STMC coding team training

Lesson 1: Hello World

Tsai Yun Chen

February 11, 2024

What we are going to do today?

1. Setup Environment for coding
2. Run your first program `helloworld.ipynb`
3. Understand different parts of `helloworld.ipynb`

Tools we are going to use

- As mentioned we need a compiler to translate our program to machine code
- Unfortunately we do not have compiler installed in our computer yet
- To solve this, we are going to use a online tool named `Google Colab`

What is Google Colab

- Google Colab is a online notebook provided by Google
- allows user to write and run python code online
- It has several benefit over traditional local compiler:
  1. code everywhere
  2. free GPU usage
  3. can share your code to others and work together (like google doc)

What you need to do

Step 1: prepare a Google account

Step 2: Login your Google account and go to your Google drive

Step 3: Open a new folder to hold the files that you will use in this course

Running your first program (I)

Step 1: Download `HelloWorld.ipynb` from course webpage.

Step 2: Upload the file you just downloaded to the Google drive folder

Step 3: double click to open the file when the upload is finished

Running your first program (II)

Now you should see the page below, click the start button beside the code: You should see the following results:

```
1  Hello world!
```

Experiments

Congrats, you just compile your first program. Now, let's explore the function of different parts of the program by doing some experiments:

1. Change "Hello World" to "Bye bye world" and rerun, what do you observe?
2. Similar to the first line, add more `print` to see if you can print multiple lines
3. Change the text behind # in line 1 and recompile, does it change anything about the code? Now try to add # before `print`, what happens?

Explaining helloworld.py: print

- `print` is a function used for printing things
- In `helloworld.py`, `print` is used to print our hello message ”Hello world”
- You can also do something like this:

```
1  print("Text 1", "Text 2", "Text 3")
```

  These text will be separated by spaces (Try it!)

- You can read more about `print` from the documentation

Explaining helloworld.py : Comments

- Those lines after # are called **comments**
- They are ignored by compiler and will not affect how the code run
- Their are notes left by programmers to help himself/herself/others to understand the code
- **For more complicated program, comments are necessary**. Otherwise, code will be very difficult to comprehend and debug

Second step

- Now the computer knows how to talk with you, but we can’t talk to the computer !?
- Next step we are going to communicate with it, slightly click any blank space of the second block
- then you should see the start button appear at the left side similar to what you just did, now click it
- the computer will probably ask you a question, answer it by typing, click enter after your input and see what it answer.

Explaining helloworld.py : Input

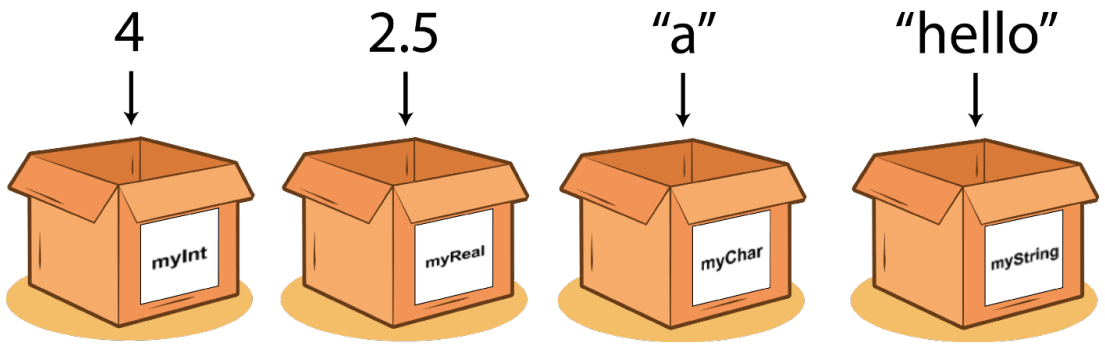
- As you can probably see, `input` ask you a question which you can answer
- You answer by typing some input and press enter to submit
- Furthermore, the text inside `input` is displayed when it prompts for answer
- So in our program you see the following line is acutally asking your name and waiting for you input:

```
1       input("What is your name? ")
```

- But here is a problem: How can we save and manipulate these input?

Variables - Your handy storage box

- Imagine that if you never remember things, then how can you answer questions?
- To store data, a computer uses something called a **variable**
- A variable is a ”storage box” with a name
- It stores data temporarily so that the value inside can be retrieved for further processing



Source: https://stevenpcurtis.medium.com/what-is-a-variable-3447ac1331b9

Explaining helloworld.py : Input

- So back to our example, we see that the full line of our program is actually

```
1       name=input("What is your name? ")
```

- now we know that name is a variable that stores your input, which is your name.
- Further we see the line follows,

```
1       print("Hi,",name)
```

- here we are trying to response to what user input
- so refering to our experiment before we can expect the response is what we expect in the comment: `Hi, [your_name]`

More experiments

Let’s try out something more:

- Change the variable `name` to sth else, say your favorite character or song or etc, does the code still works?
- try to input something else when answering the question, like entering number, what you expect to get?
- can youy modify the code to ask more question? thens print out all the information you gather in one line.