Hello 🌍🌎🌏

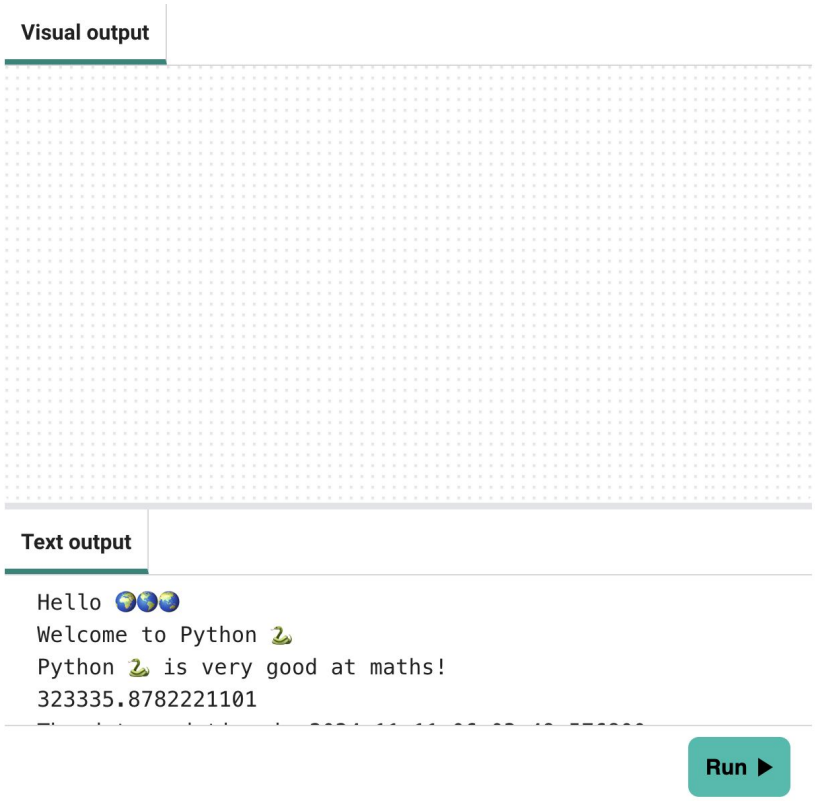
Introduction

This project provides step-by-step instructions to help you learn Python programming.

What you will make

Write an interactive program that uses emojis 🙌 🙌🏻 🙌🏼 🙌🏽 🙌🏾.

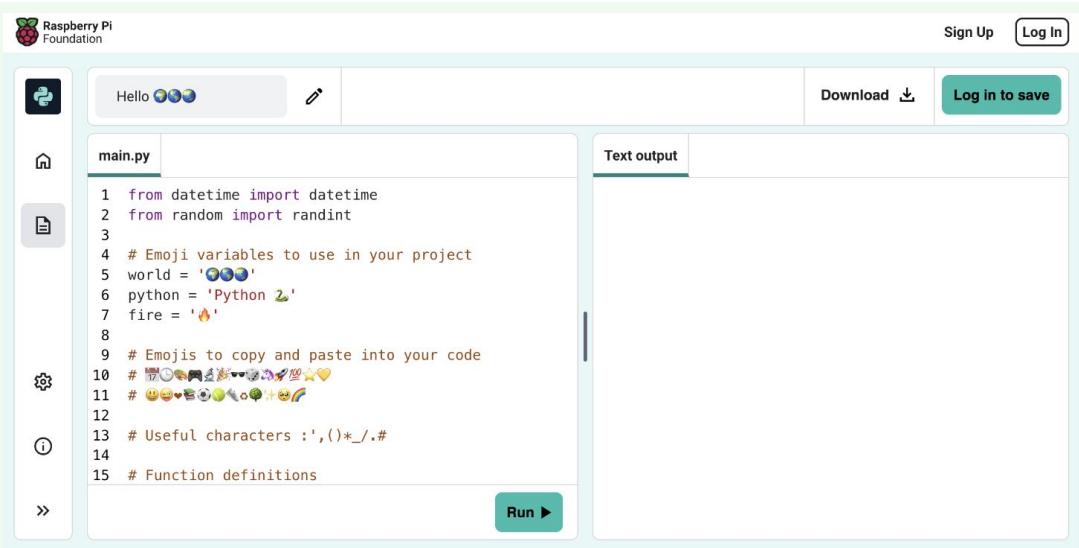
To get an idea of what we will make using this instruction brief, open the URL (rpf.io/pp-py-hw-1) in your browser and click on **Run.**



Print hello

In Python, `print()` outputs strings (words or numbers) to the screen.

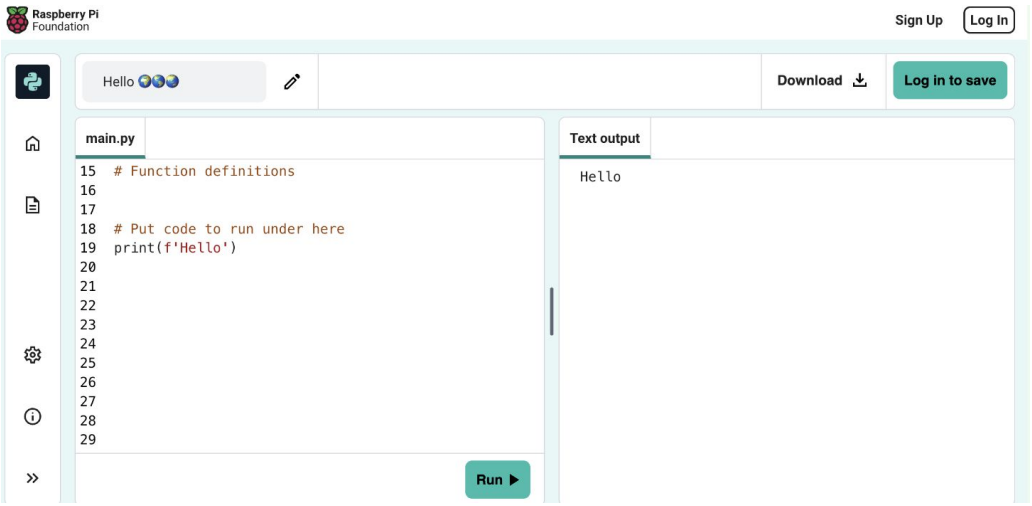
Open the URL (rpf.io/pp-py-hw-2) in your browser to preview the Hello 🌍🌎🌏 starter project.



Find the `# Put code to run below here` line.
Click below that line. The flashing `|` is the cursor and shows where you will type.
Type the code to `print()` Hello to the screen:

```
17 | # Put code to run under here.
18 | print(f'Hello')
```

Test: Click on the **Run** button to run your code. This is what you should see when you run your code.



A **variable** is used to store values such as text or numbers. We have included some variables that store emoji characters.
Change your code to also `print()` the contents of the `world` variable. You can do this by adding the variable name in curly brackets `{}`.

```
17 | # Put code to run under here
18 | print(f'Hello {world}')
```

The `f` character inside the print lets you easily print variables along with strings of text.
Test: Run your code to see the result:

Add another line to your code to `print()` more text and emojis:

```
17 # Put code to run under here
18 print(f'Hello {world}')
19 print(f'Welcome to {python}')
```

Test: Click **Run**.

Tip: It's a good idea to run your code after every change so you can fix problems quickly.

Sums and dates

In Python you can work with numbers and dates.
You can use arithmetic operators such as + and - to do calculations:

+	add
-	subtract
*	multiply
/	divide
**	to the power

Add two more `print()` lines to your code including a multiplication for Python to calculate:

main.py

```
17 # Put code to run under here
18 print(f'Hello {world}')
19 print(f'Welcome to {python}')
20 print(f'{python} is good at maths!')
21 print(f'{3 * 9}')
```

Test: Click the **Run** button.
This is what you should see when you run your code.

```
Hello 🌍🌎🌏
Welcome to Python 🐍
Python 🐍 is good at maths!
27
```

Python has many **modules** that you can use in your code to help perform certain tasks.

The `datetime` module helps with writing code that uses dates and times.
Add another line to your code to `print` the current date and time by using the `now()` method from the `datetime` library:

main.py

```
20 print(f'{python} is good at maths!')
21 print(f'{3 * 9}')
22 print(f'The date and time is {datetime.now()}')
```

Test: Run your code a couple of times to see the time update.

Roll a dice 🎲

Functions are blocks of code that perform specific tasks. They can be used over and over again.
Here is an example of a function:

```
def add_one_and_one():
    x = 1 + 1
    print(x)
```

The name of this function is `add_one_and_one`.

The code for the task you want the function to do needs to be **indented**, which means that you need to add **four spaces** before each line of code.

Calling a function runs the code inside it. You **call** a function by using its name. In this case `add_one_and_one()`.

Look for the comment in the **main.py** file that says

`# Function definitions.`

Create a function called `roll_dice()`, that prints out the number 4.

```
15 # Function definitions
16 def roll_dice():
17     print(f'You rolled a {4}')
18
19 # Put code to run under here
```

Then, call the function at the bottom of your code.

```
24 print(f'The date and time is {datetime.now()}')
25 roll_dice()
```

Test: Run your project several times to see the dice roll each time - it will always be 4.

Another module called `random` can be used to create random numbers.
Change your code to use the `randint` function to choose a random number between 1 and 6 for the dice roll

```
15 # Function definitions
16 def roll_dice():
17     print(f'You rolled a {randint(1, 6)}')
```

Test: Click the Run button.
Now when you run your code, a new random number between 1 and 6 will be chosen each time.

In Python you can multiply strings such as emojis or whole words by a number, so they print out several times.

Change your function to store the random number in a variable called `roll`.

```
15  # Function definitions
16  def roll_dice():
17      roll = randint(1,6)
```

Multiply the random number stored in `roll` by the 🔥 emoji, and print the result.

```
15  # Function definitions
16  def roll_dice():
17      roll = randint(1,6)
18      print(f'You rolled a {roll} {fire * roll}')
```

Test: Click the **Run** button.
Your output code should look something like this:

```
Hello 🌍🌎🌏
Welcome to Python 🐍
Python 🐍 is good at maths!
12345678987654321
The date and time is 2023-11-21 16:14:45.140000
You rolled a 4 🔥🔥🔥🔥
```

Get input

You can use `input()` to ask the person using your program to enter text.

Change your function to ask the person using your program to enter how many sides on the dice, and save it as a variable.

```
15  # Function definitions
16  def roll_dice():
17      max = input('How many sides on your dice?:')
18      print(f'That is a D {max}')
19      roll = randint(1,6)
20      print(f'You rolled a {roll} {fire * roll}')
```

Test: Click the **Run** button and type in a number of sides.
Ensure you press the **Enter** key after inputting how many sides.
This is what you should see when you run your code.

```
Hello 🌍🌎🌏
Welcome to Python 🐍
Python 🐍 is good at maths!
12345678987654321
The date and time is 2023-11-21 16:20:41.323000
How many sides on your dice?:
20
That is a D 20
You rolled a 1 🔥
```

In Python you can multiply strings such as emojis or whole words by a number, so they print out several times.

Change your function to store the random number in a variable called `roll`.

```
15  # Function definitions
16  def roll_dice():
17      roll = randint(1,6)
```

Multiply the random number stored in `roll` by the 🔥 emoji, and print the result.

```
15  # Function definitions
16  def roll_dice():
17      roll = randint(1,6)
18      print(f'You rolled a {roll} {fire * roll}')
```

Test: Click the **Run** button.
Your output code should look something like this:

```
Hello 🌍🌎🌏
Welcome to Python 🐍
Python 🐍 is good at maths!
12345678987654321
The date and time is 2023-11-21 16:14:45.140000
You rolled a 4 🔥🔥🔥🔥
```

How to add emojis?

To add an emoji on **Windows**, follow the steps:

- press Win key + . (period) or Win key + ; (semicolon) to open the emoji panel,
- then click the emoji you want to insert.

To add an emoji on a **Mac**, follow the steps:

- press Control + Command + Spacebar to open the emoji and symbols palette,
- click to add an emoji to your text.

Get input

You can use `input()` to ask the person using your program to enter text.

Change your function to ask the person using your program to enter how many sides on the dice, and save it as a variable.

```
15  # Function definitions
16  def roll_dice():
17      max = input('How many sides on your dice?:')
18      print(f'That is a D {max}')
19      roll = randint(1,6)
20      print(f'You rolled a {roll} {fire * roll}')
```

Test: Click the **Run** button and type in a number of sides.

Ensure you press the **Enter** key after inputting how many sides.

This is what you should see when you run your code.

```
Hello 🌍🌎🌏
Welcome to Python 🐍
Python 🐍 is good at maths!
12345678987654321
The date and time is 2023-11-21 16:20:41.323000
How many sides on your dice?:
20
That is a D 20
You rolled a 1 🔥
```

Inputs are always stored as text, but we need to use the input stored in `max` to specify the largest number that could be rolled.

`max` is a string, so it needs to be changed to an integer `int()`.

```
15  # Function definitions
16  def roll_dice():
17      max = input('How many sides on your dice?:')
18      print(f'That is a D {max}')
19      roll = randint(1, int(max))
20      print(f'You rolled a {roll} {fire * roll}')
```

Test: Click the **Run** button a few times. Check that the dice rolls a random number each time.

Challenge

Practise adding more `print` lines to your code.

Here are some sentence starters that you can use:

```
23  roll_dice()
24  print(f'I ❤ ...')
25  print(f'... makes me 😃')
26  print(f'I would like to make ... with {python}')
```

Here is a list of some emojis you might like to use:

🎊 🙌 🙌🏻 🙌🏼 🙌🏽 🙌🏾 🙌🏿 # 😃 🕒 🎨 🎮 🔬 🎉 🕶 🎲 😊
🦄 🚀 💯 ⭐ 💛 ❤️ 📚 ⚽ 🏏 🏀 🥋 🏆 ✨ 🥲 🌈 🔥 ♻️ 🌳
🧜 🧜🏻 🧜🏼 🧜🏽 🧜🏾 🧘 🧘🏻 🧘🏼 🧘🏽 🧘🏾 🙋 🙋🏻 🙋🏼 🙋🏽 🙋🏾 🙋🏿

Congrats! Project complete!