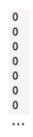
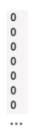
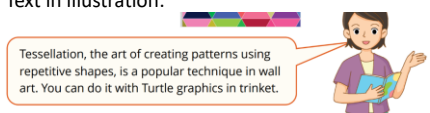
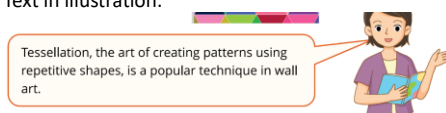


Errata for Junior Secondary One Click ICT Python Programming

As Trinket will cease operations in early August 2026, the corresponding sections of the textbook require the following amendments:

Page	Original	Change
Contents	2.1 Using trinket platform  Using trinket platform	2.1 Using Python coding platform  Using Python coding platform
7 – 10	Activity and book content	See Appendix
24	Procedure 1. In Trinket, type the following codes to try out the input function.	Procedure 1. In the coding platform, type the following codes to try out the input function.
28	Procedure 1. Think and write down the output of the following Python codes. Then type and run them in Trinket to check your answers.	Procedure 1. Think and write down the output of the following Python codes. Then type and run them in the coding platform to check your answers.
35	Text in illustration: 	Text in illustration: 
70	Text in illustration: <pre>x = 0 while x < 5: print(x)</pre>   (Note: In case of an infinite loop happens, click the stop button  in trinket. If it does not work, reload the page of trinket.)	Text in illustration: <pre>x = 0 while x < 5: print(x)</pre>   (Note: In Python Lab, click  to stop the infinite loop. It may not stop immediately. Reload the page to stop it when needed.)
Project	Text in illustration: 	Text in illustration: 

Appendix

7

Chapter

2

Python Basics and Variables

2.1 Using Python coding platform

Who invented Python and why?

Python was invented in 1991 by Guido van Rossum who is a Dutch programmer. Python codes were designed for easy reading and allows programmers to solve problems with fewer lines.



Fig. 1 The Python logo (two-coloured snakes)

Using Python coding platform

In Scratch, we need the Scratch platform to write our codes. This is the same with Python and there are many online and offline platforms available for writing codes. By using an online platform, we do not need to install any software, and we can even use a tablet or smartphone to do coding. We can use Python Lab from code.org to write Python programs.

Activity 1 Using Python coding platform



Scenario

In this Activity, you will use a Python coding platform.

Learning objectives

- Open a Python coding platform.
- Know how to use the platform.

Procedure

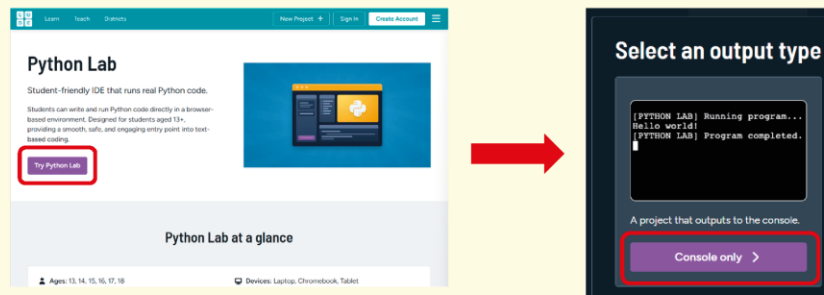
Opening the Python coding platform

01

By using a computer or an iPad, start your browser and open a Python coding platform, e.g. Python Lab, <https://code.org/en-US/tools/python-lab>.

02

Click "Try Python Lab" and choose "Console only".



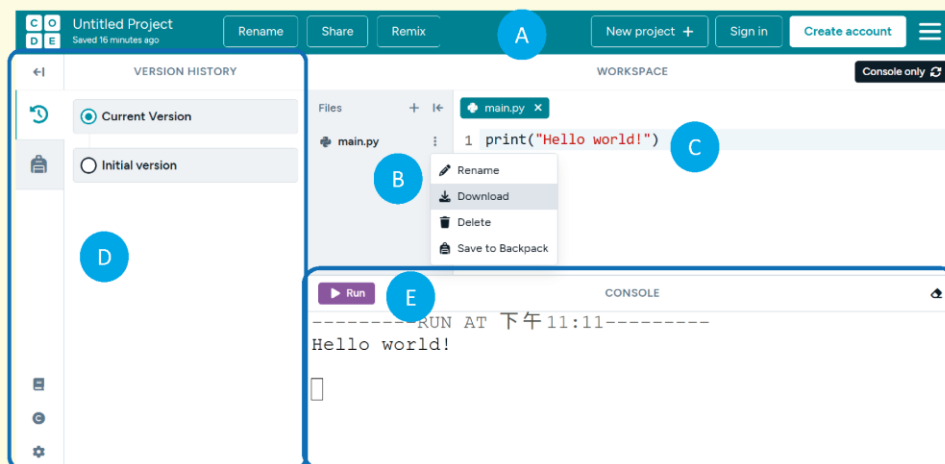
Knowing the user interface

03

After loading, you will see the default "Hello World!" program.

04

There are several areas in the user interface of the platform. Try to explore different functions in the interface.



- A. Menu bar. You can rename, share, remix and create your project. You may create an account and sign in to save your projects in your account.
- B. File manager. Click near the file name and you can rename, download or delete the file. You can click to hide the file manager.
- C. Code editor. This is where you edit your Python code.
- D. Side panel. There are panels with different functions:
 - : Version history – Track the changes of your code
 - : Backpack – Save your code in the backpack
 - : Documentation – Information about Python Lab
 - : Copyright – Copyright notice
 - : Settings – Setting font size, theme colour etc.
- E. Console. This is where you enter your input and see the output of the program. Click to run the program.





2.2 Using the print() function

How do we output text on the screen?

Remember in Scratch, we use the say command to output a text on the screen. In Python, this output is done by the `print()` function and the text has to be enclosed by quotation marks.

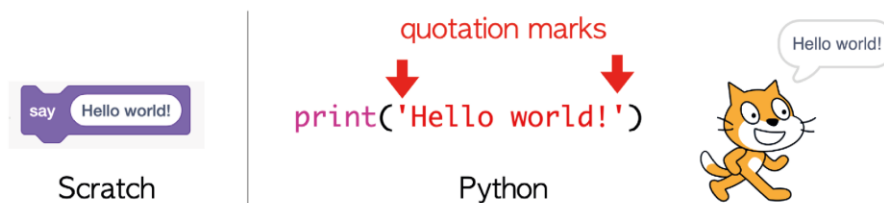


Fig. 2 Commands in Scratch and Python which are used to output a text on the screen.

Activity 2 Displaying a simple text



Scenario

In this Activity, you will try `print()` function.

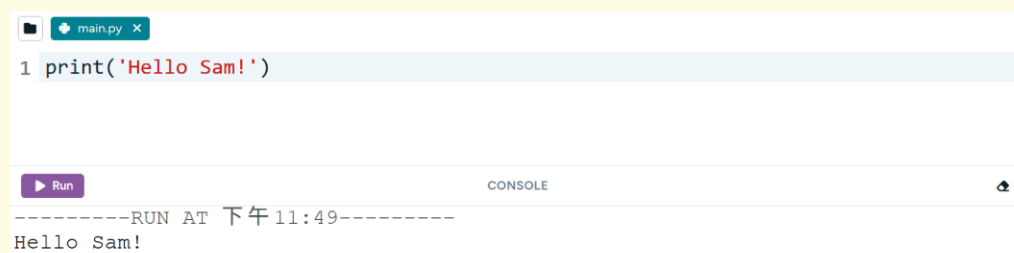
Learning objectives

- Use `print()` function.
- Know common syntax errors.

Procedure

01

In the code editor, type the following Python codes and run it. If there are no typing mistakes, you will see the following in the console:



02

Suppose you make a typing mistake, such as typing `Print` instead of `print`. Try to run the program. What will be displayed?

03

Suppose you make a typing mistake, such as omit the quotation mark. Try to run the program. What will be displayed?



Python is case-sensitive

Syntax error occurs when we do not follow the rules of Python programming. In this case, as Python statements are **case-sensitive**, therefore **Print** is different from **print**.

```

1 Print('Hello Sam!')
Syntax error
-----RUN AT 下午11:57-----
File "/Files/main.py", line 1, in <module>
    Print('Hello Sam!')
    ^^^^^
NameError: name 'Print' is not defined. Did you mean:
    print'?
  
```

Fig. 3 Python is case-sensitive. `Print('Hello world!')` will cause a syntax error as `Print` is different from `print`.

Missing quotation marks for texts (or strings)

Text (or string) in Python must be enclosed by a pair of single or double quotation marks, for example: `'Hello Sam!'` or `"Hello Sam!"`.

```

1 print(Hello Sam!)
Syntax error
-----RUN AT 下午11:58-----
File "/Files/main.py", line 1
    print(Hello Sam!)
          ^
SyntaxError: invalid syntax. Perhaps you forgot a comma?
  
```

Fig. 4 A string must be enclosed by quotation marks, otherwise it will cause a syntax error.

Printing on the same line

In Python, each print statement automatically starts a new line. Python adds a new line character (`\n`) at the end of the string. Therefore,

Python code	Output
<code>print('Hello world!')</code>	Hello world!
<code>print('Hello world!')</code>	Hello world!

Printing multiple strings

To print a string for multiple times, we can use the `*` operator as follows:

Python code	Output
<code>print('Hello world!' * 2)</code>	Hello world!Hello world!

syntax error 語法錯誤

case-sensitive 區分大小寫

Learn more

To print two 'Hello world!' on the same line, we specify the end of string to something else such as a space character (' '):

Python code:

```
print('Hello world!', end=' ')
print('Hello world!')
```

Output:

Hello world! Hello world!

