TEMASEK
JUNIOR COLLEGE

Temasek Junior College
2022 JC1 H2 Computing
Practical 5 – User Interactivity via `Input()`

Session Objectives

By the end of this session, you will learn:

(i) how to collect user input.

(ii) write simple programs that collect and process user input.

§5.1 The `input()` Function

The `input()` function can be used to collect information from the program user. A statement is usually used in the input function to prompt the user on the type of input that the program expects. In addition, a variable is often assigned to store the user input for ease of processing.

When the `input()` statement in a program is executed, it pauses to wait for an input from the user before continuing.

In Python, user input will be interpreted as a string.

Exercise 1

Enter and execute the following code into the REPL. Observe what happens.

```
your_age = input('How old are you? ')
your_age
type(your_age)
```

Notice that there is a spacing between the question mark `?` and the closing single quotation mark `'` in the input statement. This is to provide a spacing between the input statement displayed when the program runs. This improves readability and formatting.

Exercise 2

Enter and execute the following code into the Python file editor and run it.

```
your_age = int(input('How old are you? '))
age_in_10_years = your_age + 10
print("10 years later you will be", age_in_10_years)
```

Since the user input is interpreted as a string by default, if you would like to convert the user input to another type, such as `int` type, you will need to use `int()` as shown in Exercise 2 to convert the string value of `your_age` to integer value.

Exercise 3

Enter and execute the following code into the Python file editor and run it.

```
your_age = eval(input('How old are you? '))
age_in_10_years = your_age + 10
print("10 years later you will be", age_in_10_years)
```

You may choose to use `eval()` to evaluate the given source, which in this case is the string that the user input. If the user key in 5, it will be evaluated as an `int` type with value 5. If the user key in 5.0, it will be evaluated as an `float` type with value 5.0.

Tutorial

Problem 1 – Greeting Program

Implement a simple greeting program that will:

- Allow the user to input his / her name on the line below a prompt statement.
- Print out a customised greeting message with the user's name.

**You are reminded to use meaningful identifiers and identifier conventions adhering to PEP 8.*

Sample Output

```
What is your name?
Python
Hello Python!
```

* Ensure that the output is exactly the same as the sample output for the user Python.

Save this Python file as **problem1.py** and submit it on Google Classroom.

Problem 2 – Area and Perimeter Calculator for Rectangles

Implement a program that calculates the area and perimeter of a rectangle. The program will

- Ask the user for the length of the rectangle.
- Ask the user for breadth of the rectangle.
- The user should enter the values on the same line as the prompt statements, 1 spacing away.
- Calculate and print the area and perimeter of the rectangle.

Sample Output

```
Enter rectangle's length: 7
Enter rectangle's breadth: 5
Rectangle's perimeter: 24.0
Rectangle's area: 35.0
```

* Ensure that the output is exactly as what you see in the sample output for the given values.

Save this Python file as **problem2.py** and submit it on Google Classroom.

Possible Extensions

- Check the validity of the inputs.
- Calculate the perimeter and area of the shape of user's choice.
- Feel free to suggest your own extension and include it in your comments in the code.

Problem 3 – Basic BMI Calculator

Implement a simple greeting program that will:

- Allow the user to input his / her responses on the same line as the prompt statements, 1 spacing away.
- Ask the user for his or her name, height in metres and mass in kilograms.
- Calculate and display the BMI of the user.

The formula for BMI is:

$$\text{Mass in kg} \div (\text{height in m})^2$$

Sample Output

```
Enter your name: Python
Enter your mass in kilograms: 70
Enter your height in metres: 1.68
Hi Python, your BMI is 24.801587301587304.
```

Ensure that the ouput is exactly as what you see in the sample output for the given values.

Save this Python file as **problem3.py** and submit it on Google Classroom.

Possible Extensions

- Check the validity of the inputs.
- Display the BMI category based on the BMI value. (research online on how BMI is categorised)
- Feel free to suggest your own extension and include it in your comments in the code.