Taking User Input

Developers often need to interact with users, either to get data or to provide some sort of result.

How to take User Input?

To get the input from the user interactively, we can use the built-in function, `input()`. This function is used in the following manner:

```
variable_to_hold_the_input_value = input(<Prompt to be displayed>)
```

For example:

```
In[] : age = input("What is your age?")
```

The above statement will display the prompt as:-

```
What is your age?_________ ←{User input here}
```

We will get the following interactive output:

```
In[] : name = input("Enter your name: ")
Enter your name: Rishabh #User Input
In[] : age = input("Enter your age: ")
Enter your age: 20 #User Input
In[] : name
Out[] : 'Rishabh'
In[] : age
Out[] : '19'
```

Note:- `input()` function always returns a value of the **String** type. Notice that in the above script the output for both name and age, Python has enclosed the output in quotes, like `'Rishabh'` and `'19'`, which implies that it is of **String** type. This is just because, whatever the user inputs in the `input()` function, it is treated as a **String**. This would mean that even if we input an integer value like 20, it will be treated like a string `'19'` and not an integer. Now, we will see how to read Numbers in the next section.

Reading Numbers

Python offers two functions `int()` and `float()` to be used with the `input()` function to convert the values received through `input()` into the respective numeric types integer and floating-point numbers. The steps will be:-

1. Use the `input()` function to read the user input.
2. Use the `int()` and `float()` function to convert the value *read* into integers and floating-point numbers, respectively. This process is called **Type Casting**.

The general way of taking Input:

```
variableRead = input(<Prompt to be displayed>)
updatedVariable = int(variableRead)
```

Here, `variableRead` is a String type that was read from the user. This string value will then be converted to Integer using the `int()` function and assigned to `updatedVariable.`

This can even be shortened to a single line of code as shown below:-

```
updatedVariable = int(input(<Prompt to be displayed>))
```

Let us take an example:-

```
In[] : age= int(input("Enter Your Age: "))
Enter Your Age: 19
In[] : age
Out[] : 19
```

Here, the output will be `19` and not `'19'`, i.e. the output is an Integer and not a String. Similarly, if we want to read a floating-point number, we can do the following:-

```
In[] : weight= float(input("Enter Your Age: "))
Enter Your Weight: 65.5
In[] : weight
Out[] : 65.5
```