Lesson 1 Arithmetic with Python

Worksheet

K-W-L Chart

What I Know	What I Wonder	What I Learned

New Commands

```
print(“What’s your name?”)
```

```
print(5 + 6)
```

Procedures

Task 1: Translate the mathematical expressions below into Python code. Use Python to calculate the results.

Example	Symbol	Name	Python Representation
5 + 6 = 11	+	Addition	+
11 – 5 = 6	−	Subtraction	-
5 × 2 = 10	×	Multiplication	*
10 ÷ 5 = 2	÷	Division	/
5^2 = 25	x^n	Exponentiation	**
11 mod 5 = 1		Modulo	%

Task 2: Calculate the results of the mathematical expressions below:

181 + 125 + 669 = ?

2160 - 439 - 57 = ?

79 × 21 × 3 × 108 = ?

40257 ÷ 1917 = ?

156^4 = ?

40257 modulo 41 = ?

Task 3: Suppose your current computing skill level is 100. If you spend 1 hour per day in practising Python, your skill level will increase by 1% per day.

Create Python code to calculate the level of your computing skills if you keep doing this for 7 days. What will happen then after 180 days (and 365 days)?

Lesson 2 Text and Numbers in Python

Worksheet

K-W-L Chart

What I Know	What I Wonder	What I Learned

New Commands

```
print(type(“What’s your name?”))
```

```
print(type(5 + 6))
```

```
print(type(40257 / 1917))
```

```
input(“What’s your name? Your answer: ”)
```

Procedures

Task 1: Calculate the expressions below and determine the type of data of the results:

100000 + 365.12 = ?

1989.12 - 917.8 = ?

36.6 × 50.1 = ?

100.25 × 40 = ?

3.3^{10} = ?

2501 mod 2.5 = ?

Task 2: Read the script below and identify a new function:

```
print(type(input("What's your name? Your answer: ")))
print(type(input("How old are you? Your answer: ")))
print(type(input("How tall are you? Your answer:(metre) ")))
```

Task 3: Modify the script below and convert the data types of the input information:

```
print(type(input("Your school's postcode: ")))
print(type(input("Your school's name: ")))
print(type(input("Street/Block number nearby: ")))
print(type(input("Your body temperature:(°C) ")))
print(type(input("Today's temperature:(°C) ")))
print(type(input("Your lucky number: ")))
```

Lesson 3 Triangle Area Calculator

Worksheet

K-W-L Chart

What I Know	What I Wonder	What I Learned

New Commands

```
my_var = 1
```

```
my_var = my_var + 1
```

Procedures

Step 1 Use

Read and run the program.

```
1.  print("Triangle Area Calculator")
2.
3.  a = float(input("Value of side a:(centimetre) "))
4.  b = float(input("Value of base b:(centimetre) "))
5.  c = float(input("Value of side c:(centimetre) "))
6.
7.  s = (a + b + c) / 2
8.
9.  area = (s * (s-a) * (s-b) * (s-c)) ** 0.5
10.
11. print("The area of this triangle is", area, "square centimetres.")
```

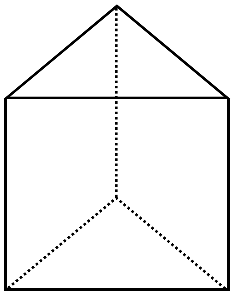
Questions:

- Why is **'float()'** used together with **'input()'**?
- How can the program remember the values of three sides of a triangle?
- How to calculate the semi-perimeter by using Python?
- How to calculate the area by using Python?
- Look at **Line 11**. Do you notice anything different?

Step 2 Modify

Task 1: Create another Triangle Area Calculator that calculates the area of a triangle with the given base and height.

Task 2: Based on the program for Triangle Area Calculator, create a Triangular Prism Volume Calculator.



Task 3: Create an application to calculate the capacity of a milk carton. Remember, you need to use the unit of measurement for liquids.

Step 3 Create

Create a calculator application that can calculate the perimeter and area of a plane shape or the volume of a solid shape, for instance a prism or prism-like figure.

Lesson 4   Body Mass Index Calculator

Worksheet

K-W-L Chart

What I Know	What I Wonder	What I Learned

New Commands

```
if BMI >= 30.0:

if BMI >= 18.5 and BMI <= 24.9:
```

Procedures

BMI Healthy Weight (Adults)

BMI	**Level**	**Meaning**
Below 18.5	Underweight	Being underweight may lead to a weakened immune system and feeling tired.
18.5 – 24.9	Healthy Weight	You have a healthy weight for your height.
25.0 – 29.9	Overweight	You are heavier than is healthy for someone of your height. Try to lose weight by keeping a balanced diet and physical activity.
30.0 and Above	Obese	

Source: UK NHS, US CDC

Calculate the BMI

Name	**Height (m)**	**Weight (kg)**	**BMI**	**Weight Status**
(*Your Name*)				
Ai Fukuhara	1.55	48		
Angelina Jolie	1.69	54		
Arnold Schwarzenegger	1.88	113		
Benedict Cumberbatch	1.83	79		
Jackie Chan	1.70	65		
Kobe Bryant	1.98	96		
Małgorzata Dydek	2.18	101		
Maya Plisetskaya	1.77	59		
Sarah Hyland	1.57	45		
Serena Williams	1.75	70		
Tom Cruise	1.70	77		
Yao Ming	2.29	141		
Zoe	1.70	72		

Step 1 Predict

Read the program.

```python
print("BMI Calculator")

height = float(input("Height:(metre) "))
weight = float(input("Weight:(kilogram) "))

BMI = weight / (height**2)
print(BMI)

if BMI >= 30.0:
    print("Obese")

if BMI >= 25.0 and BMI <= 29.9:
    print("Overweight")

if BMI >= 18.5 and BMI <= 24.9:
    print("Healthy Weight")

if BMI < 18.5:
    print("Underweight")
```

Step 2 Run

Run the program.

Step 3 Investigate

Task 1: Comment on the code. Briefly explain the function.

Task 2: Investigate the conditions of the statements. Why does the program use the ‘≥’ and ‘≤’ operators to define the lowest and/or highest values of a range?

Task 3: Investigate the indentation. Remove the leading whitespace before the `print(“obese”)/print(“overweight”)/print(“healthy weight”)/print(“underweight”)`. Run the program and see what happens. Does the program still work?

Task 4: Calculate Zoe’s BMI and output her weight status. Could the program decide Zoe’s weight status? Why or why not?

Step 4 Modify and Make

Create a BMI Calculator for children and teenagers.

References:

The BMI Calculator created by the UK’s NHS: https://www.nhs.uk/live-well/healthy-weight/bmi-calculator

The BMI Calculator created by the US’s CDC:

For children and teenagers: https://www.cdc.gov/healthyweight/bmi/calculator.html

For adults:

https://www.cdc.gov/healthyweight/assessing/bmi/adult_BMI/english_bmi_calculator/bmi_calculator.html

Lesson 5 Buy Low, Sell High

Worksheet

K-W-L Chart

What I Know	What I Wonder	What I Learned

New Commands

```
import random
```

```
random.randint(start, end, step)
```

```
elif <…>:
```

```
else:
```

Procedures

Step 1 Predict

Read the program.

```
import random

price_a = random.randint(100, 200)
print("The traded price now is", price_a)

order = int(input("Your posted price: "))

price_b = random.randint(100, 200)
print("The traded price now is", price_b)

if order > price_b:
    print("Current price is lower than your posted price. Buy it.")

elif order < price_b:
    print("Current price is higher than your posted price. Sell it.")

else:
    print("No transaction.")
```

Questions:

- ○ What is the control structure?
- ○ What does **'import random'** mean?
- ○ What are the values of the variable **'price_a'** and **'price_b'**?

Step 2 Run

Run the program.

Step 3 Investigate

Task 1: Comment on the code. Briefly explain the function.

Task 2: Investigate the conditions of the statements. How does the program compare the posted price with the traded price?

Task 3: What is the condition of **'else'**. Write down its condition.

Step 4 Modify

Task 4: Use the **if** statements instead to rewrite the example program.

Step 5 Make

Create a digital artefact with conditionals that helps with decision making.

Lesson 6 Repay Loans

Worksheet

K-W-L Chart

What I Know	What I Wonder	What I Learned

New Commands

```
for i in range(start, end, step):
```

```
round()
```

Procedures

Step 1 Predict

Read the program.

```python
print("Loan Payment Calculator: Even Principal Payments")

loan = int(input("Loan Amount (GBP £): "))
term = int(input("Number of Payments (months): "))

print("The annual interest rate is 5.4%")
rate = 0.054
balance = loan

for i in range(1, term + 1):

    principal_payment = round(loan / term, 2)
    interest_payment = round(balance * (rate/12), 2)
    print("Repayment", i, ":", principal_payment + interest_payment)

    balance = loan - principal_payment*i
    print("Unpaid Balance:", balance)
```

Questions:

- ○ What is the control structure?
- ○ What does **'range(1, term+1)'** mean?
- ○ How to calculate the unpaid balance of the loan after each repayment?

Step 2 Run

Run the program.

Step 3 Investigate

Task 1: Summarise the mathematical expressions.

Task 2: Based on the result returned by the Loan Payment Calculator, create a graph to present the loan repayments.

Step 4 Modify

Task 3: Modify the program to pay the loan repayments annually.

Task 4: The interest rate can be different across different repayment terms. Add the conditionals into the program you have created in the third task.

Term (Years)	Interest Rate
3	5.4%
5	5.6%
8	6.1%
10	5.8%

Step 5 Make

Create a Loan Repayment Calculator application for even total payments.

Lesson 7 Toss a Coin

Worksheet

K-W-L Chart

What I Know	What I Wonder	What I Learned

Procedures

Total Flips	Heads	% Heads	Tails	% Tails
2				
4				
6				
8				
10				
12				

Step 1 Predict

Read the program.

```python
import random

countheads = 0
counttails = 0

n = int(input("The number of times to throw up a coin: "))

for i in range(0, n):
    result = random.randint(0, 1)

    if result == 0:
        countheads += 1
        print(result)

    else:
        counttails += 1
        print(result)

print("Result:")
print("Heads up:", countheads)
print("Tails up:", counttails)

print("Chance:")
print("Heads : Tails", round(countheads/n, 2), ":", round(counttails/n, 2))
```

Questions:

- ○ How does the program represent the results of heads and tails?
- ○ How does the program count the number of heads or tails?
- ○ Explain the use of the **if…else** statement and the **for** loop.

Step 2 Run

Run the program.

Step 3 Investigate

Answer the questions.

Step 4 Modify

Task 1: Add a line of code to calculate the probabilities of getting heads and tails.

Task 2: Change the number of the toss into a random number generated between 4040 and 80640.

Task 3: Bet on the toss of the coin. Modify the example program to ask the user to input their guess before the coin-tossing experiment. Report the probability result and the betting result.

Step 5 Make

Based on the example program, modify the functions. Suppose the program is to model the dice-rolling process, how should you edit the code?

Record the probability to get each number of the dice.

Lesson 8 Guess the Number

Worksheet

K-W-L Chart

What I Know	What I Wonder	What I Learned

New Commands

```
while True:
```

```
break
```

Procedures

Step 1 Predict

Read the program.

```python
import random

num = random.randint(1, 100)

t = 7

print("Guess a number between 1 and 100. You have 7 attempts.")
print("Let's start.")

while True:
    guess = int(input("Guess the number: "))
    if guess == num:
        print("BINGO!")
        break

    elif guess > num:
        t -= 1
        print("High! You have", t, "attempts left.")

    elif guess < num:
        t -= 1
        print("Low! You have", t, "attempts left.")

    if t == 0:
        print("The correct number is", num)
        print("Game Over!")
        break
```

Step 2 Run

Run the program.

Step 3 Investigate and Modify

Round	1st Try	2nd Try	3rd Try	4th Try	5th Try	6th Try	7th Try	Number
Example	*7*	*85*	*60*	*40*	*20*	*11*	*9*	***9***

Note

The 33 reserved words in Python:

and	as	assert	break	class	continue
def	del	elif	else	except	finally
for	from	False	global	if	import
in	is	lambda	nonlocal	not	None
or	pass	raise	return	try	True
while	with	yield			

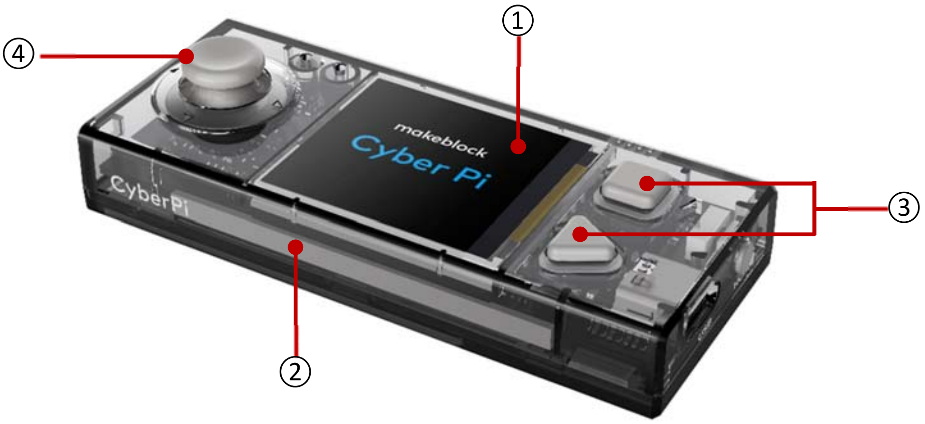
Lesson 9~10 Python Quizzes

Worksheet

K-W-L Chart

What I Know	What I Wonder	What I Learned

New Commands



`import cyberpi`

① **Screen:** `cyberpi.display`, `cyberpi.console`

`cyberpi.display.clear()`

`cyberpi.console.print()`

`cyberpi.console.println()`

② **LED Strip:** `cyberpi.led`

`cyberpi.led.off()`

`cyberpi.led.off(id="3")`

`cyberpi.led.on()`

`cyberpi.led.on("green", id="all")`

`cyberpi.led.on("red", id=3)`

`cyberpi.led.show("orange yellow cyan blue purple")`

`cyberpi.led.play(name="firefly")`

③ **Buttons A and B:** `cyberpi.controllor.is_press()`

`cyberpi.controller.is_press("a")`

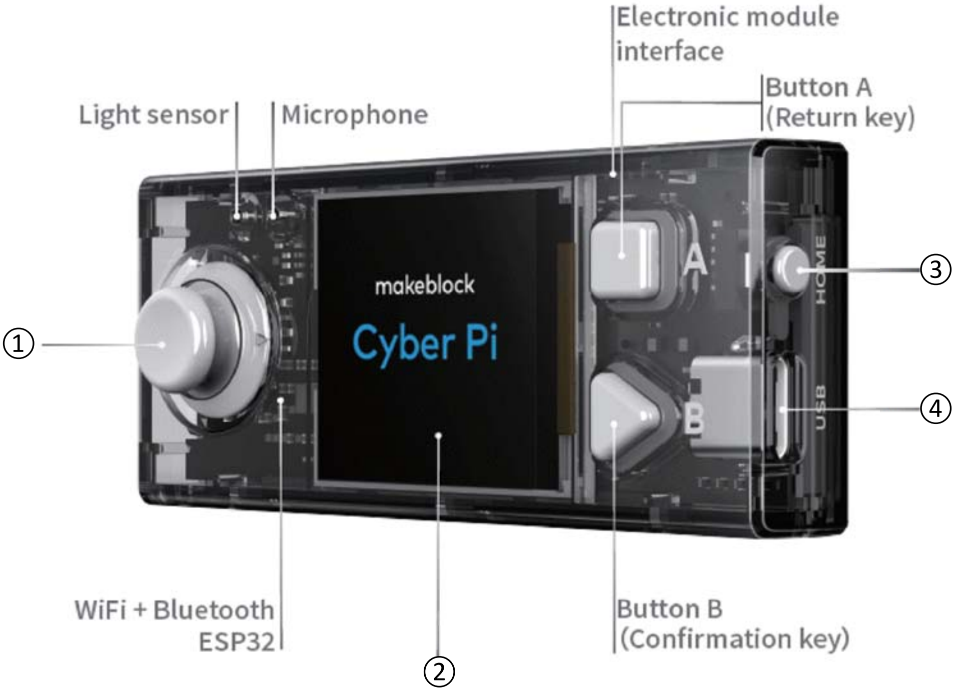
`cyberpi.controller.is_press("b")`

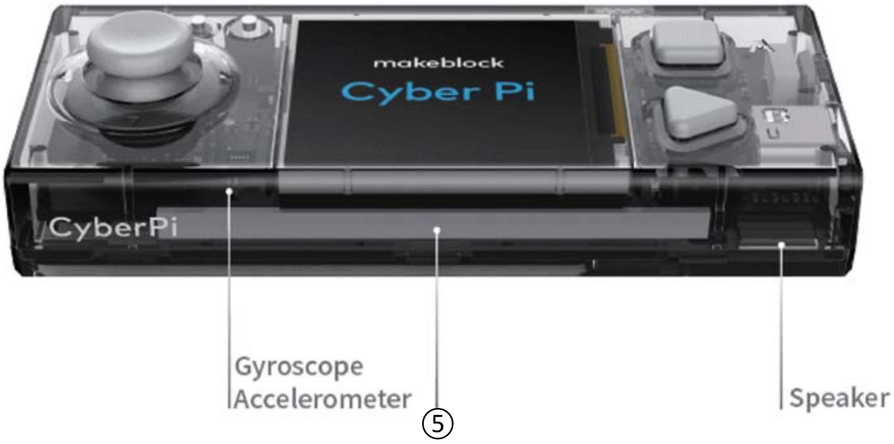
④ **Joystick:** `cyberpi.controller.is_press()`

```
cyberpi.controller.is_press(“up”)

cyberpi.controller.is_press(“down”)

cyberpi.controller.is_press(“right”)

cyberpi.controller.is_press(“left”)

cyberpi.controller.is_press(“middle”)
```

Procedures

Features of CyberPi





Step 1 Predict

Read the program.

```python
import cyberpi

cyberpi.display.clear()
cyberpi.led.off()

cyberpi.console.println("A - True")
cyberpi.console.println("B - False")

cyberpi.led.on(255, 255, 255)
cyberpi.console.println("Python is a compiled language.")
cyberpi.console.println("Your Answer:")

while True:

    if cyberpi.controller.is_press("a"):
        cyberpi.led.on(255, 0, 0)
        cyberpi.console.println("Incorrect.")
        cyberpi.console.println("Correct Answer: False")
        break

    if cyberpi.controller.is_press("b"):
        cyberpi.led.on(0, 255, 0)
        cyberpi.console.println("Correct!")
        break
```

Questions:

- Which module is imported? Explain why we should import this module.
- How to print text on the screen?
- How to light up/off the LED strip?
- How to set the color of the LED? How to represent the LED colors in Python?
- How to enter the answer?
- How to evaluate the answer?

Step 2 Run

Run the program.

Step 3 Investigate

Investigate the physical components and API code samples of the screen, the LED strip, the buttons, and the joystick.

Step 4 Modify and Make

Task 1: Modify the quiz and the answer in the example program.

Task 2: Modify the LED's lighting effects.

Tip:

Firefly: `cyberpi.led.play(name=“firefly”)`

Rainbow: `cyberpi.led.play(name=“rainbow”)`

Spoondrift: `cyberpi.led.play(name=“spoondrift”)`

Meteor Shower: `cyberpi.led.play(name=“meteor_blue”)`; `cyberpi.led.play(name=“meteor_green”)`

Flash: `cyberpi.led.play(name=“flash_orange”)`; `cyberpi.led.play(name=“flash_red”)`

Task 3: Use the joystick instead of the buttons to enter the answer.

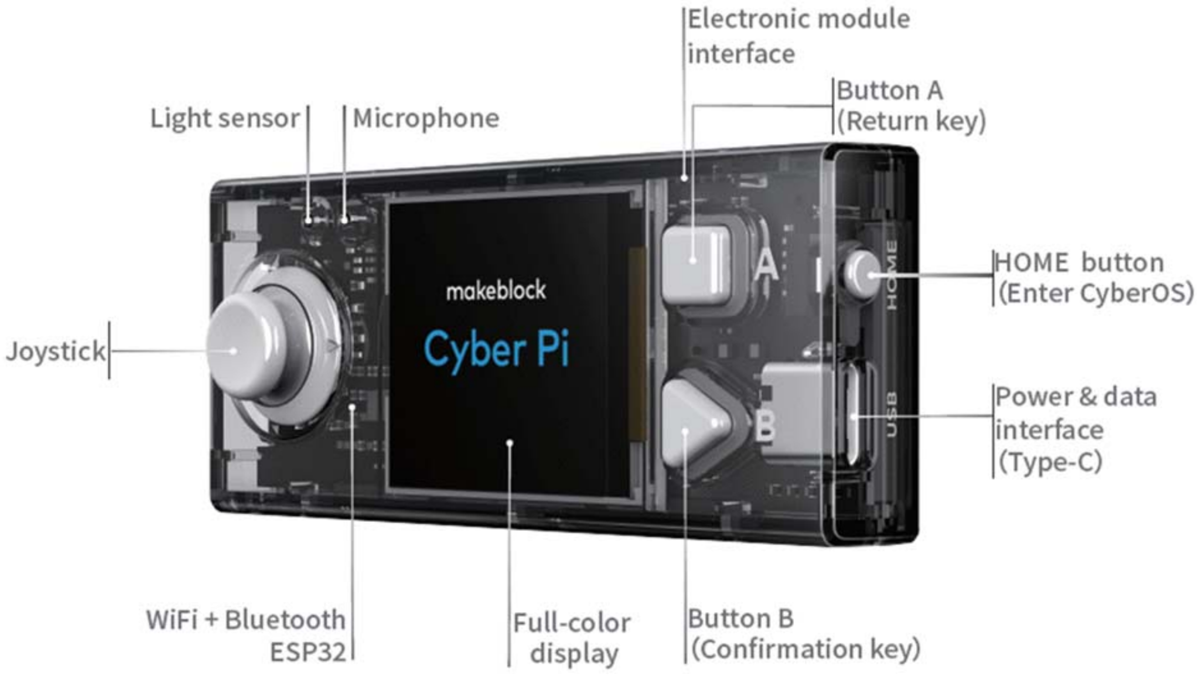
Lesson 11 Data Protection and Passwords

Worksheet

K-W-L Chart

What I Know	What I Wonder	What I Learned

Features of CyberPi



Procedures

Step 1 Predict

Read the program.

```python
import cyberpi

cyberpi.display.clear()
t = 0

while True:
    print("Create a password")
    pin_1 = input("Type password: ")
    pin_2 = input("Type password again: ")
    if pin_2 == pin_1:
        print("Success!")
        break
    else:
        print("Passwords don't match. Try again.")


print("Sign in your account")
cyberpi.console.println("Sign in")

while t < 3:
    pin = input("Password: ")
    cyberpi.console.print("Password: ")
    cyberpi.console.println(pin)
    if pin == pin_1:
        cyberpi.console.println("Success!")
        break
    else:
        cyberpi.console.println("Incorrect. Try again.")
        t += 1
        if t == 3:
            cyberpi.console.println("Too many failed attempts.")
```

Questions:

- How to create a password for a new account?
- How to verify the password entered by a user?
- Compare the two **while** loops used in the example program. What is the difference between them?
- Identify the syntax that enables the following function: ‘A user is given 3 attempts to enter the password of the account.’ If the user fails 3 times, the user is blocked from entering the password .

Step 2 Run

Run the program.

Step 3 Investigate

Answer the questions.

Step 4 Modify

Task 1: Add the function that allows the user to create a username when the user signs in.

Task 2: Add some lighting effects as the indicator of a successful login. Use what you have learned to program the CyberPi.

Tip:

```
cyberpi.led.on(255, 255, 255)

cyberpi.led.on(“green”, id = “all”)

cyberpi.led.on(“red”, id = 3)

cyberpi.led.show(“orange yellow cyan blue purple”)

cyberpi.led.play(name = “firefly”)

cyberpi.led.off(id = “3”)
```

Task 3: Modify the second **‘while’** loop. Use the LEDs as an indicator to remind the number of attempts.

Task 4: Modify the conditional expressions. Verify the following three conditions:

The input username is incorrect;

The input password is incorrect;

Both the username and password are incorrect.

Step 5 Make

Develop a bank card reader

- First, create a PIN for the bank account and store it on the computer.
- When a sender starts a transaction, ask the sender to type the name (or other identification code) of the receiver and the amount of the transaction.
- Then ask the sender to type the PIN.
- Generate a random 4-digit verification code and display it on CyberPi's screen. Ask the sender to enter the verification code.
- Check the PIN and verification code. If both are correct, display the transaction information (including the receiver's name or ID and the transaction amount) on the screen.

  However, if either the PIN or the verification code is incorrect, ask the sender to insert them again. The sender has limited attempts (for example, 3 attempts).
- Ask the sender to check the transaction information and confirm the transaction by pressing Button B of the CyberPi.

Lesson 12 Normal Distribution

Worksheet

K-W-L Chart

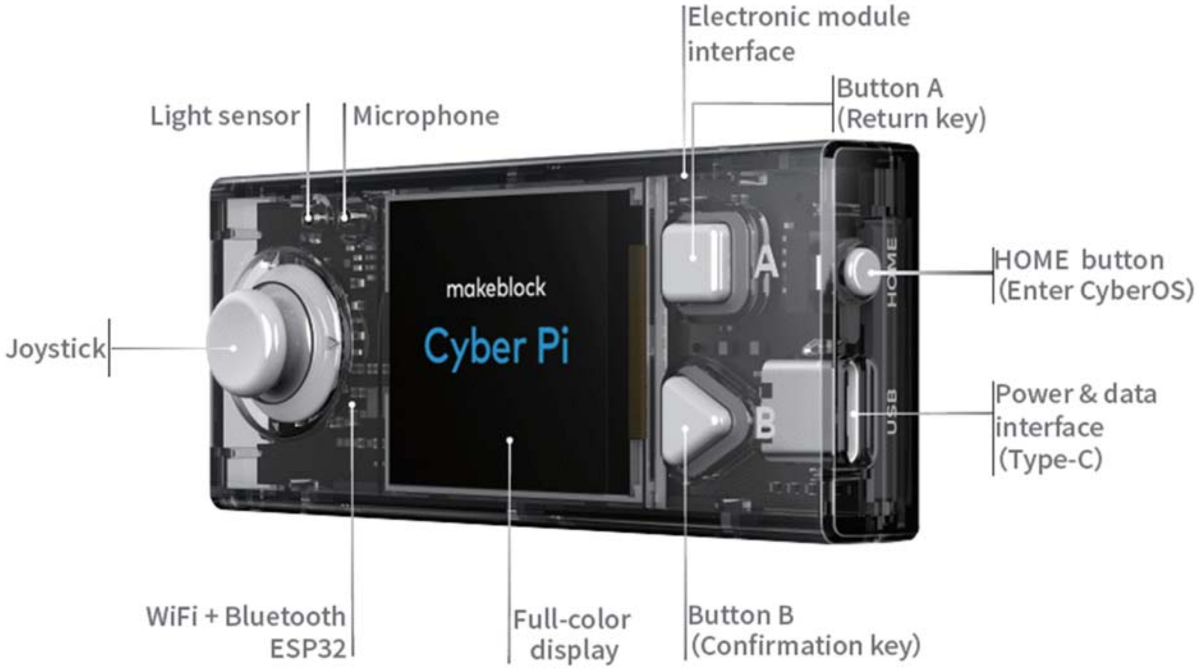
What I Know	What I Wonder	What I Learned

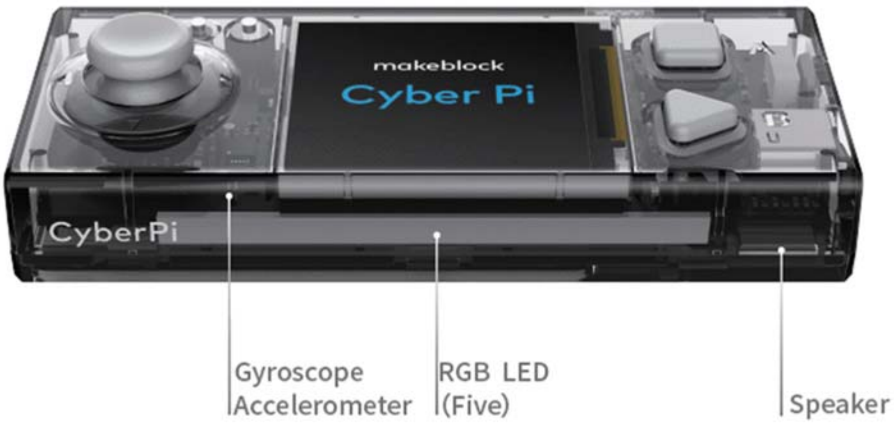
New Commands

```
cyberpi.display.set_brush()
```

```
cyberpi.barchart.add()
```

Features of CyberPi





Procedures

Step 1 Predict

Read the program.

```python
import cyberpi, random

cyberpi.display.clear()

n = int(input("The number of times to roll 2 dice: "))
t = 0
sum_list = [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]
count_list = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

while t < n:
    dice_x = random.randint(1, 6)
    dice_y = random.randint(1, 6)
    print("(", dice_x, ",", dice_y, ")")
    result = dice_x + dice_y
    t += 1
    sum_index = sum_list.index(result)
    count_list[sum_index] += 1

cyberpi.display.set_brush(128, 0, 0)
cyberpi.barchart.add(round(count_list[0]/n * 200, 2))
cyberpi.display.set_brush(220, 20, 60)
cyberpi.barchart.add(round(count_list[1]/n * 200, 2))
cyberpi.display.set_brush(255, 0, 0)
cyberpi.barchart.add(round(count_list[2]/n * 200, 2))
cyberpi.display.set_brush(205, 92, 92)
cyberpi.barchart.add(round(count_list[3]/n * 200, 2))
cyberpi.display.set_brush(233, 150, 122)
cyberpi.barchart.add(round(count_list[4]/n * 200, 2))
cyberpi.display.set_brush(255, 69, 0)
cyberpi.barchart.add(round(count_list[5]/n * 200, 2))
cyberpi.display.set_brush(255, 165, 0)
cyberpi.barchart.add(round(count_list[6]/n * 200, 2))
cyberpi.display.set_brush(255, 215, 0)
cyberpi.barchart.add(round(count_list[7]/n * 200, 2))
cyberpi.display.set_brush(240, 230, 140)
cyberpi.barchart.add(round(count_list[8]/n * 200, 2))
cyberpi.display.set_brush(255, 255, 0)
cyberpi.barchart.add(round(count_list[9]/n * 200, 2))
cyberpi.display.set_brush(154, 205, 50)
cyberpi.barchart.add(round(count_list[10]/n * 200, 2))
```

Step 2 Run

Run the program.

Step 3 Investigate

Investigate the points below:

- The modules imported in this program;
- The variables displayed in the bar chart;
- Are **'sum_list'** and **'count_list'** variables? What are the values of them?
- The function that sets the colour of the bar chart;
- The function that creates the bars.

Step 4 Modify and Make

Task: Simulate the experiment of tossing three coins simultaniously. Calculate all the possible outcomes in this experiment and visualise the distribution with a bar chart.

Head, Head, Head	

Lesson 13　　Data Storage

Worksheet

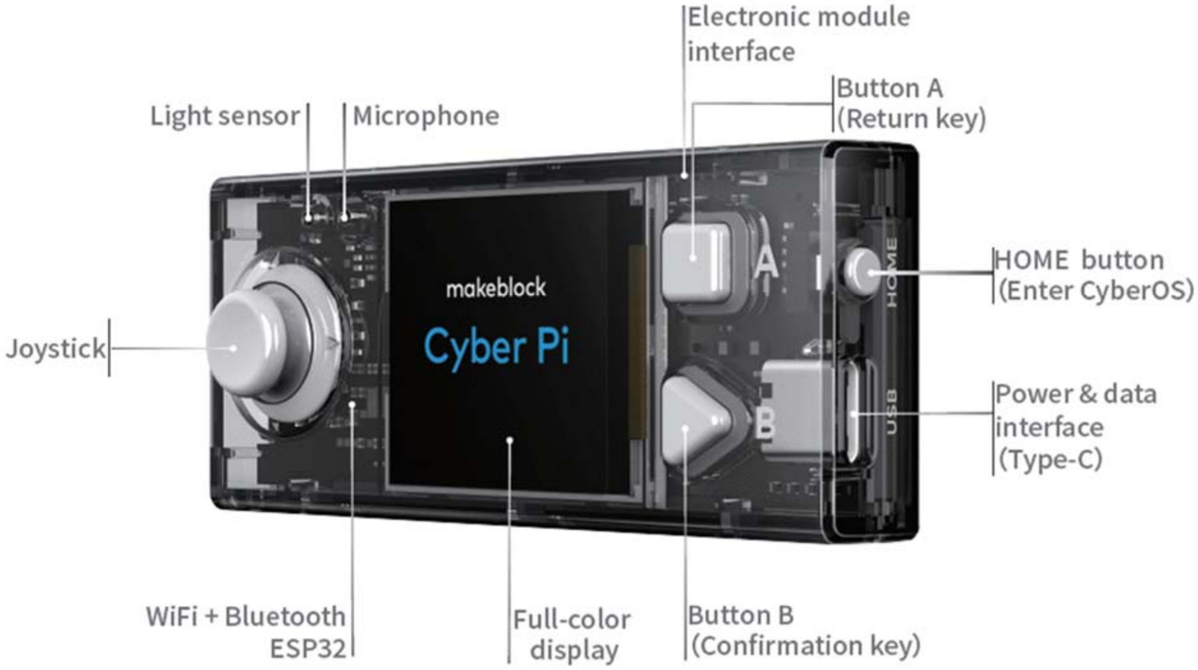
K-W-L Chart

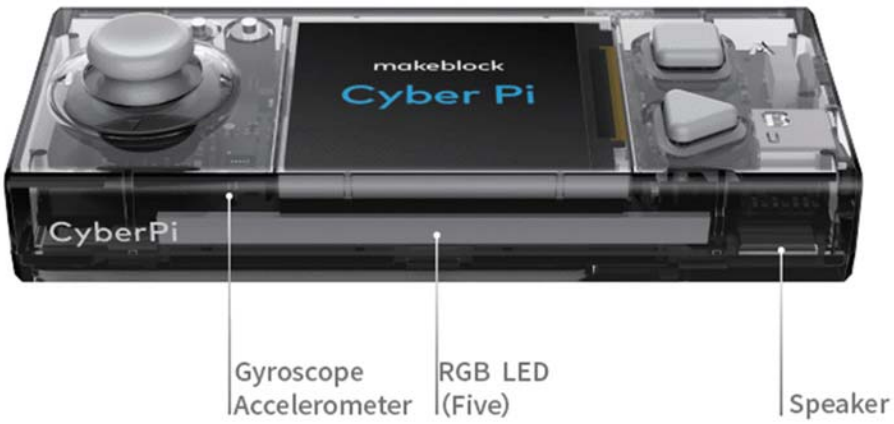
What I Know	What I Wonder	What I Learned

New Commands

```
import psutil

psutil.cpu_percent()

psutil.cpu_count()

psutil.cpu_freq()

psutil.virtual_memory()

cyberpi.linechart.add()

cyberpi.chart.set_name()

cyberpi.audio.play_tone()
```

Features of CyberPi

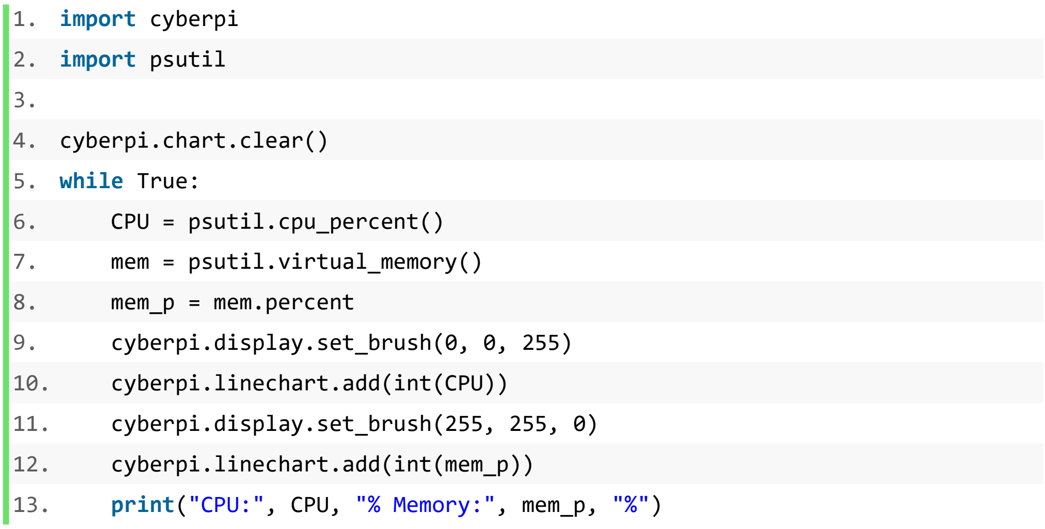




Procedures

Step 1 Predict

Read the program.



```
import cyberpi
import psutil

cyberpi.chart.clear()
while True:
    CPU = psutil.cpu_percent()
    mem = psutil.virtual_memory()
    mem_p = mem.percent
    cyberpi.display.set_brush(0, 0, 255)
    cyberpi.linechart.add(int(CPU))
    cyberpi.display.set_brush(255, 255, 0)
    cyberpi.linechart.add(int(mem_p))
    print("CPU:", CPU, "% Memory:", mem_p, "%")
```

Step 2 Run

Run the program.

Step 3 Investigate

Investigate the points below:

- The library for calling functions to monitor the CPU and memory usage
- The two variables plotted in the line chart;
- The function that plots the lines;
- The function that set the color of the line.

Step 4 Modify and Make

Task: Add control structures and light effects to the example program. Define the thresholds of the alarm and the corresponding alarm indicators.

Note: CPU performance

Lower CPU performance	Higher CPU performance
Single-core	Multi-core
Low clock speed	High clock speed
Small or no cache	Large, multi-level cache

Lesson 14 Remix Culture

Worksheet

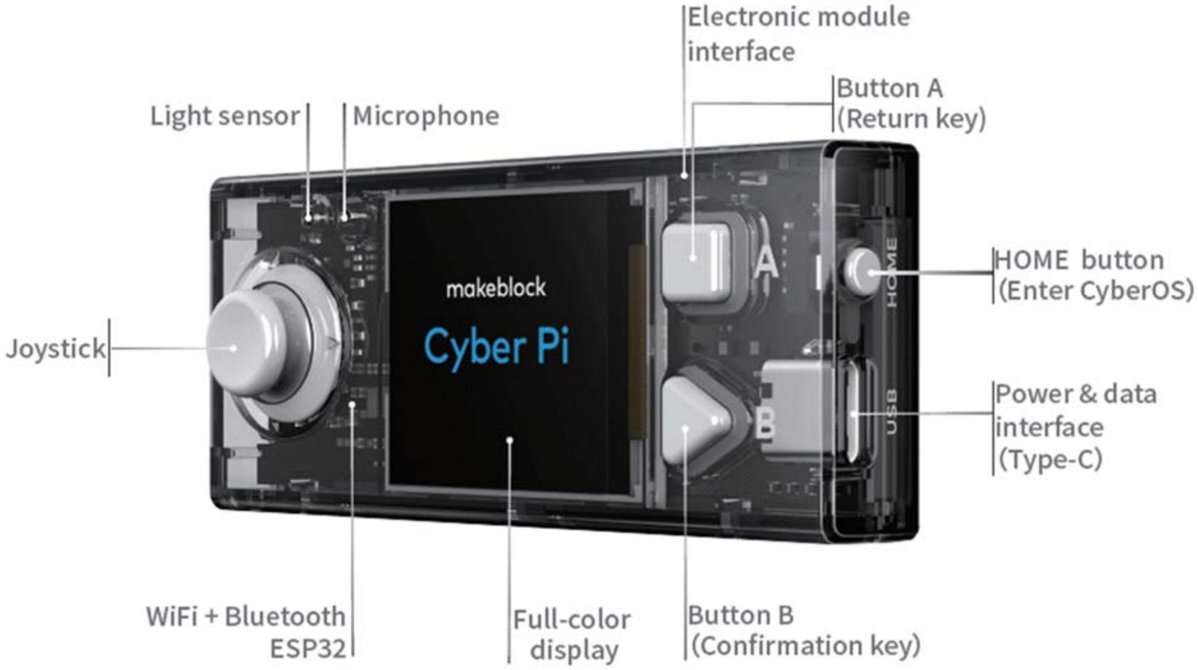
K-W-L Chart

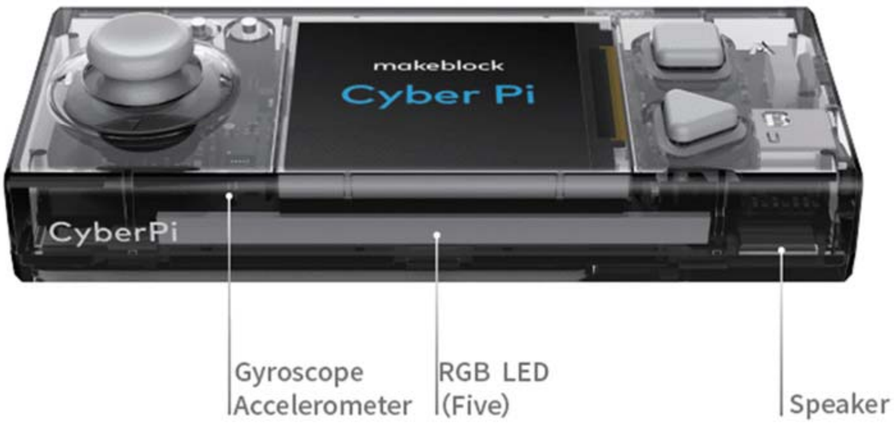
What I Know	What I Wonder	What I Learned

New Commands

```
from pynput.keyboard import Key, Listener

from pynput.keyboard import Button, Controller

cyberpi.is_tiltleft

cyberpi.is_tiltright

cyberpi.is_tiltforward

cyberpi.is_tiltback

cyberpi.audio.play_music()
```

Features of CyberPi


Electronic module interface
Light sensor
Microphone
Button A (Return key)
HOME button (Enter CyberOS)
Joystick
makeblock
Cyber Pi
Power & data interface (Type-C)
WiFi + Bluetooth ESP32
Full-color display
Button B (Confirmation key)



Procedures

Step 1 Predict

Read the program.

```python
import cyberpi
from pynput.keyboard import Key, Listener
cyberpi.audio.set_vol(50)

def bar1():
    cyberpi.audio.play_music(60, 0.2)
    cyberpi.audio.play_music(64, 0.2)
    cyberpi.audio.play_music(67, 0.2)

def bar2():
    cyberpi.audio.play_music(64, 0.2)
    cyberpi.audio.play_music(65, 0.2)
    cyberpi.audio.play_music(69, 0.2)

def bar3():
    cyberpi.audio.play_music(64, 0.2)
    cyberpi.audio.play_music(67, 0.2)
    cyberpi.audio.play_music(71, 0.2)

def bar4():
    cyberpi.audio.play_music(65, 0.2)
    cyberpi.audio.play_music(69, 0.2)
    cyberpi.audio.play_music(72, 0.2)

def on_press(key):
    if key.char == "1":
        cyberpi.led.on("red")
        bar1()
    if key.char == "2":
        cyberpi.led.on("orange")
        bar2()
    if key.char == "3":
        cyberpi.led.on("yellow")
        bar3()
    if key.char == "4":
        cyberpi.led.on("green")
        bar4()

def on_release(key):
    cyberpi.led.off()
    pass

with Listener(on_press=on_press, on_release=on_release) as listener:
    listener.join()
```

Step 2 Run

Run the program.

Step 3 Investigate

Investigate the points below:

- Identify the new module that can monitor the input from your keyboard.
- What is meant by the Python keyword ‘def’?
- Why does it separate the set of expressions within each ‘def’ code block?
- How to produce interactive sound and light effects?

Step 4 Modify and Make

Task 1: Modify the parameters in the example program. Use letters to replace the numeric parameters.

Task 2: Modify the ‘bar’ functions. Find some pieces of songs from music textbooks and combine different parts of them together to make a new song. Consider how to combine the sound effects and LED lights.