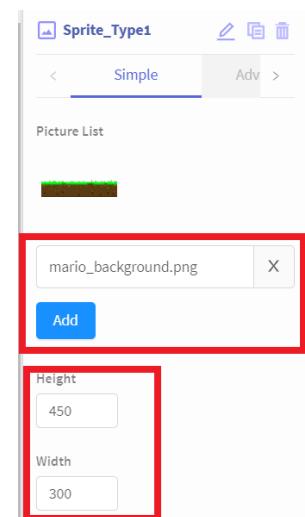


4. Click Stage1. Select “Add Sprite Type” so that there are “Sprite_Type2” and “Sprite_Type3”. Also, drag “Sprite_Type2” into the screen so that there is “Sprite1”.



5. Change the Picture List, Height and Width of Sprite_Type1, Sprite_Type2 and Sprite_Type3. Remember to click “Add” under Picture List.

	Sprite_Type1	Sprite_Type2	Sprite_Type3
Picture List	Mario_background.png	MARIO1.png MARIO2.png	Mario_enemy1.png Mario_enemy2.png
Height	450	57	50
Width	300	57	50
Ignore gravity	True	false	true

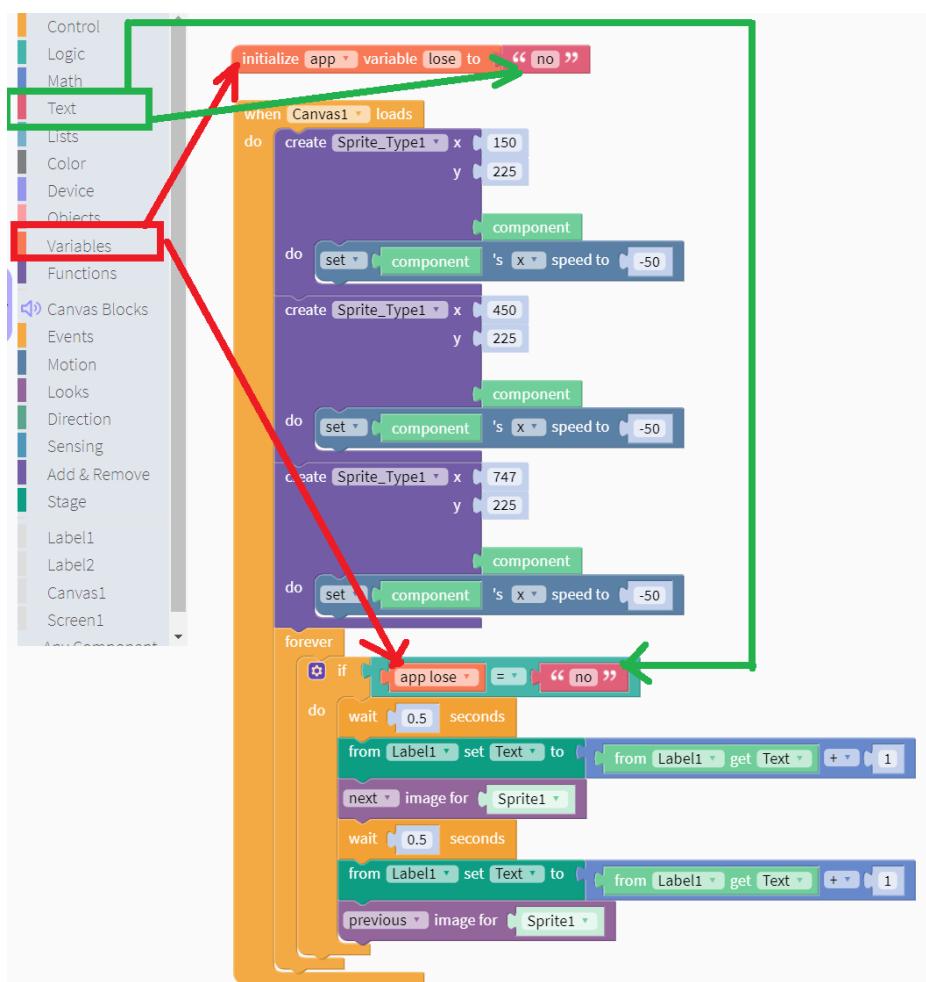


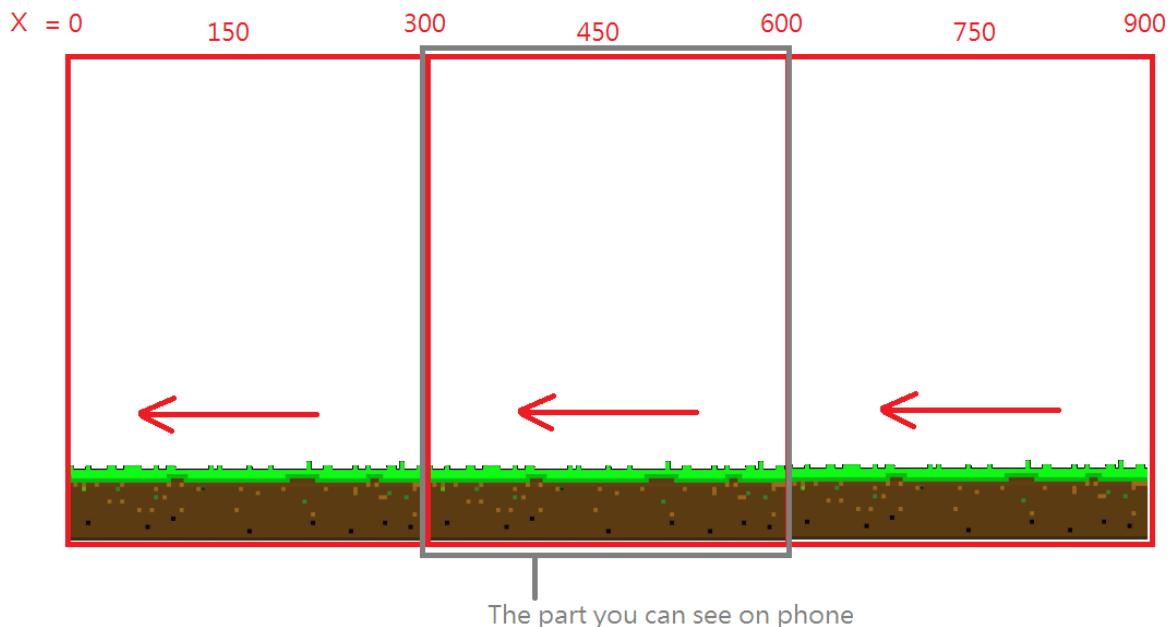
6. Select the main character (Sprite1) on the screen, and change its X and Y into 374 and 261 respectively.



7. Select “Blocks” to start programming.

- Initialize a variable lose as “no”
- At the beginning (when canvas1 loads), it creates 3 ground and set it moves to the left (x’s speed to -50)
- Then, if lose equals “no”, it will forever change to image of Sprite1 (main character), and add marks by 1 in every 0.5 seconds.





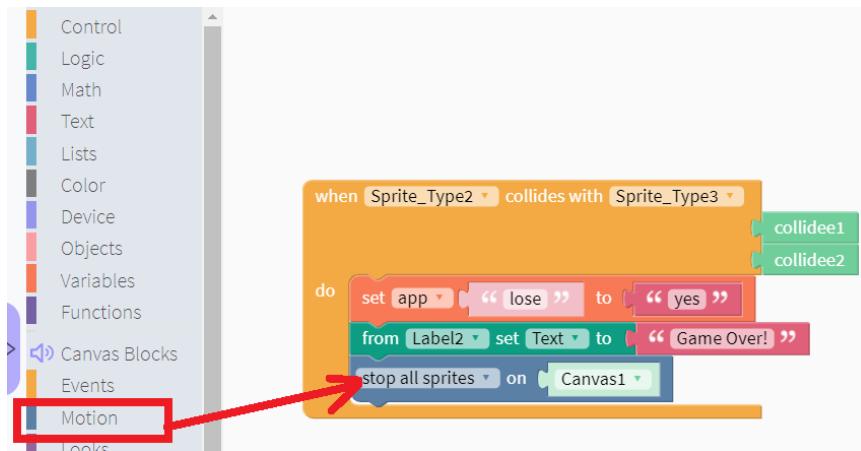
8. When mouse cursor (pointer) is down, if not yet lose (lose = "no"), the Y's speed of Sprite1 (main character) is changed to -200 (upwards), and the gravity Y is set to 130 (**as its speed is changing**). Then, when Y of Sprite 1 is less than 261 (i.e. main character reaches the ground), Y's speed of Sprite1 (main character) is changed to 0 (at rest), and its gravity Y is set to 0.

```

when pointer is down on Canvas1
  pointer
  do
    if app lose = "no"
      do
        set Sprite1 's y speed to -200
        from Stage1 set Gravity Y to 130
      end
      wait 1 seconds
      repeat while Sprite1 's y < 261
        do
          set Sprite1 's y speed to 0
          from Stage1 set Gravity Y to 0
        end
      end
    end
  end
end

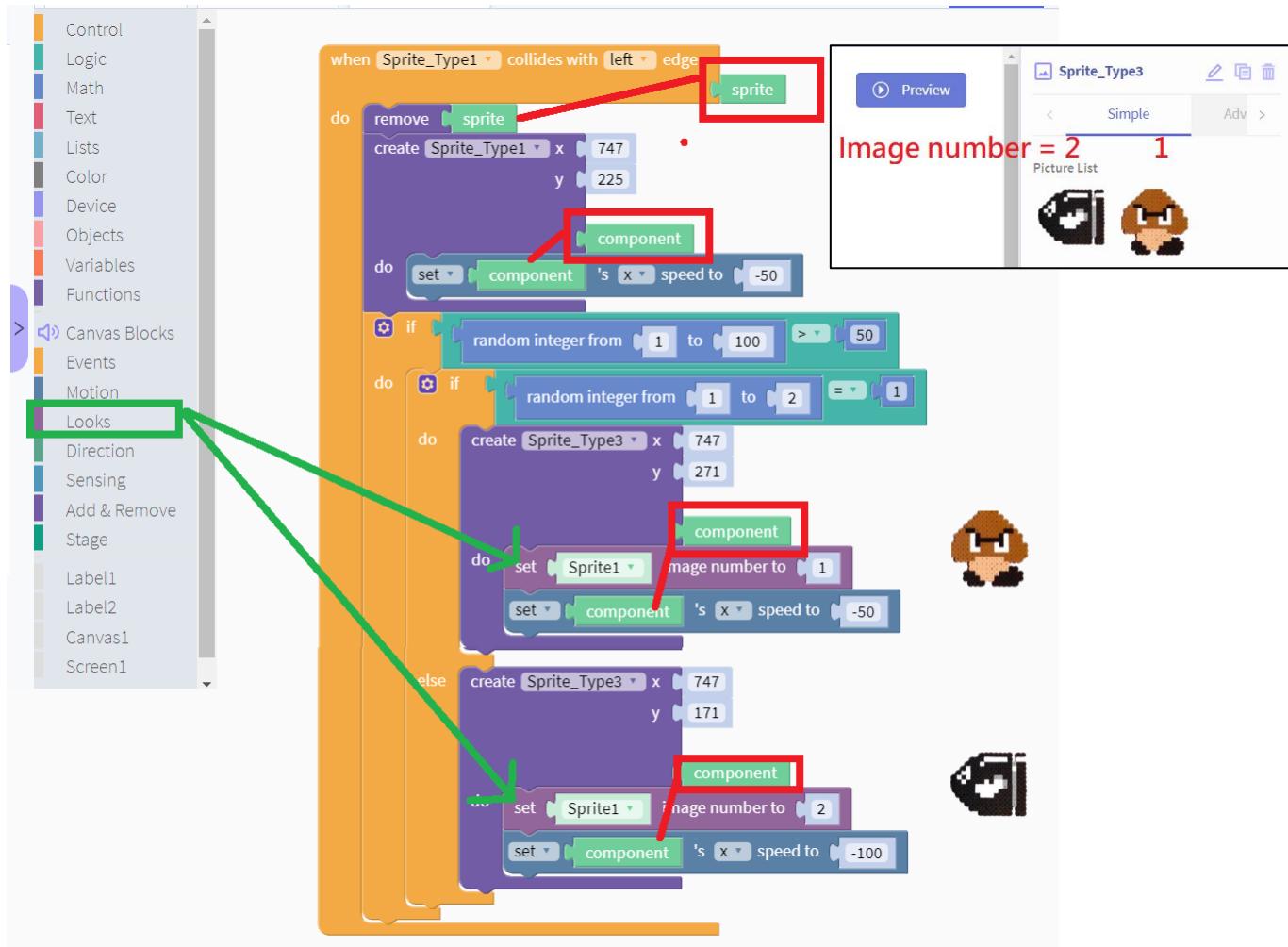
```

9. When Sprite_Type2 (main character) collides with Sprite_Type3 (enemy), lose is changed to “yes”, and the label2 changed to “Game Over!”, then stop every sprite.



10. When a ground (Sprite_Type1) collides with left edge, it would remove the ground, and create one more ground at the right.

Then it create enemy (Sprite_Type3) randomly. There are two different enemies, with different image, speed and Y.



Thunkable (9): JSON:API for Weather and Finance Data, and Layout Design

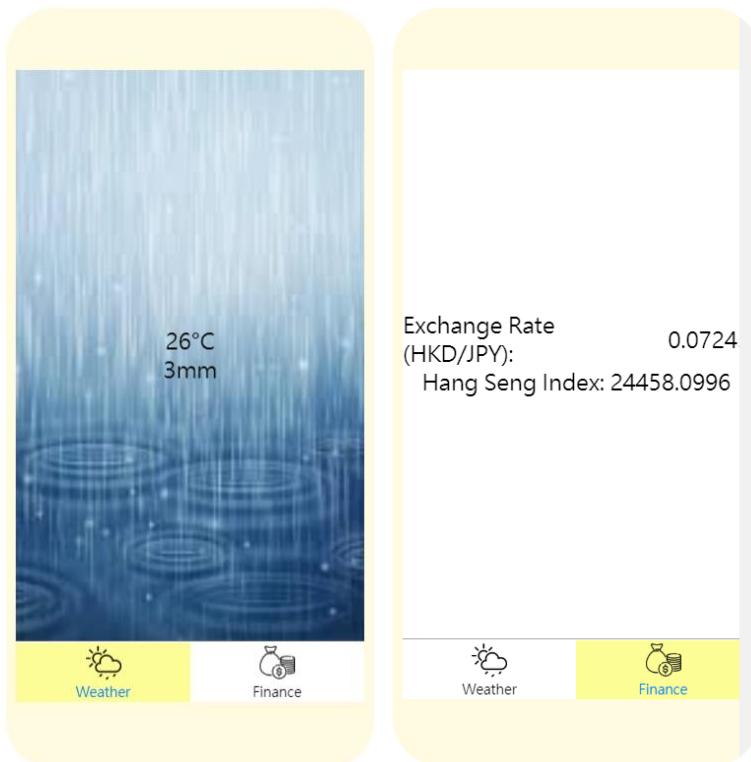
In many applications, external data would be fetched in order to perform certain functions:

- Fetching weather information, and remind users to bring umbrella
- Fetching exchange rate (匯率) information, and remind users to buy foreign currencies (外幣)
- Fetching data of stock (股票) price, and remind users to buy or sell stock.

■ Weather and Finance App

In this example, we would write an app that contains 2 tabs (screens), namely Weather and Finance.

- In Weather, it would display temperature and rainfall, and the background would change based on rainfall.
- In Finance, it would display exchange rate (HKD/JPY) and Hang Seng Index (恆生指數).

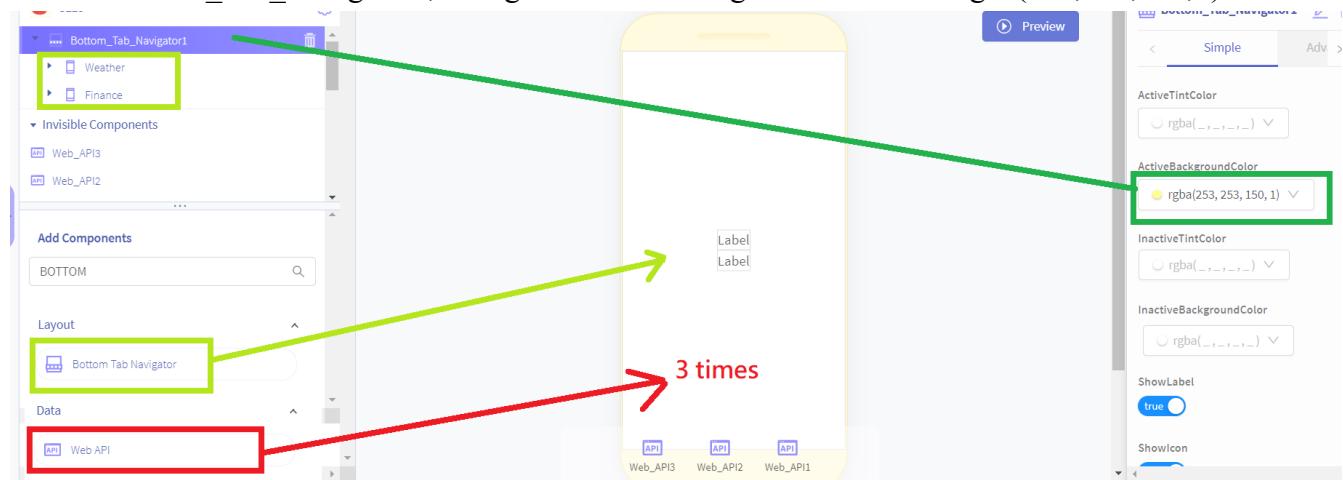


1. Create a new project “CL18”.

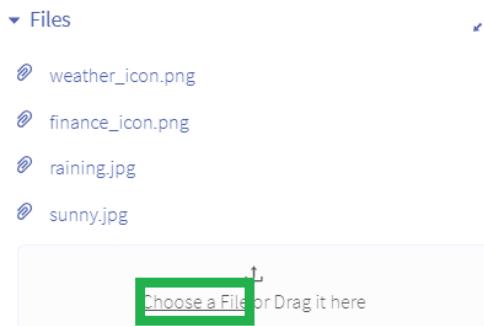
2. Create the following interface by dragging **Bottom Tab Navigator** and **3 Web API**.

Change the name of Screen into “Weather” and “Finance”, and remove unnecessary screens.

Select “Bottom_Tab_Navigator”, change its ActiveBackgroundColor into rgba(253,253,150,1).

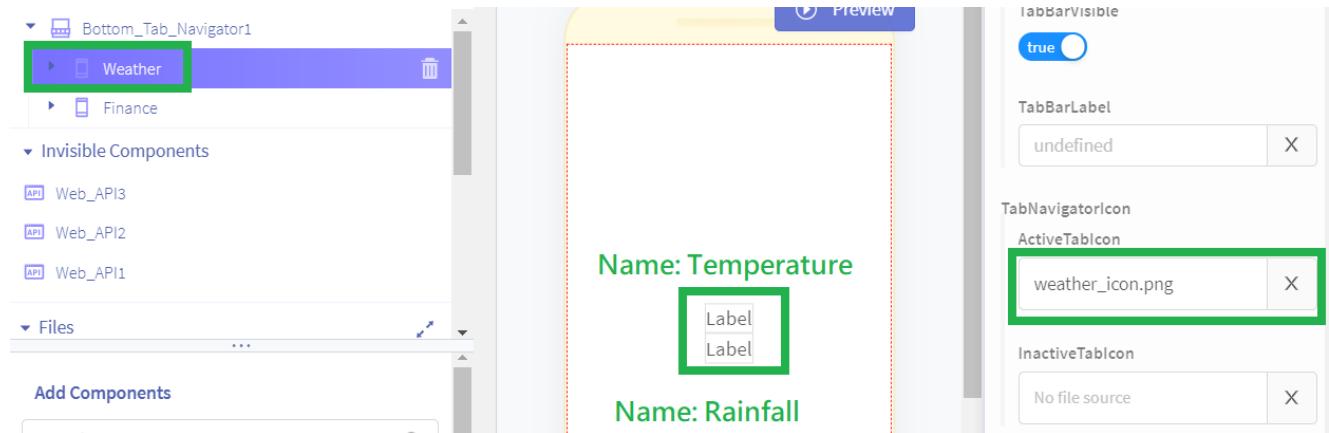


3. Upload the files into the app, including weather_icon.png, finance_icon.png, raining.jpg and sunny.jpg.



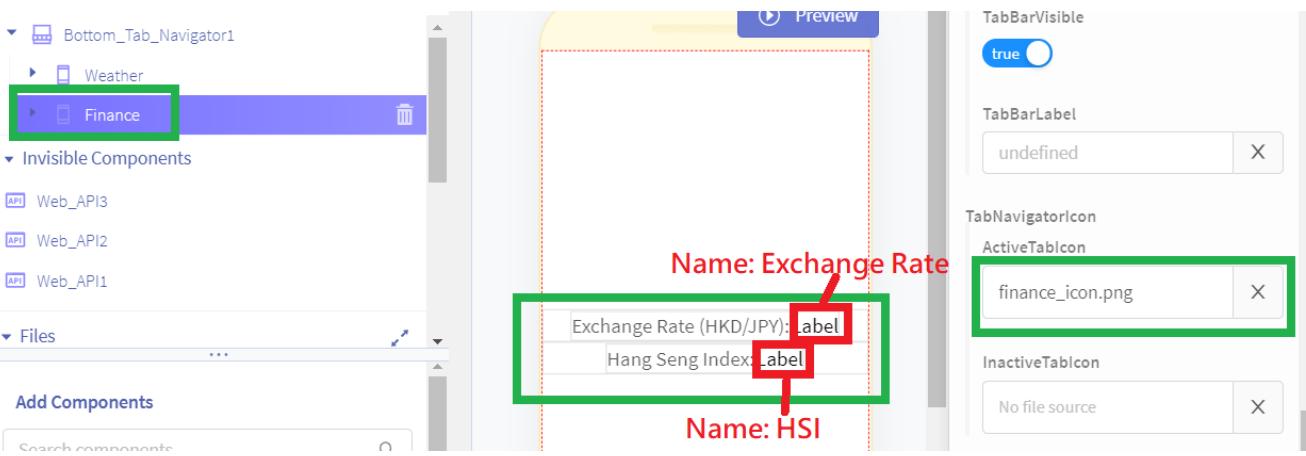
4. Select “Weather”, drag two labels on the screen. Change their names into “Temperature” and “Rainfall” respectively.

Change its ActiveTabIcon into “weather_icon.png”.



5. Select “Finance”, drag 4 labels and 2 rows on the screen. Change name of the two labels into “Exchange Rate” and “HSI” respectively.

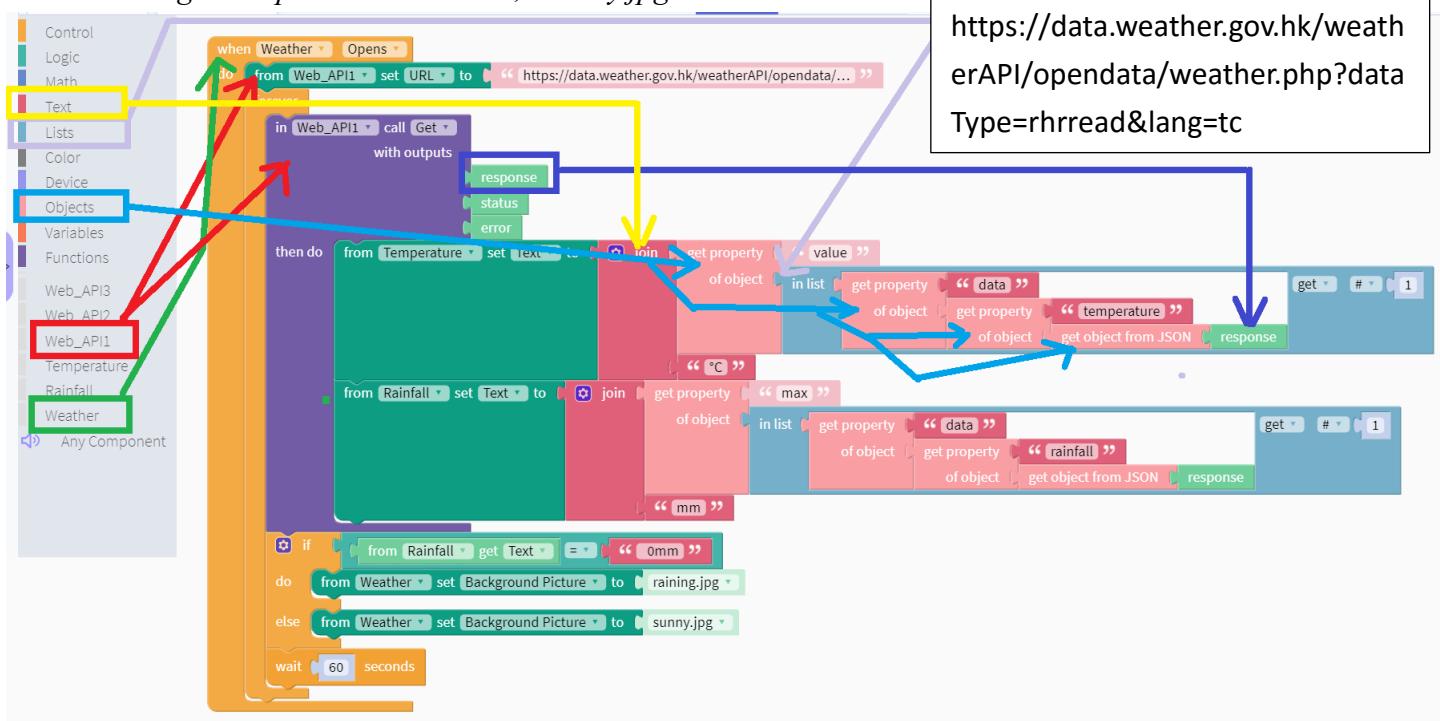
Change its ActiveTabIcon into “finance_icon.png”.



6. Select “Weather” and then “Blocks” to start programming.

Use Web_API1 to access API from Hong Kong Observatory.

If the rainfall is not “ 0mm ” (there are 2 leading spaces), “raining.jpg” would be set as the background picture. Otherwise, “sunny.jpg” would be set.



In the API,

```
{"rainfall": {"data": [{"unit": "mm", "place": "中西區", "max": 0, "main": "FALSE"}, {"unit": "mm", "place": "東區", "max": 0, "main": "FALSE"}, {"unit": "mm", "place": "葵青", "max": 0, "main": "FALSE"}, {"unit": "mm", "place": "離島區", "max": 2, "main": "FALSE", "min": 0}, {"unit": "mm", "place": "北區", "max": 2, "main": "FALSE", "min": 0}, {"unit": "mm", "place": "西貢", "max": 0, "main": "FALSE"}, {"⋮}, {"⋮}, {"⋮}]};
```

List #1

List #2

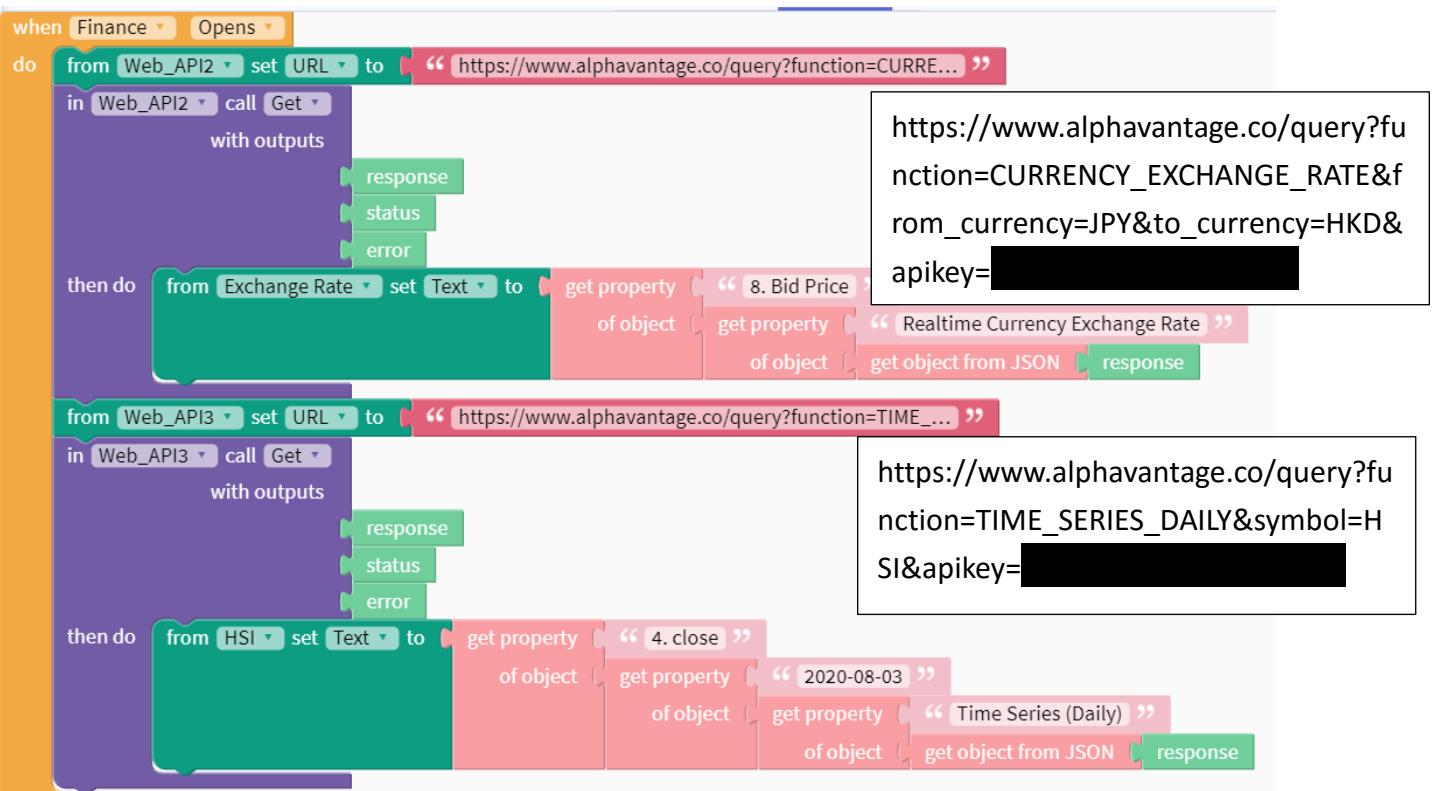
List #3

Therefore, in the program above, it is fetching the rainfall from Central & Western district (中西區). Do you know how we can fetch the rainfall data from Kwai Tsing District?

7. Select “Finance” Tab.



8. (optional) Use Web_API2 and Web_API3 to access API from Alpha Vantage for Exchange Rate and Hang Seng Index respectively.

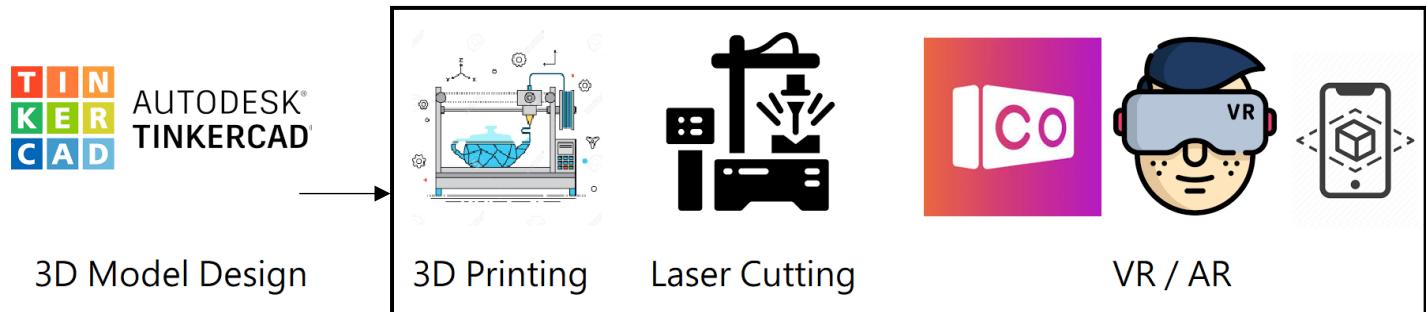


9. Test the program.

10

TinkerCAD (1): Basic 3D Design

In this chapter, we will learn how to build our 3D model.



The 3D models built could have a lot of applications, such as:

- For 3D printing
- For laser cutting
- For virtual reality (_____): a simulated experience that is fully immersive. User would wear a VR headset and see the VR simulation.
- For augmented reality (_____): overlay of digital content onto the physical world. User would usually look through the camera of a mobile device (e.g. smartphone) and see the digital content mixed with the real one.

Before that, we have to learn how to build a 3D model. One of the most popular platform is TinkerCAD.

1. Registration of an account

- (a) Go to [tinkercad.com](https://www.tinkercad.com)
- (b) Click “Sign up now”



(c) Select “set up individual account”, then “use email address to register”.

開始編輯

您將如何使用 Tinkercad ?

在學校 ?

教師從這裡開始

學生，加入課程

您自己

建立個人帳戶

開始編輯

您將如何建立帳戶 ?

已擁有帳戶 ?
登入

使用電子郵件註冊

The screenshot shows the initial steps of creating a Tinkercad account. It asks if you're at school (with 'Teacher' and 'Student' buttons) or yourself (with a 'Create Account' button). It also provides links to start editing ('Edit Now') or log in ('Log In'). A red box highlights the 'Create Account' button under 'Yourself'.

(d) Fill in the location (as “Hong Kong”) and your date of birth. Then, click “Next”.

建立帳戶

國家、地區或區域

香港

生日

一月 9 1970

下一步

This screenshot shows the 'Create Account' form. It has fields for 'Country/Region' (set to Hong Kong) and 'Birth Date' (set to January 9, 1970). A red box highlights the 'Birth Date' field. Below the form is a link for existing users to log in.

(e) Enter email address, password, and tick the checkbox, then click “create an account”. Then click “finish”.

建立帳戶

電子郵件

xxxxxx@lamwoo.edu.hk

密碼

.....

我同意 Tinkercad 服務條款和 Autodesk 隱私權聲明。

建立帳戶

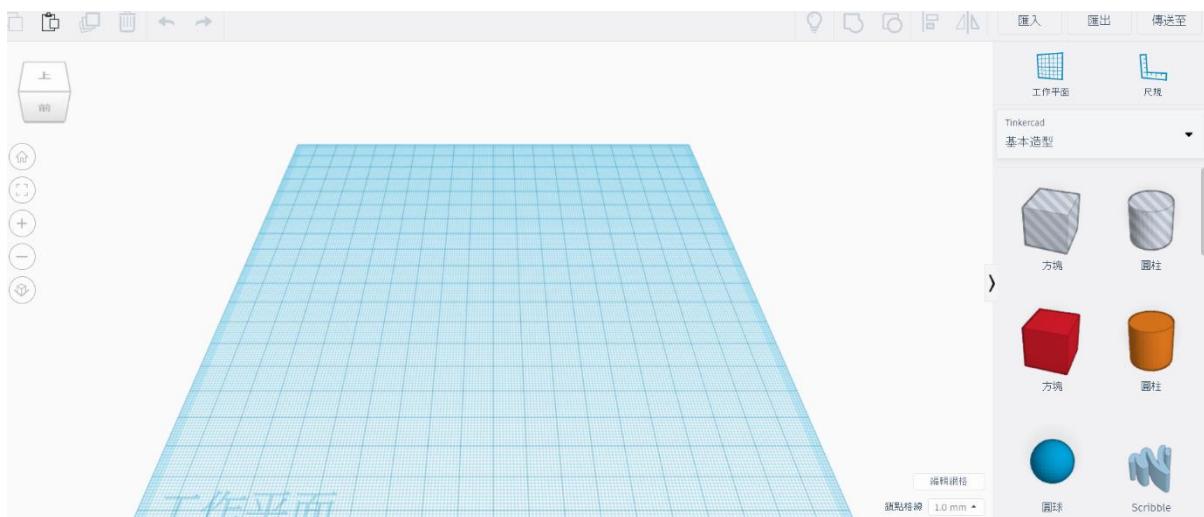
This screenshot shows the final step of the account creation process. It requires entering an email address, setting a password, accepting terms and conditions, and clicking 'Create Account'. A red box highlights the 'Create Account' button.

(f) Then, click the TinkerCAD icon at the top left corner. Select “Create a new design”.

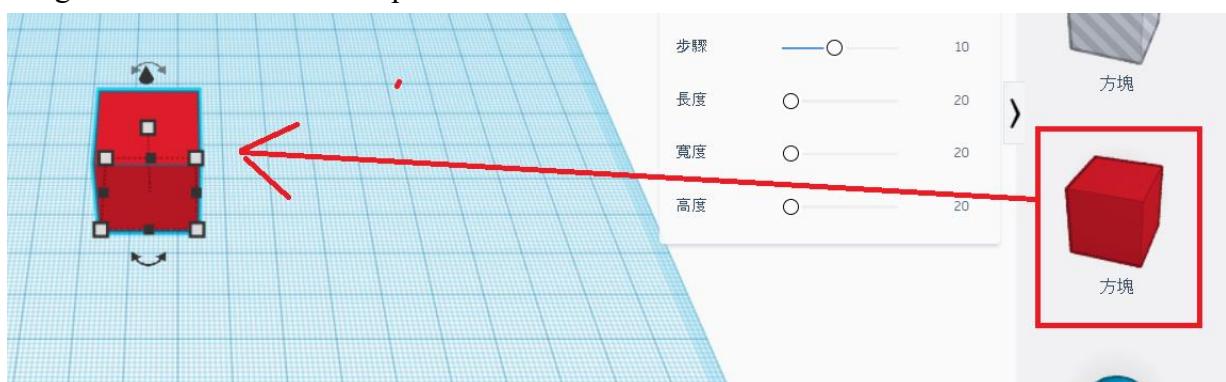


2. Control of camera

(a) You will see the user interface of TinkerCAD.



(b) Drag a cube towards the workplane.

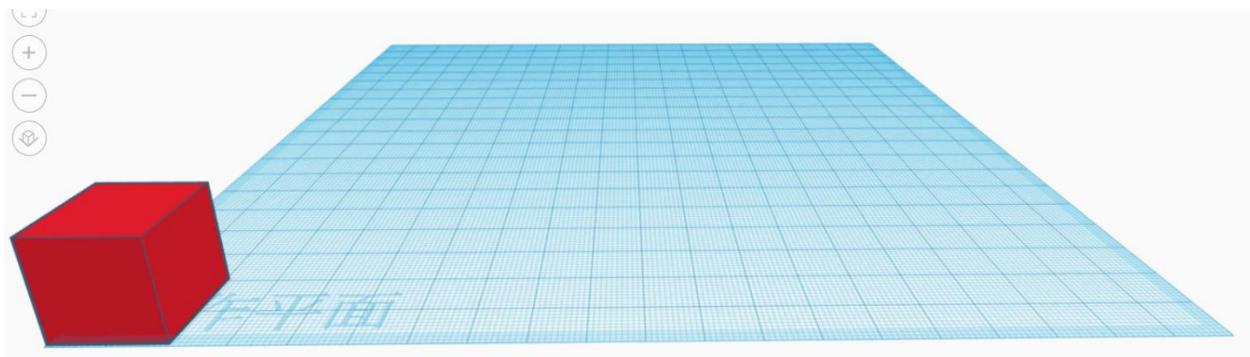


(c) Try to right-click and drag your mouse, to see what happens.

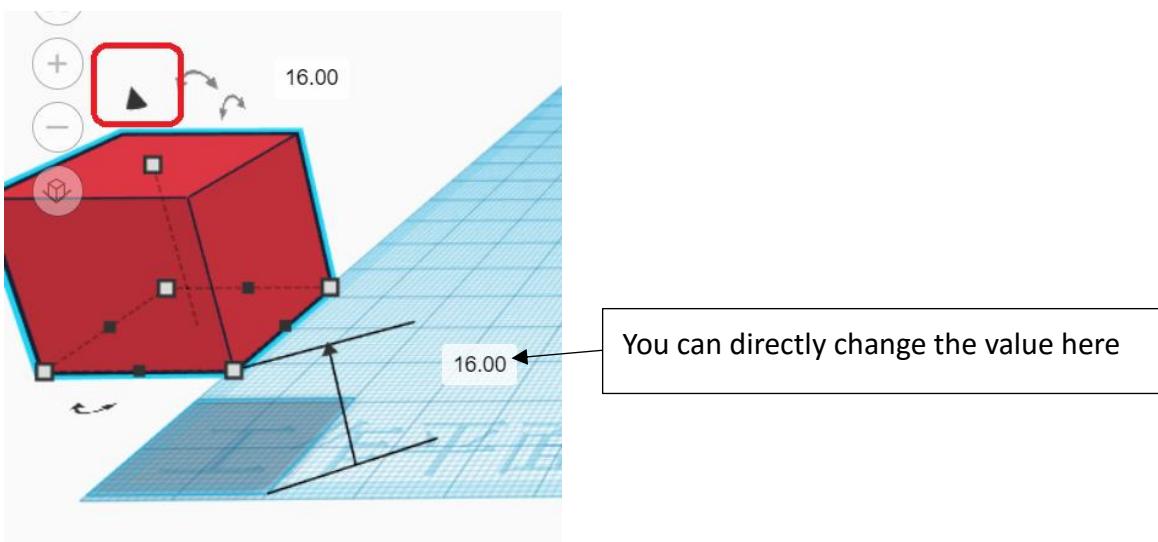
(d) Try to zoom in and out using the mouse wheel.

3. Transformation of an object

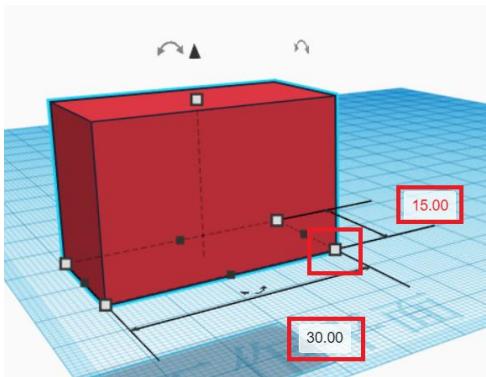
- (a) (Translation) Try to drag the cube and move it to the left bottom corner.



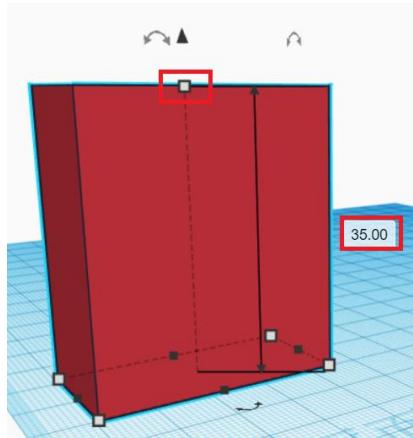
- (b) (Translation) Try to click the triangle icon above the cube, and move it away from the workplane by 16 units (i.e. 16mm).



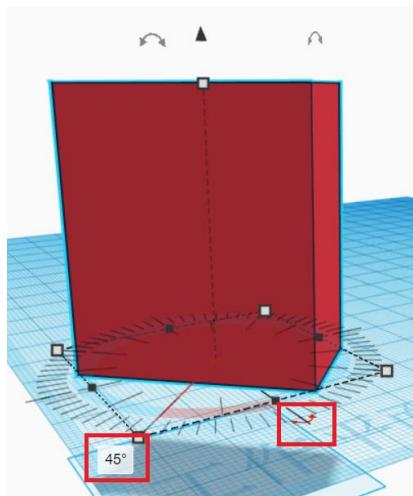
- (c) (Enlargement / Reduction) Drag the square icon at the bottom vertex of the cube, and change the size of the cube base into 30×15 .



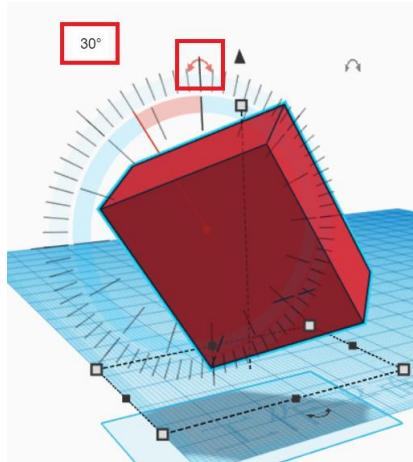
(d) (Enlargement / Reduction) Drag the square icon at the upper edge of the cube, and change the height of the cube into 35.



(e) (Rotation) Drag the two-way arrow at the bottom and rotate the cube by 45° .

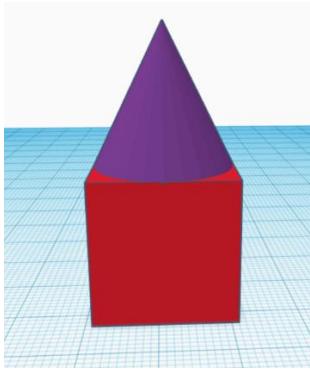


(f) (Rotation) Drag the two-way arrow at the top and rotate the cube by 30° .

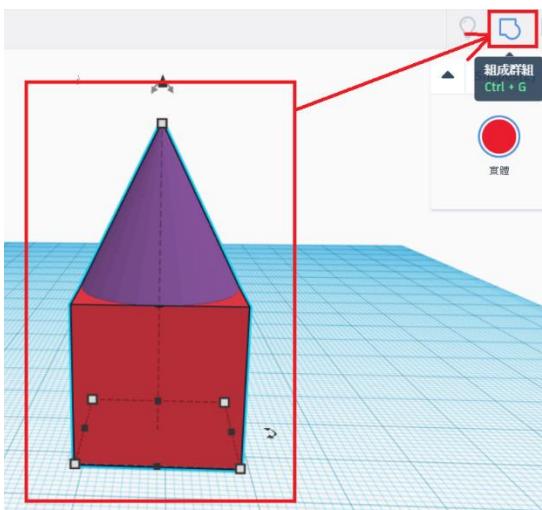


4. Grouping of objects

- (a) Create a cube, and a cone. The cone is put above the cube.



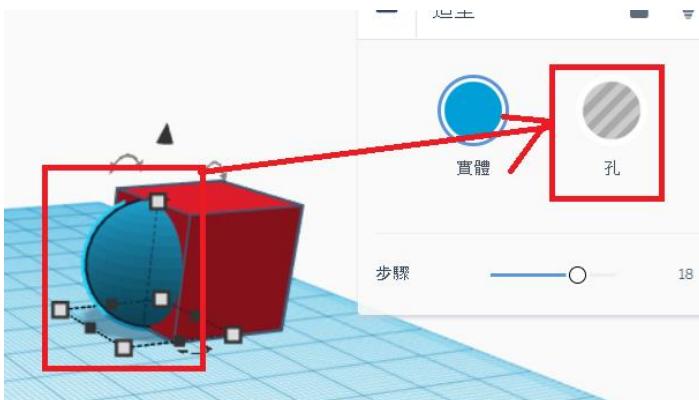
- (b) Select both objects, and click “form group”.



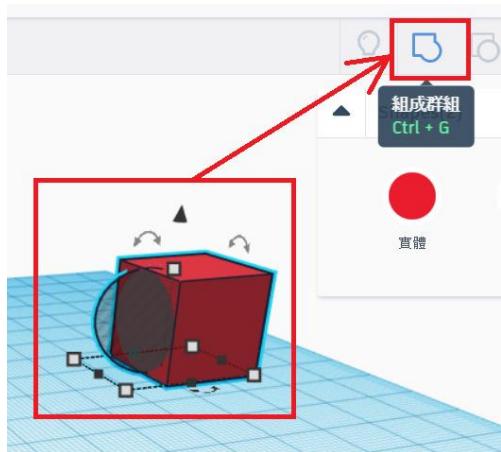
- (c) Try to move the object, and you will find they would move together.

5. Making holes

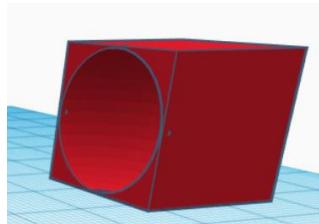
- (a) Create a cube, and a sphere. Half of the sphere is put inside the cube. Then, select the sphere, and click “hole”.



(b) Then, select both objects, and click “form group”.

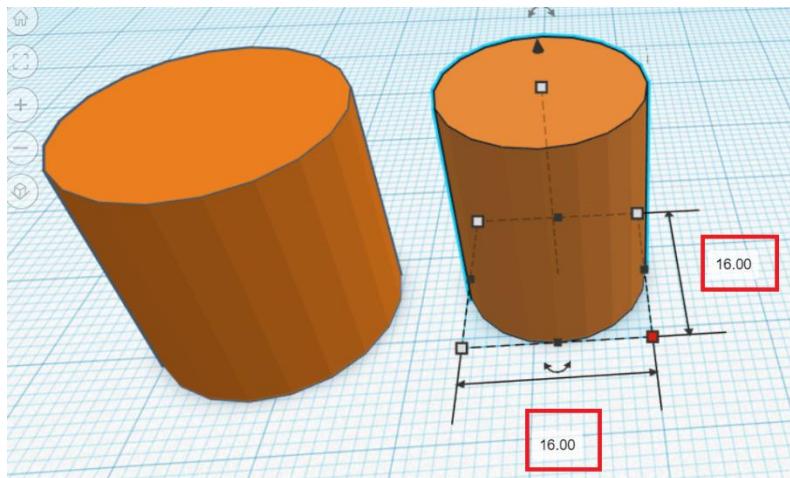


(c) You will find the cube will have a hole of hemisphere (半球體).

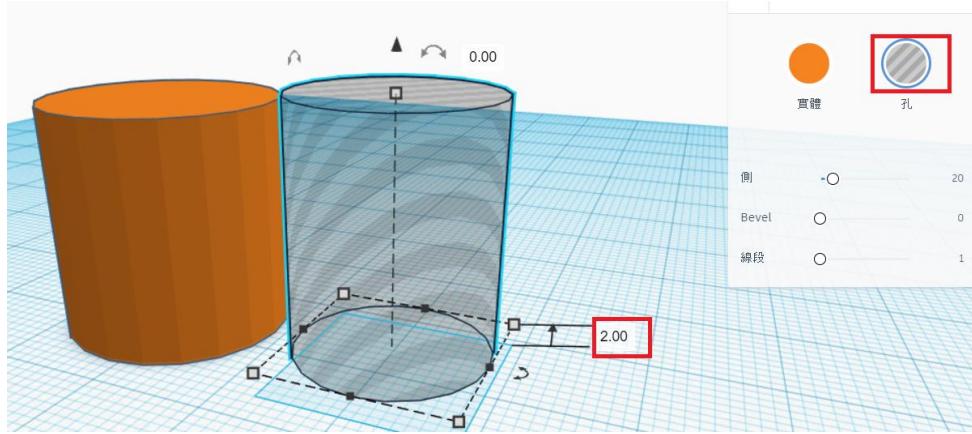


6. Alignment of objects

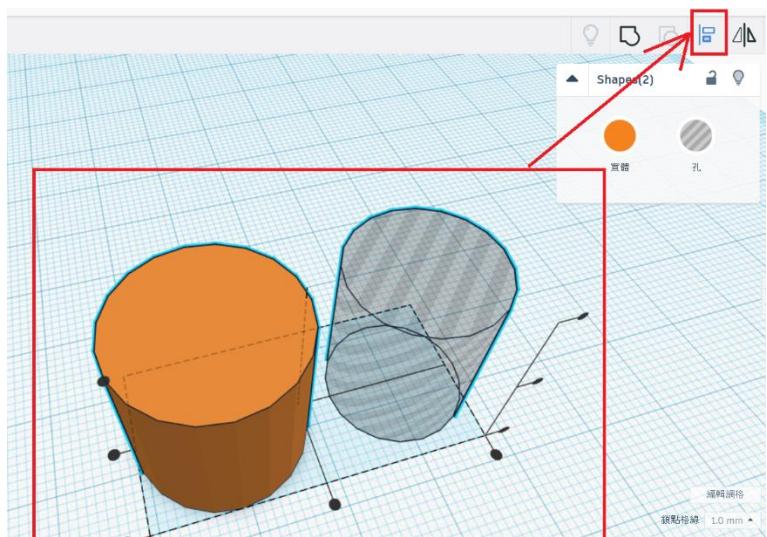
(a) Create two cylinders. Reduce the size of one cylinder's base into 16×16 .



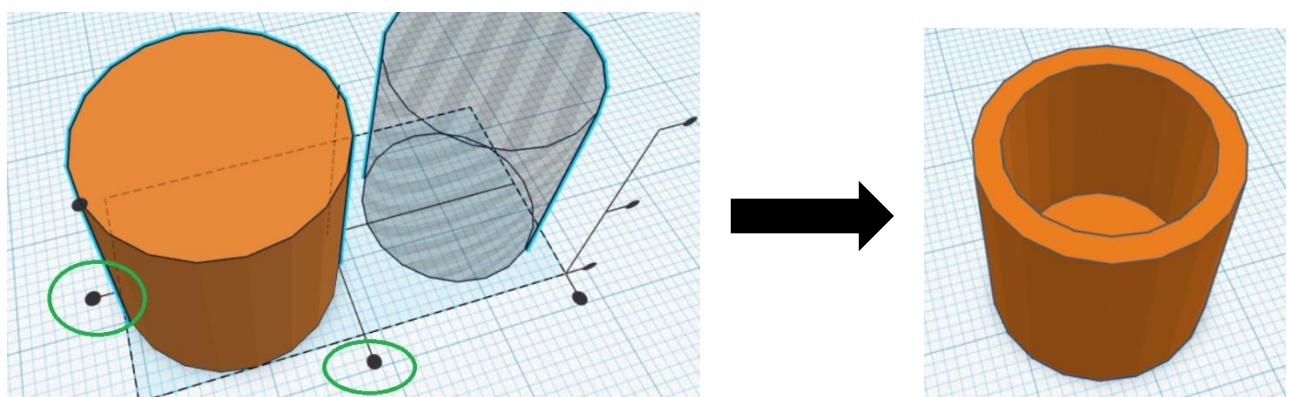
(b) Move the smaller cylinder away from the workplane by 2 units. Change it into a hole.



(c) Select both cylinders, and click “align”

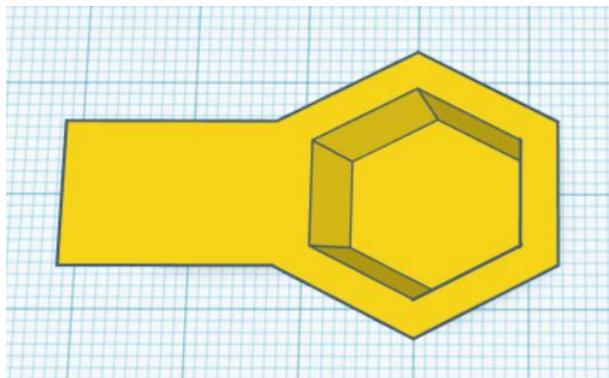
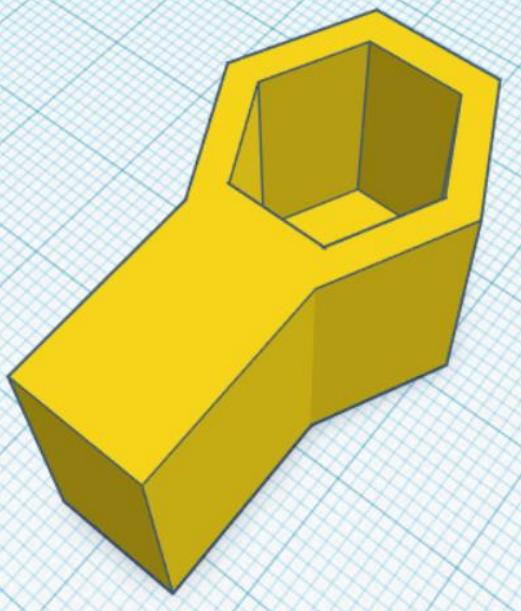


(d) Select the two dots on the workplane to align their centres. Then group them into group.



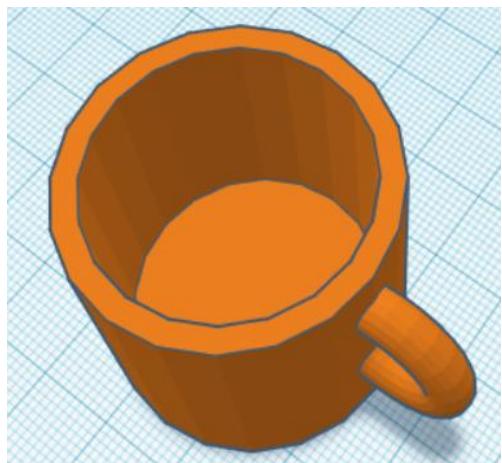
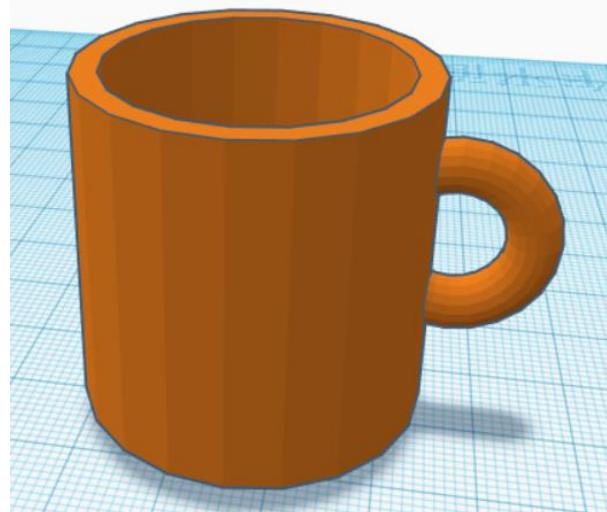
Exercise 1

Create the following shape on the TinkerCAD. (The size does not matter.)



Exercise 2

Create the following shape on the TinkerCAD. (The size does not matter.)

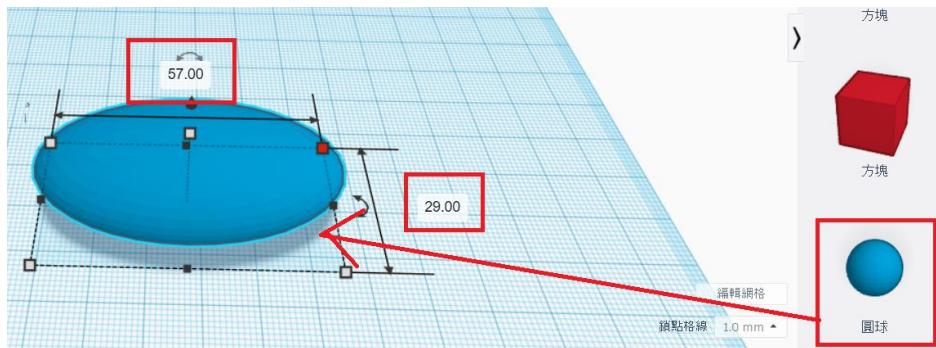


TinkerCAD (2): Advanced 3D Design

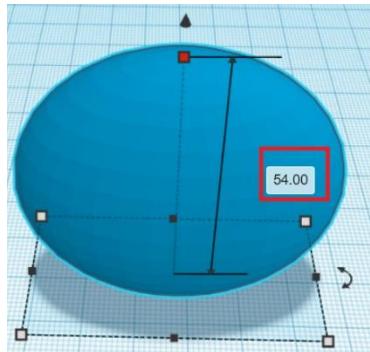
In the previous chapter, we have learnt some basic design functions in TinkerCAD such that you are able to create basic 3D design. In this chapter, we are going to learn some advanced 3D Design skills so that you can make more sophisticated 3D design to cope with your design need.

1. Copy and paste to create symmetric pattern

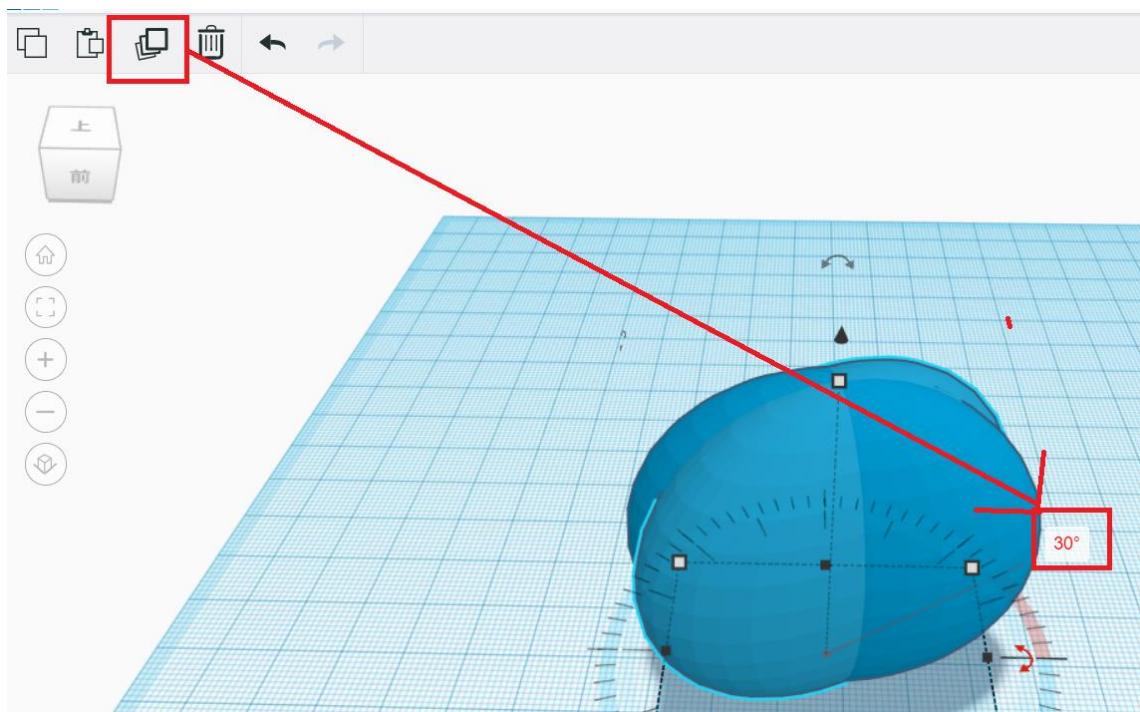
- Drag a sphere on the workplane. Then, resize the sphere into 57×29 .



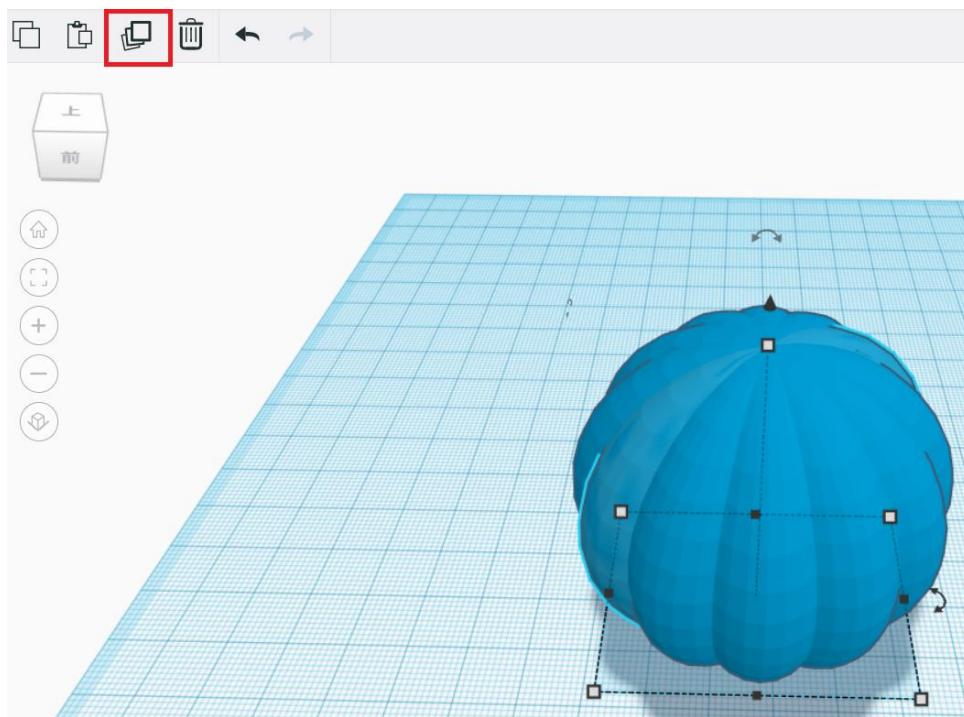
- Then, adjust the height to 54.



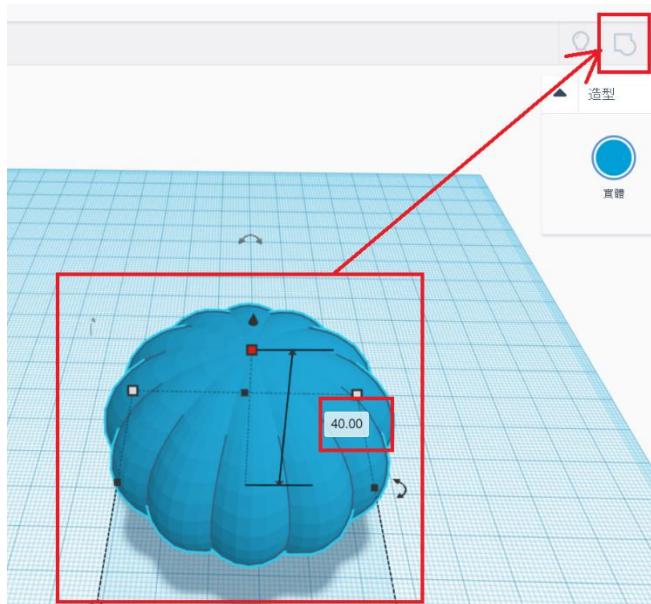
(c) Then, copy the object, and rotate it by 30° .



(d) Then, copy it continuously until it forms a complete shape as below.

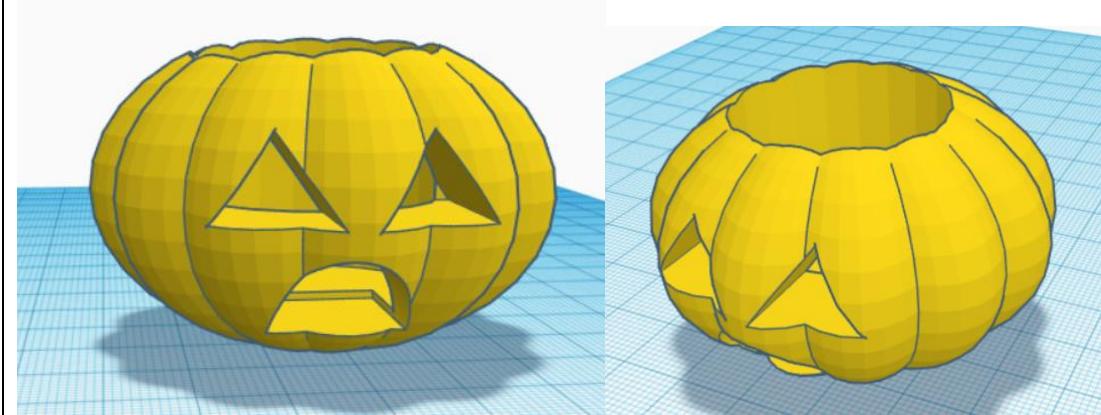


(e) Then, select all objects, and group them. Set the height of the grouped object to 40.



Exercise 1

Continue your work and make a pumpkin-shaped container.

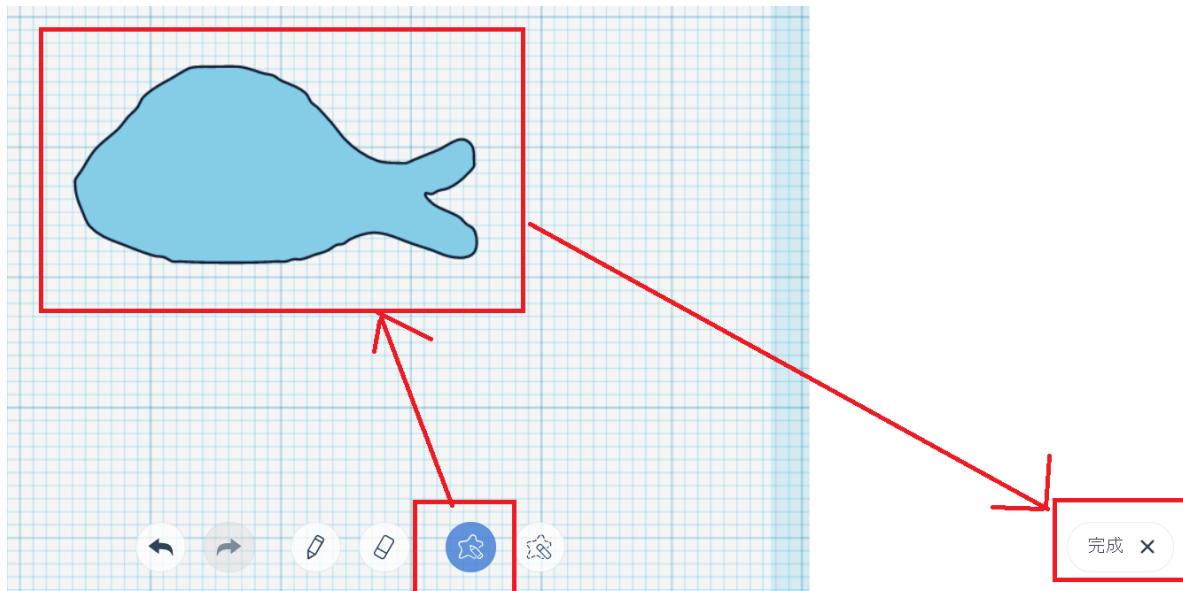


2. Scribble

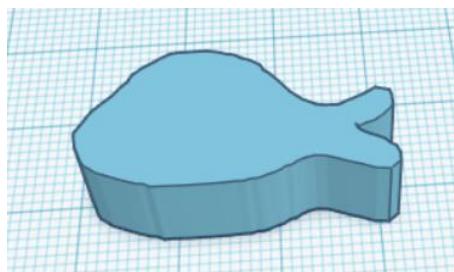
(a) Drag a scribble onto the workplane.



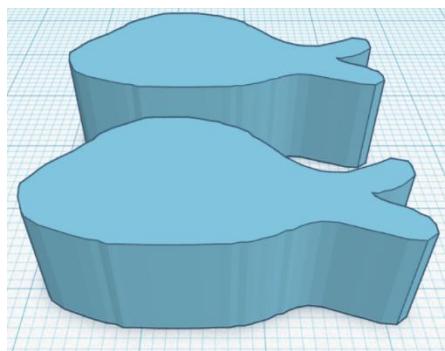
(b) Click the icon “draw model”, and draw a fish, and then click “finish”.



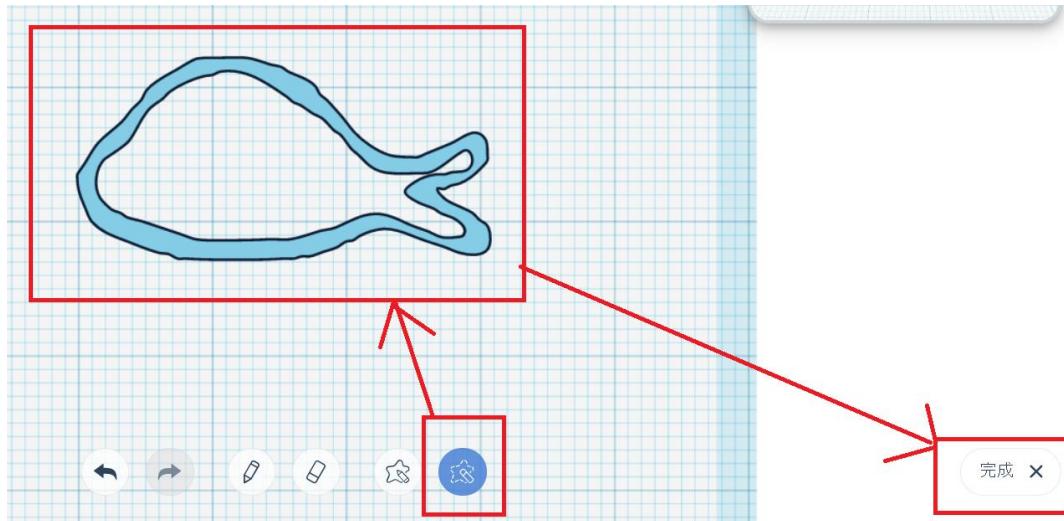
(c) Then, you can see a fish-based figure appeared.



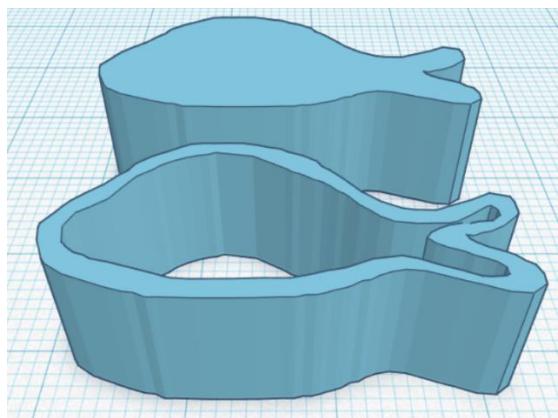
(d) Copy the fish object, and double click a fish object.



(e) Click “Delete model”, and draw the part you want to delete (the inner fish), and click “finish”.

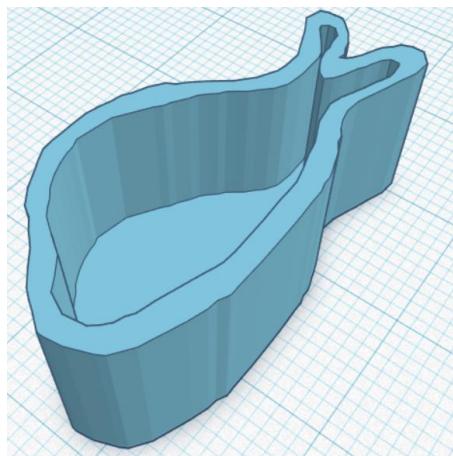
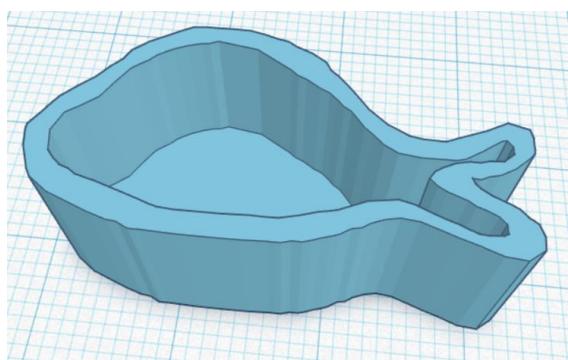


(f) Then, you will see one hollow fish, and one solid fish.



Exercise 2

Continue your work and make a fish-shaped container.



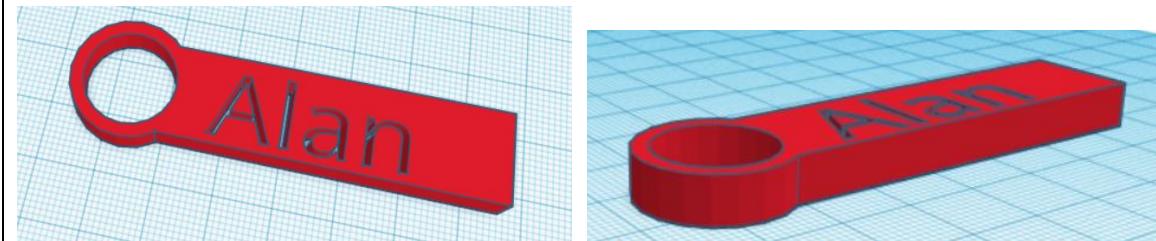
3. Text

- Drag the Text on the workplane.
- Change the text into “LW”.



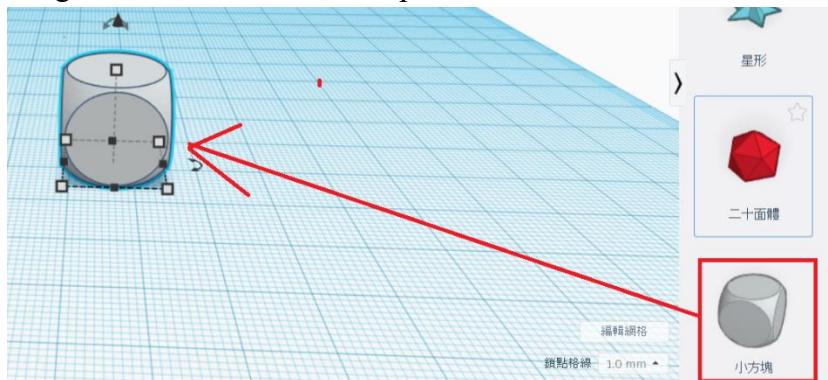
Exercise 3

Create the following keychain with your name.

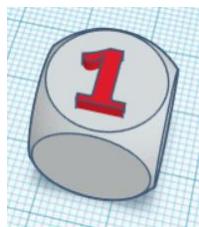


4. Changing workplane

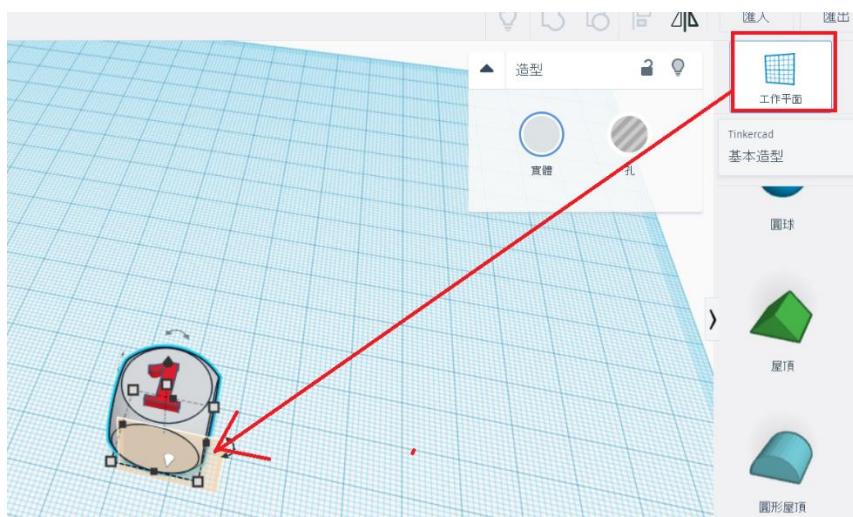
- Drag a small cube on the workplane.



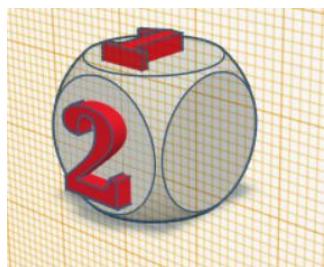
- Put the “1” on the top of the cube.



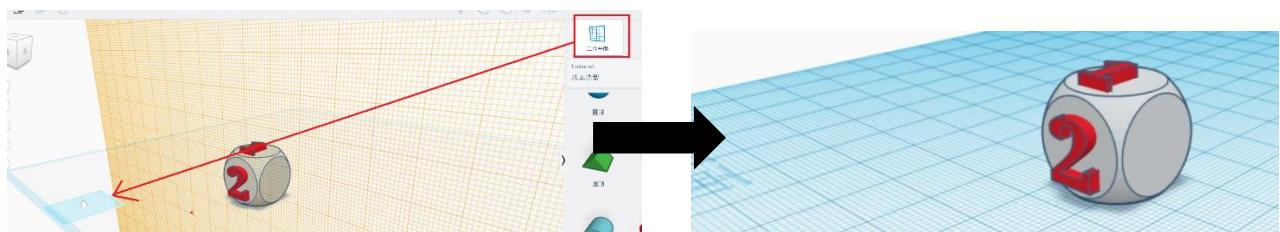
(c) Set another face of the cube as the workplane.



(d) Put “2” on that face of the cube.



(e) Resume the original workplane.



Exercise 4

Continue your work and make a dice.

