

Introduction to Programming Language Python

Overview

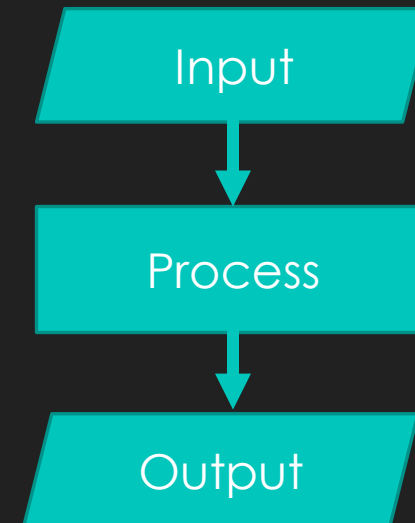
- Introduction to Programming and Python
- Setup environment Python in your computer
- Hello World
- Calculator in Python
- Commenting
- Variable in Python
- String in Python
- Taking an input from user
- Selection
- Looping

Introduction to Programming

- Programming is an algorithm(steps) that has been write using programming language, so that it can be executed by a computer.
- Example of popular Programming Language:
 - Python
 - C++
 - C
 - Java
 - PHP

Introduction to Programming (Cont.)

- To get a simple idea about programming is it is a step by step to solve a problem.
- Basic procedure in programming is:
 - Input
 - Process
 - Output



Introduction to Python

- Python is an easy to learn, powerful programming language.
- It has efficient high-level data structures and a simple but effective approach to object-oriented programming.
- Suitable for beginner to get into programming.

Setup environment Python in your computer

1. Download the installer from official Python websites:
 - <https://www.python.org/downloads/release/python-361/>
 - Click at a link named Windows x86 executable installer
2. If already have, then double click the installer. Name of the installer is something like this "python-3.6.1.exe"

Setup environment Python in your computer

3. This window will pop up after you double click the installer
4. Check the radio button “Add Python 3.6 to Path” (in red rectangle shape)
5. Then click Install Now



Setup environment Python in your computer

6. Wait until the installation finished
7. Click close button

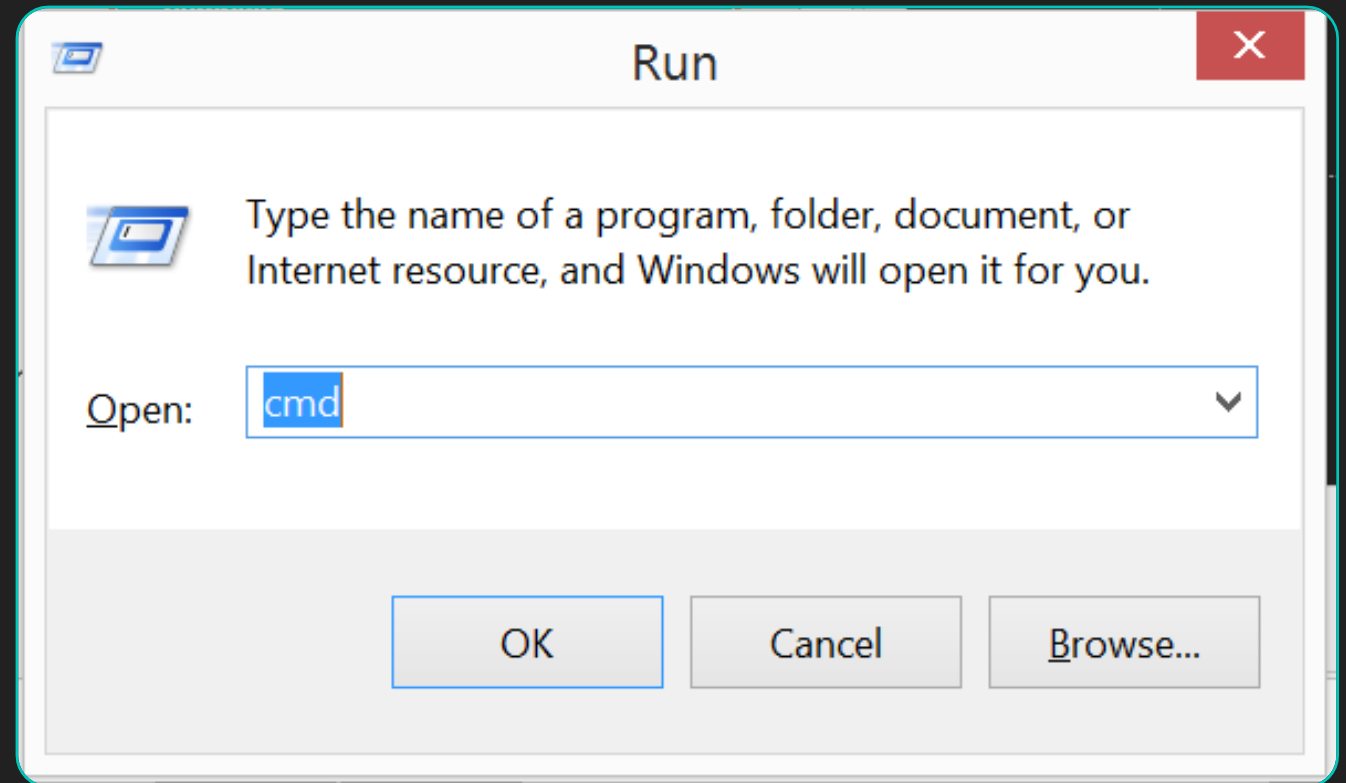


Setup environment Python in your computer

8. Next open command prompt, by type on your keyboard Start + R

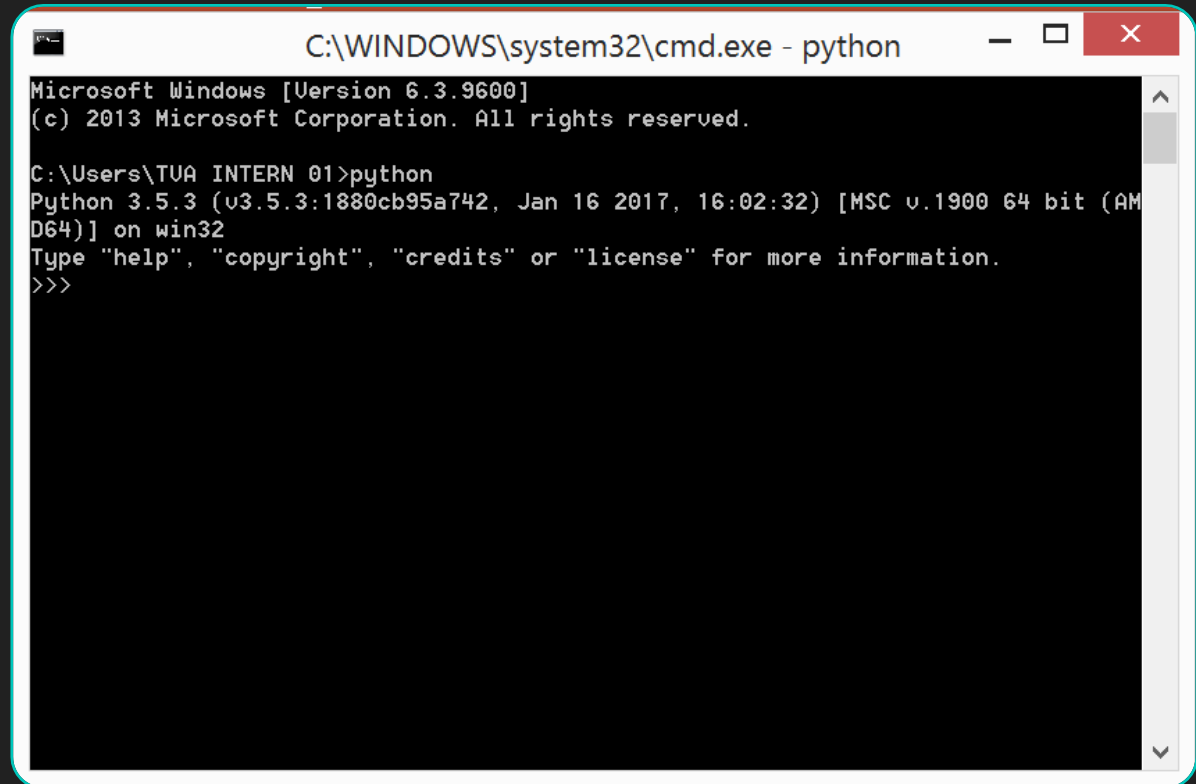
9. Then type cmd and click Okay.

p/s: Other method that will work is go to search, and type CMD, then click on the result named 'Command Prompt'



Setup environment Python in your computer

10. Command Prompt will popup.
11. Type 'python' and it will display text like in the picture
12. Congratulations! You success install Python in your computer



```
C:\WINDOWS\system32\cmd.exe - python
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\TUA INTERN 01>python
Python 3.5.3 (v3.5.3:1880cb95a742, Jan 16 2017, 16:02:32) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Setup environment Python in your computer

Installing PyCharm

1. If you using Windows, download <https://www.jetbrains.com/pycharm/download/download-thanks.html?platform=windows&code=PCC>
2. If you using Mac, download <https://www.jetbrains.com/pycharm/download/download-thanks.html?platform=mac&code=PCC>
3. If you using Linux, download <https://www.jetbrains.com/pycharm/download/download-thanks.html?platform=linux&code=PCC>



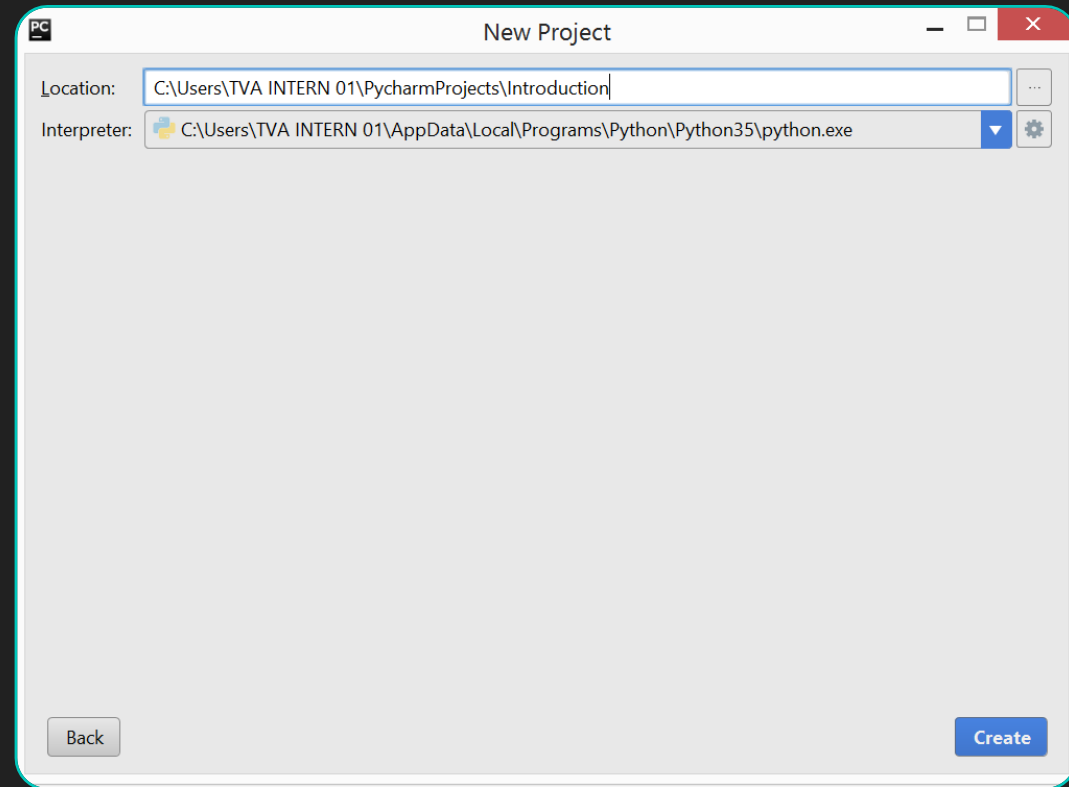
Setup environment Python in your computer

1. Install it as usual. Just click Next and Okay until finish install.
2. Open PyCharm, and just click okay if a dialog popup for configuration.
3. Then you will see the front page of PyCharm like in the picture]
4. Click *Create New Project*



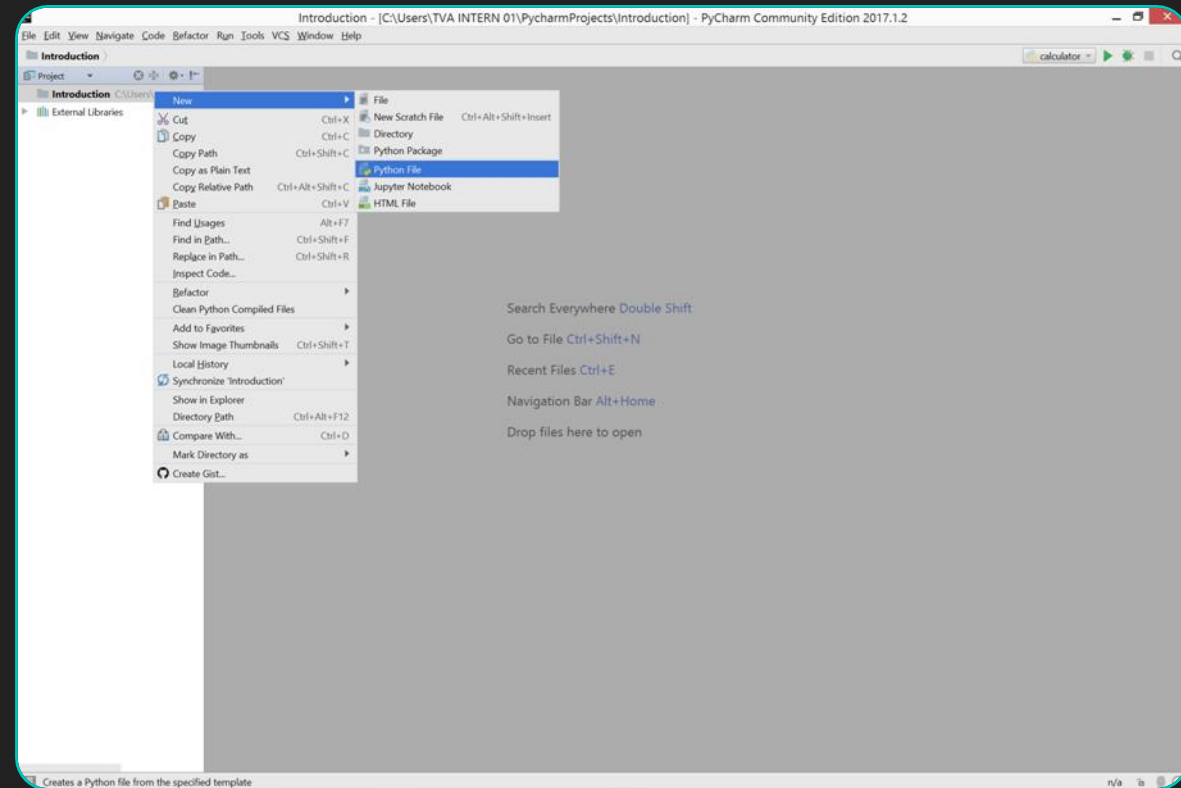
Setup environment Python in your computer

4. Type Introduction as name of your project
5. Click *Create*



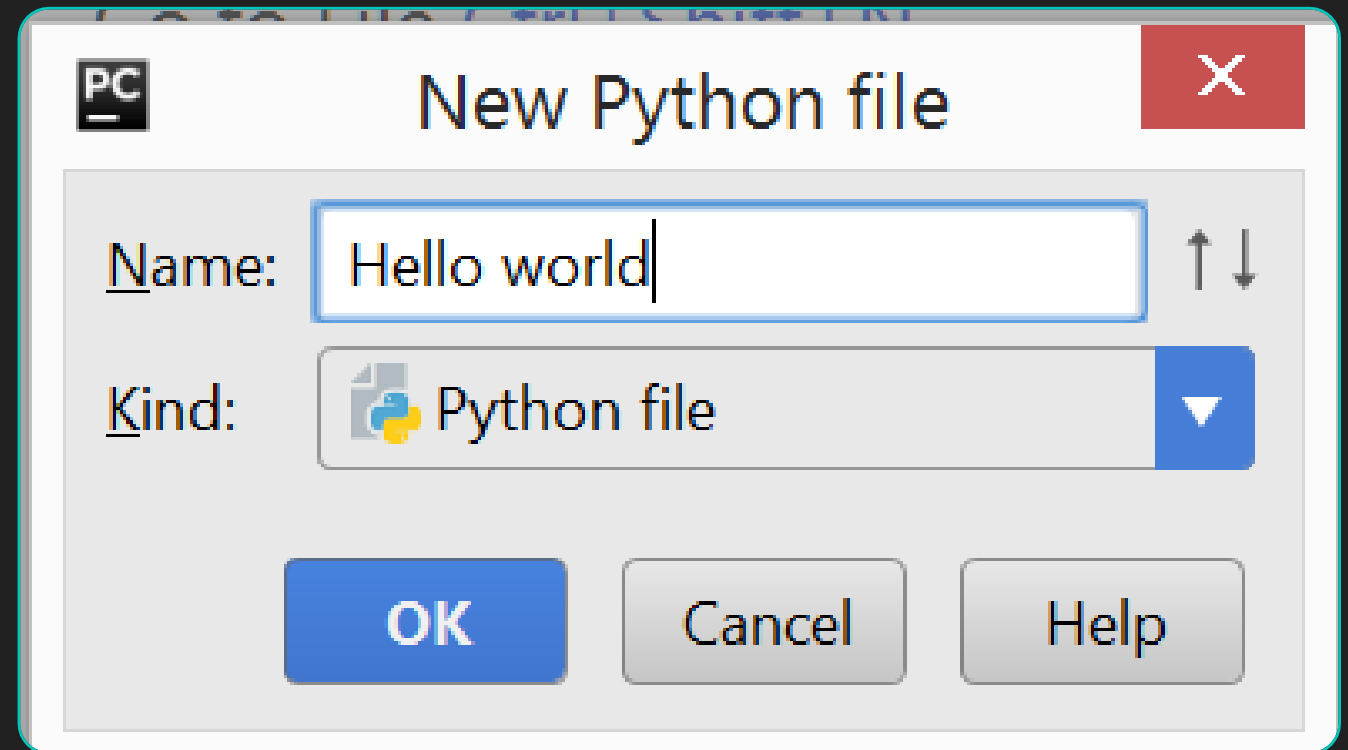
Setup environment Python in your computer

4. Right click on the Introduction Project
5. Choose New
6. Click Python File



Setup environment Python in your computer

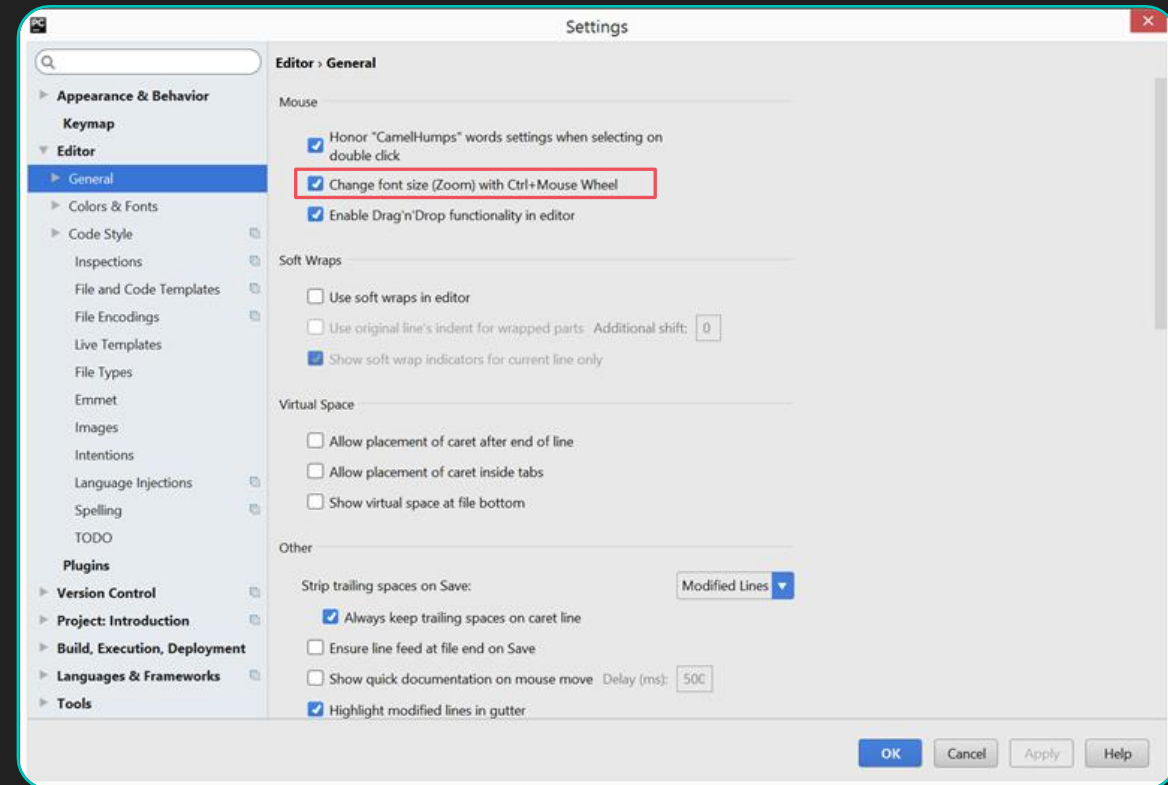
1. Type 'Hello world' and make sure Kind textbox is *Python file*
2. Click OK



Setup environment Python in your computer

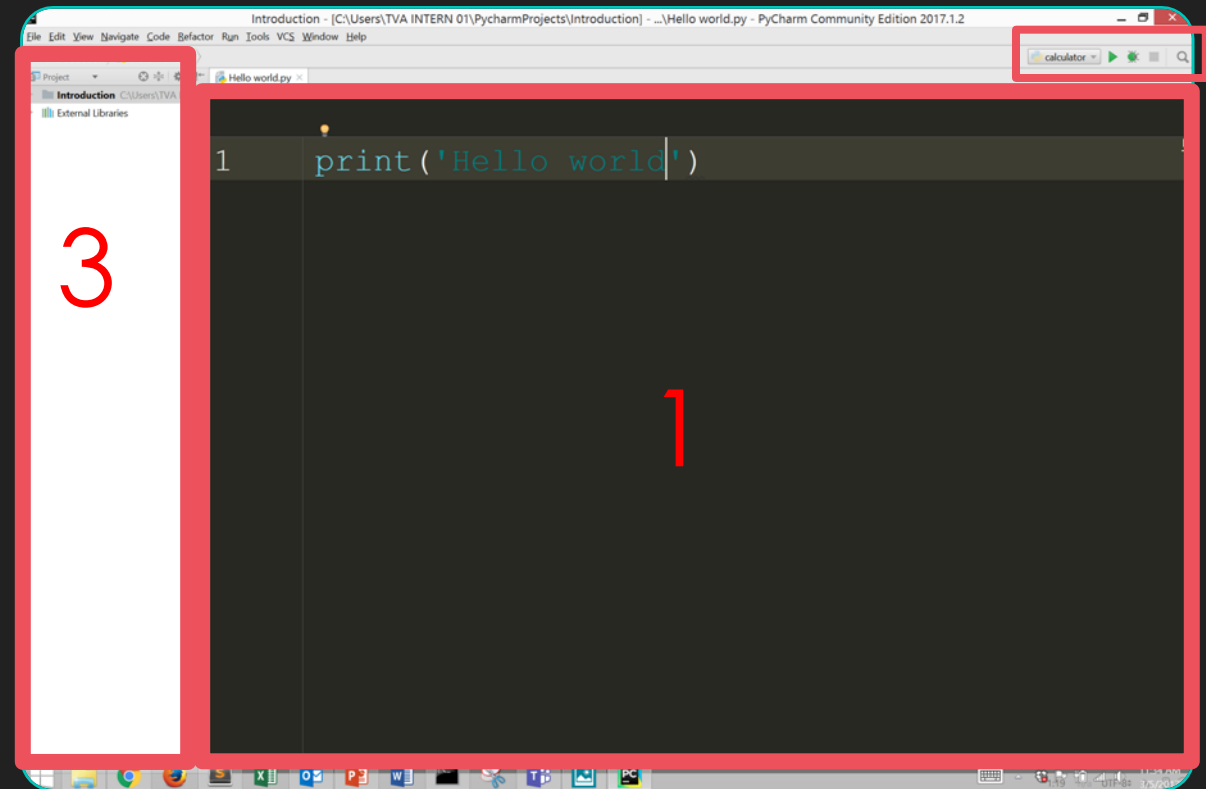
If you find that the font is too small, then you need to resize it

1. Click on the File -> Settings or just simply press Ctrl + Alt + S
2. Go to Editor -> General
3. Check the radio button 'Change font size (zoom) with Ctrl + Mouse Wheel'
4. Click OK
5. Then you just need to press Ctrl + Mouse Wheel to resize the font



Setup environment Python in your computer

- So as you can see, the font is much bigger now
- Here is the simple tour before you start learn python:
 1. Called Editor, where you will write your Python code here
 2. Called Run toolbar, where you will click the 'Play' button to run the code
 3. List directory and file in PyCharm project

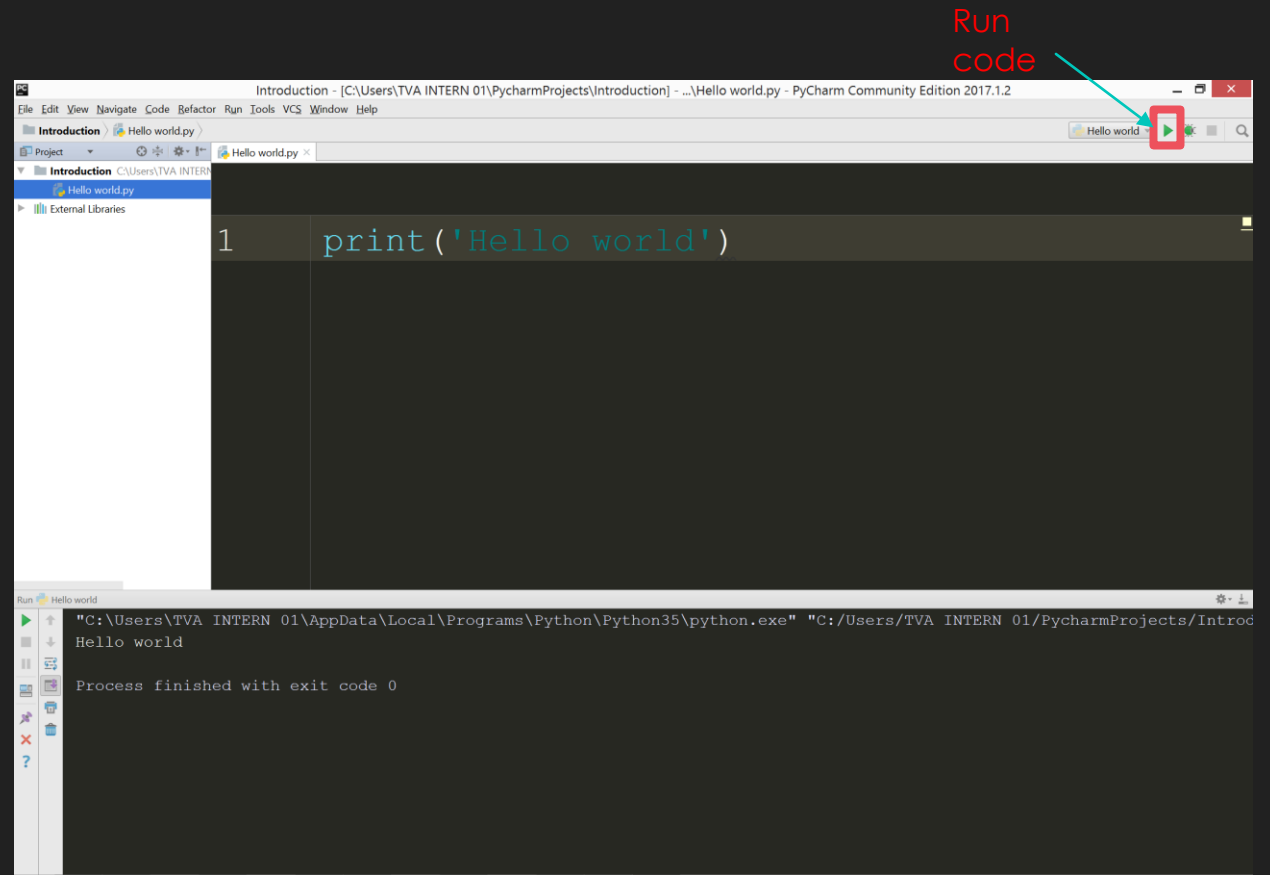


Hello World

- Simple command in all programming language tutorial is `print 'Hello world!'`
- In Python, to display output, the syntax is `'print'` followed by open and close bracket.
- Example:

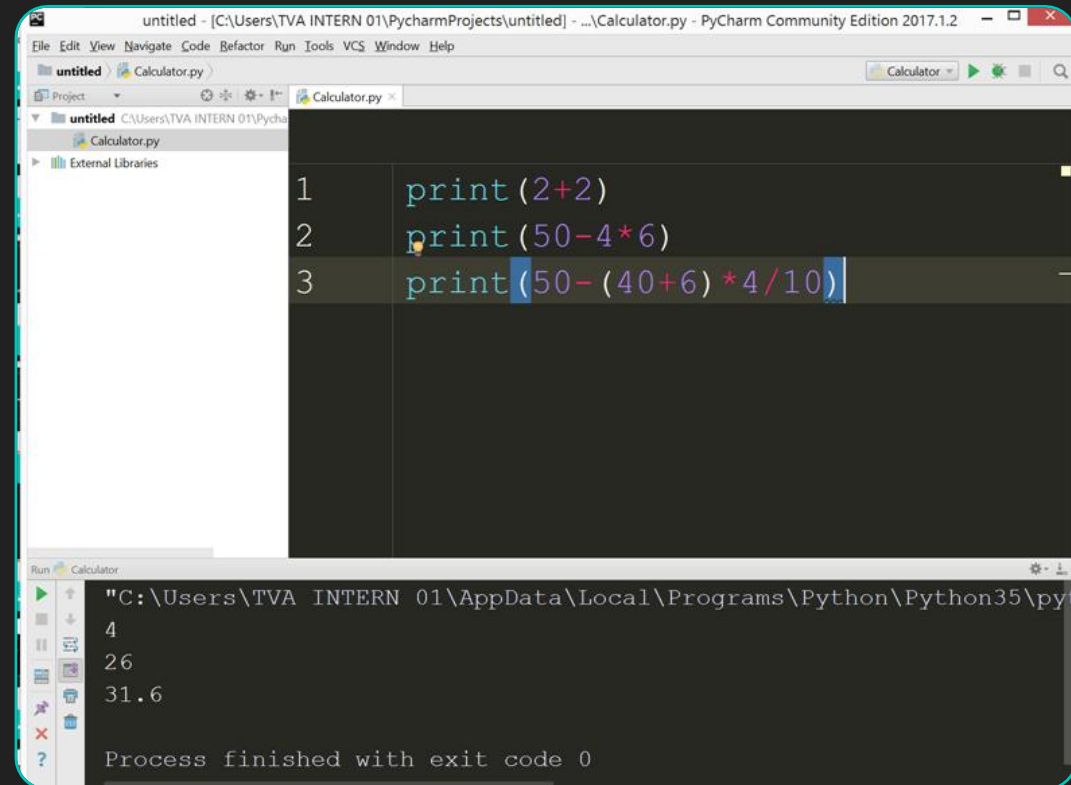
```
print('Hello world')
```

Output : Hello world
- Click the play button on the top right hand side to run the code
- The output is at the below window



Calculator in Python

- Create another file named 'Calculator'
- You can do calculation by straight away right the formula
- In most programming language:
 - + symbol is plus/add
 - - symbol is minus/subtract
 - / symbol is divide
 - * symbol is multiply
- Example:
 - 2 + 2
 - Output: 4
 - 50 – 4*6
 - Output: 26
 - 50 – (40 + 6) * 4 / 10
 - Output: 31.6



The screenshot shows the PyCharm IDE interface. The main editor window displays a Python file named 'Calculator.py' with the following code:

```
1 print(2+2)
2 print(50-4*6)
3 print(50-(40+6)*4/10)
```

The Run tool window at the bottom shows the output of the program:

```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\pyt
4
26
31.6

Process finished with exit code 0
```

Calculator in Python (Cont.)

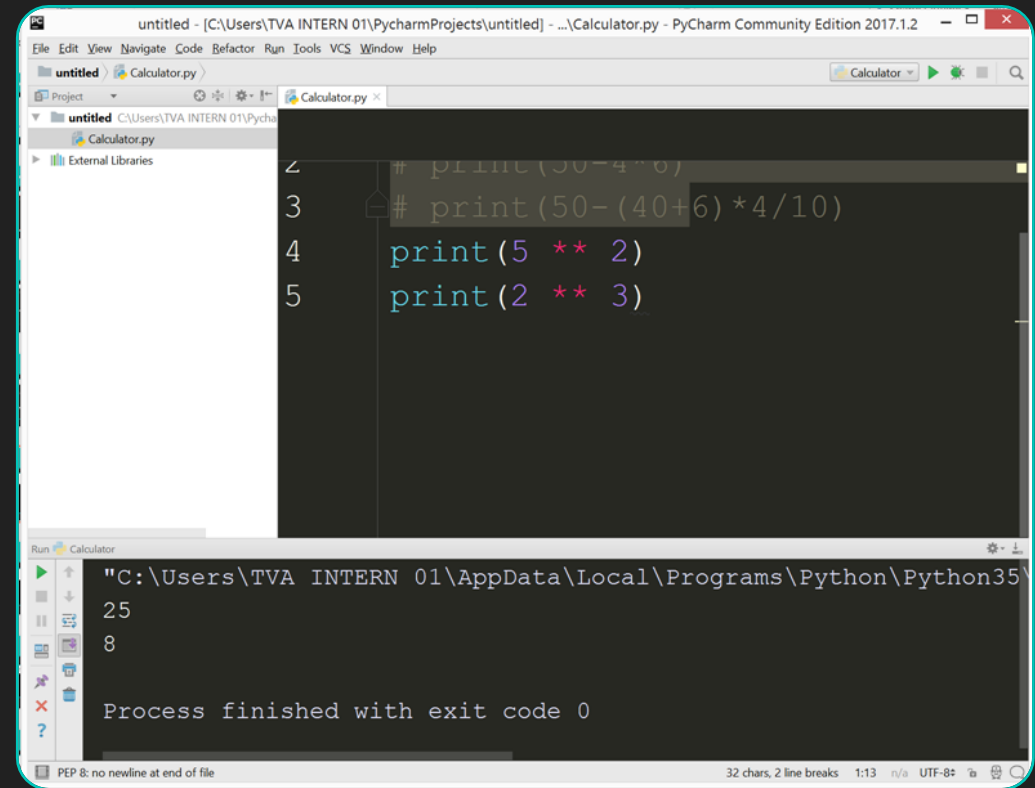
- Calculate power, using double asterisk `**`
- For example we want to calculate 5^2

`5 ** 2`

Output: 25

`2 ** 3`

Output: 8



The screenshot shows the PyCharm IDE interface. The main editor window displays a Python script named `Calculator.py` with the following code:

```
2 # print(50-4*6)
3 # print(50-(40+6)*4/10)
4 print(5 ** 2)
5 print(2 ** 3)
```

The Run tool window at the bottom shows the output of the script:

```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\
25
8
Process finished with exit code 0
```

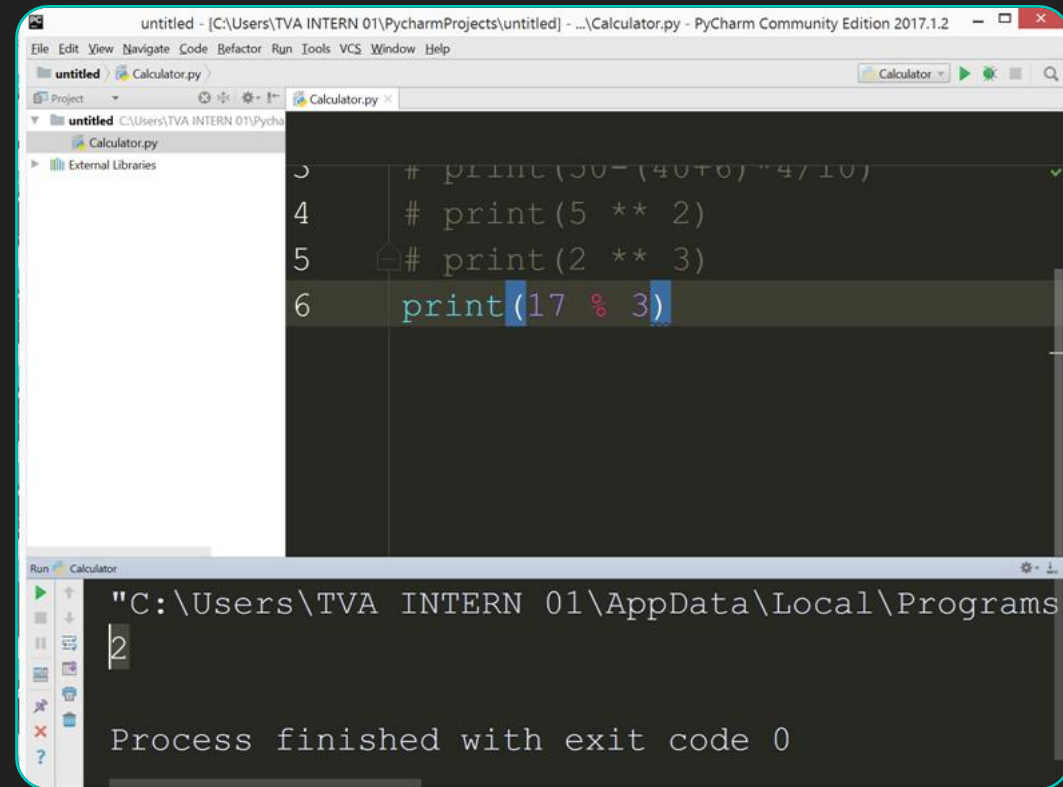
The status bar at the bottom indicates "PEP 8: no newline at end of file" and "32 chars, 2 line breaks 1:13 n/a UTF-8".

Calculator in Python (Cont.)

- Calculate remainder of the division using symbol percent '%'
- For example we want to calculate remainder of 17 divide by 3

$17 \% 3$

Output: 2



The screenshot shows the PyCharm IDE interface. The main editor window displays a Python script named 'Calculator.py' with the following code:

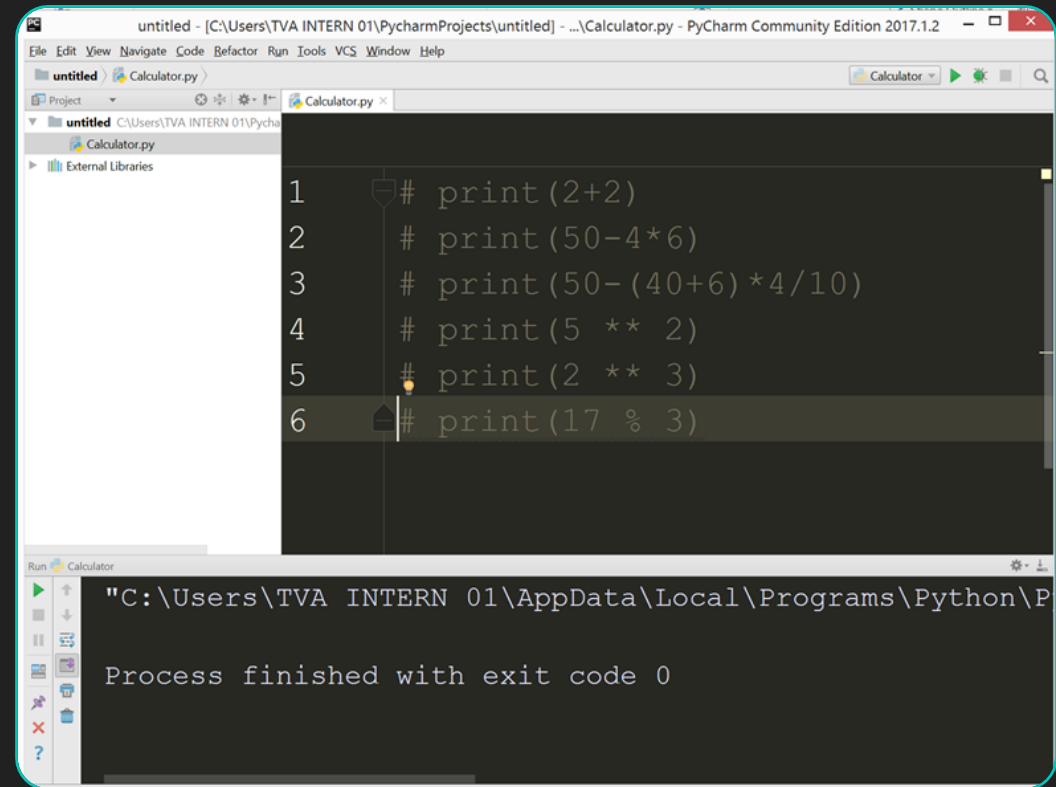
```
3 # print(50 - (40 + 6) * 4 / 10)
4 # print(5 ** 2)
5 # print(2 ** 3)
6 print(17 % 3)
```

The line `print(17 % 3)` is highlighted. Below the editor, the 'Run' console shows the output of the script:

```
"C:\Users\TVA INTERN 01\AppData\Local\Programs
2
Process finished with exit code 0
```

Commenting

- In computer programming, a comment is a programmer-readable explanation or annotation in the source code of a computer program. They are added with the purpose of making the source code easier for humans to understand, and are generally ignored by compilers and interpreters.
- In Python, # or hashtag is used to declare it as a comment.
- For example:
 # print(2 + 2)
 Output: Nothing because it is commented



The screenshot shows the PyCharm IDE interface. The main editor window displays a Python file named 'Calculator.py' with the following code:

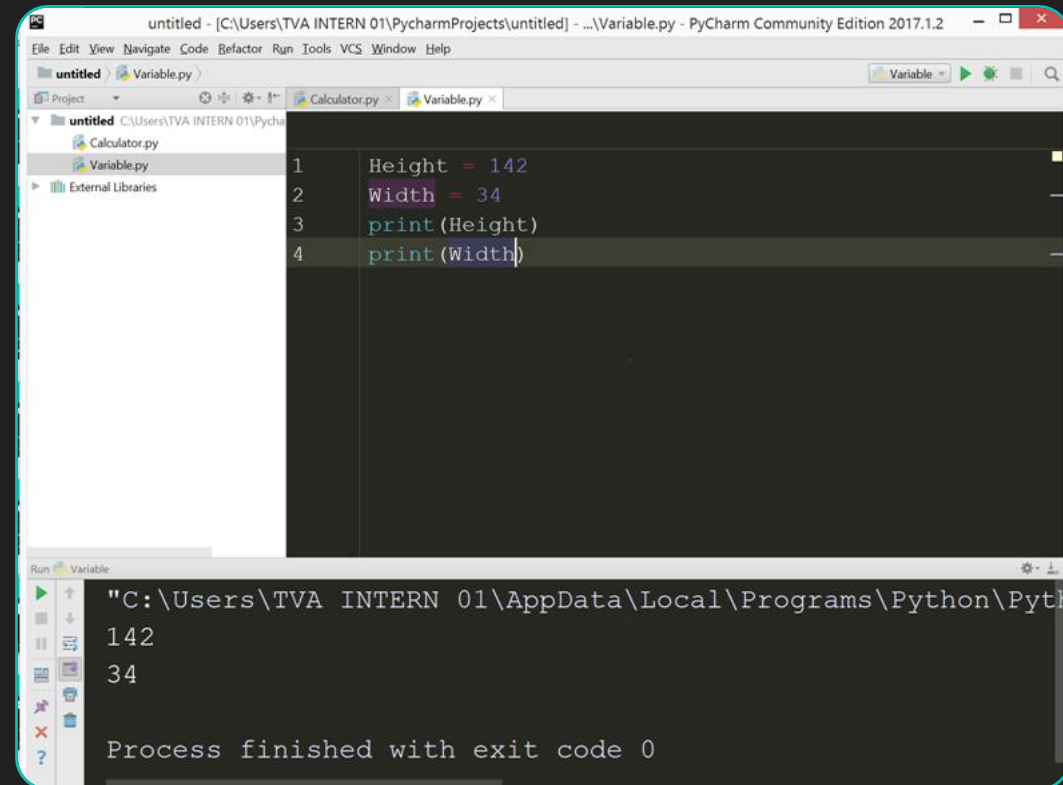
```
1 # print(2+2)
2 # print(50-4*6)
3 # print(50-(40+6)*4/10)
4 # print(5 ** 2)
5 # print(2 ** 3)
6 # print(17 % 3)
```

The left sidebar shows the project structure with 'Calculator.py' selected. The bottom terminal window shows the execution path:

```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python37-32\python.exe"
Process finished with exit code 0
```

Variable in Python

- The problem with previous example is the value calculated is not stored in the memory. So if you want to use it again, then you need to type the formula again.
- To overcome this problem, variable is used.
- Assign the value to the variable.
- For example:
Height = 142
Width = 34
- The variable is Height with value 142, second variable is Width with value 34



The screenshot shows the PyCharm IDE interface. The main editor window displays a Python script named 'Variable.py' with the following code:

```
1 Height = 142
2 Width = 34
3 print(Height)
4 print(Width)
```

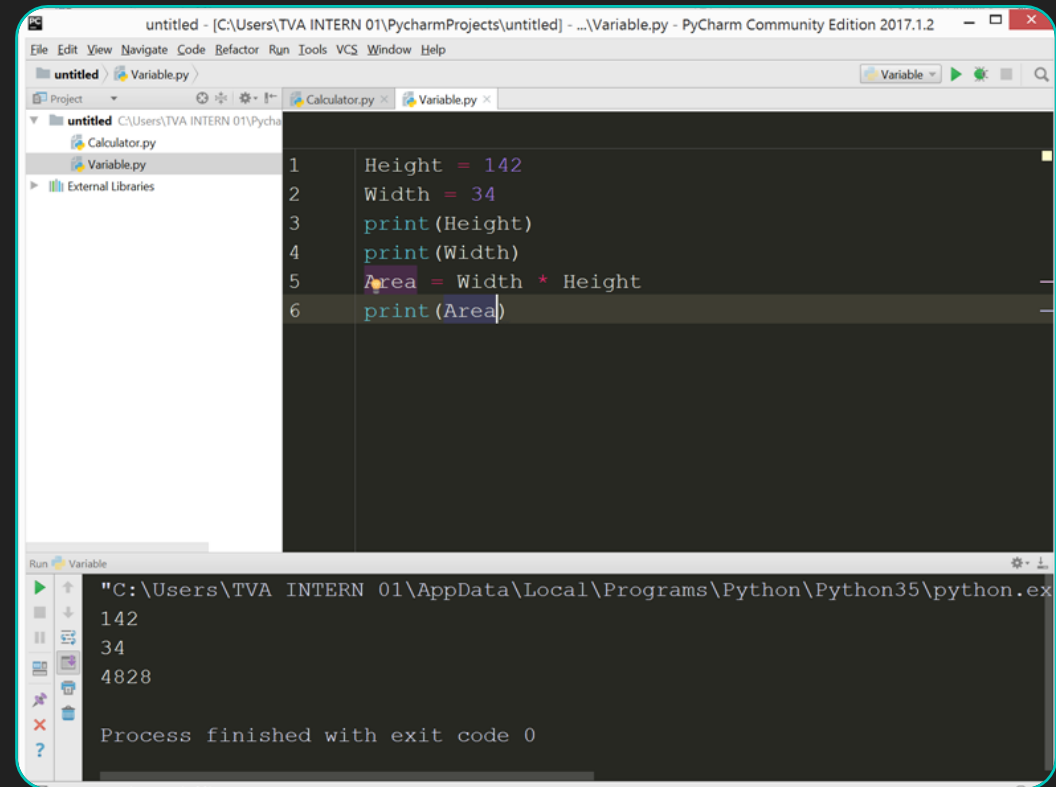
The left sidebar shows the project structure with 'untitled' and 'Variable.py' files. The bottom panel shows the 'Run' output, indicating the script was executed successfully and the output was:

```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python38\python.exe" C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python38\python.exe C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python38\python.exe
142
34
Process finished with exit code 0
```


Variable in Python (Cont.)

- You also can assign a formula to a variable
- For example:
$$\text{Area} = \text{Height} * \text{Width}$$

`print(Area)`
Output:
- Example above shows that, variable Height and Width is used to calculate area. The value is stored in variable Area.



The screenshot shows the PyCharm IDE interface. The main editor window displays a Python script in `Variable.py` with the following code:

```
1 Height = 142
2 Width = 34
3 print(Height)
4 print(Width)
5 Area = Width * Height
6 print(Area)
```

The left sidebar shows the project structure with `untitled` and `Variable.py` selected. The bottom panel shows the Run output, indicating the script was executed successfully and produced the following output:

```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\python.exe
142
34
4828
Process finished with exit code 0
```


Exercise for Variable

01

Show the calculation of BMI using Python

02

Formula for BMI:

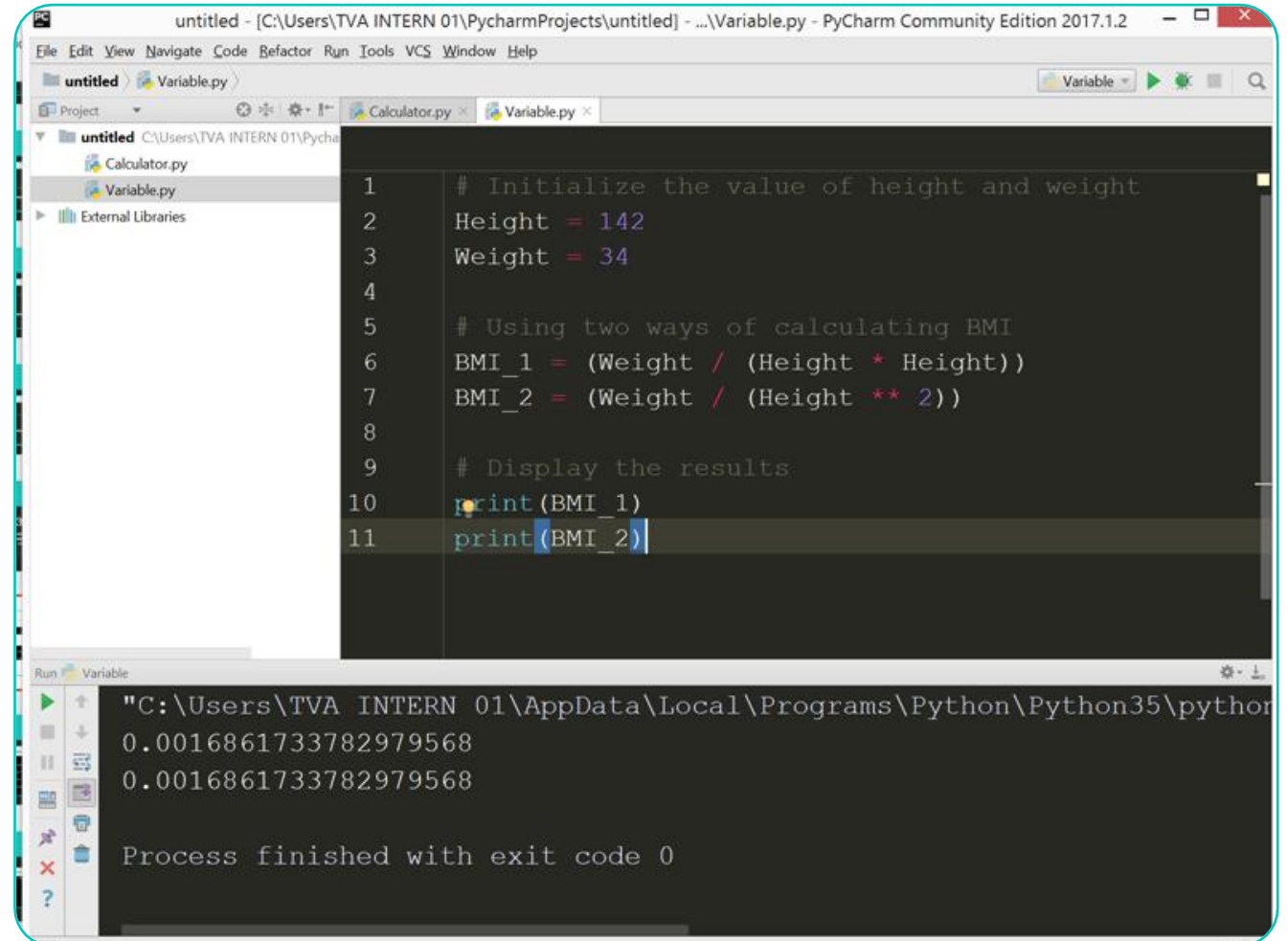
- $BMI = (Weight / (Height \times Height))$

03

All values must be assigned to the variables

Answer for Exercise

- Assign the value to Height and Weight variable
- Then make a formula and assign it to BMI variable
- There are many ways you can do to calculate BMI as shown in the picture



The screenshot shows the PyCharm IDE interface. The main editor window displays a Python script named `Variable.py` with the following code:

```
1 # Initialize the value of height and weight
2 Height = 142
3 Weight = 34
4
5 # Using two ways of calculating BMI
6 BMI_1 = (Weight / (Height * Height))
7 BMI_2 = (Weight / (Height ** 2))
8
9 # Display the results
10 print(BMI_1)
11 print(BMI_2)
```

The left sidebar shows the project structure with `untitled` and `Variable.py` files. The bottom panel shows the Run output, displaying the calculated BMI values for both methods:

```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\python
0.0016861733782979568
0.0016861733782979568

Process finished with exit code 0
```

Strings in Python

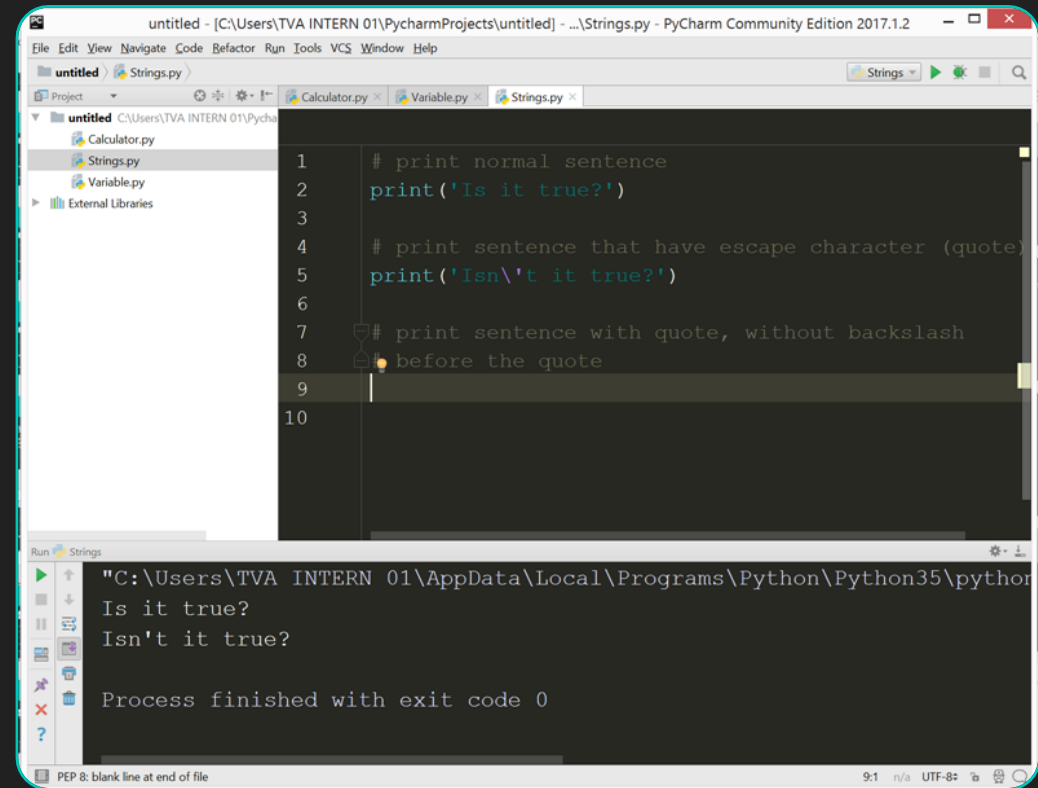
- Strings is enclosed in quotes and special character are escaped with backslashes.
- For example:

```
print('Is it true?')
```

Output: Is it true?

```
Print('Isn\'t it true?')
```

Output: Isn't it true?
- There are two examples given above. First example is easy to understand. The second example is different from the first one, where it has backslash before the quotes.
- It means that the quote is a special character, so you need to use backslash in order to display it. If you don't put the backslash, then it will show an error.



The screenshot shows the PyCharm IDE interface. The main editor window displays a Python script with the following code:

```
1 # print normal sentence
2 print('Is it true?')
3
4 # print sentence that have escape character (quote)
5 print('Isn\'t it true?')
6
7 # print sentence with quote, without backslash
8 before the quote
9
10
```

The Run console at the bottom shows the output of the script:

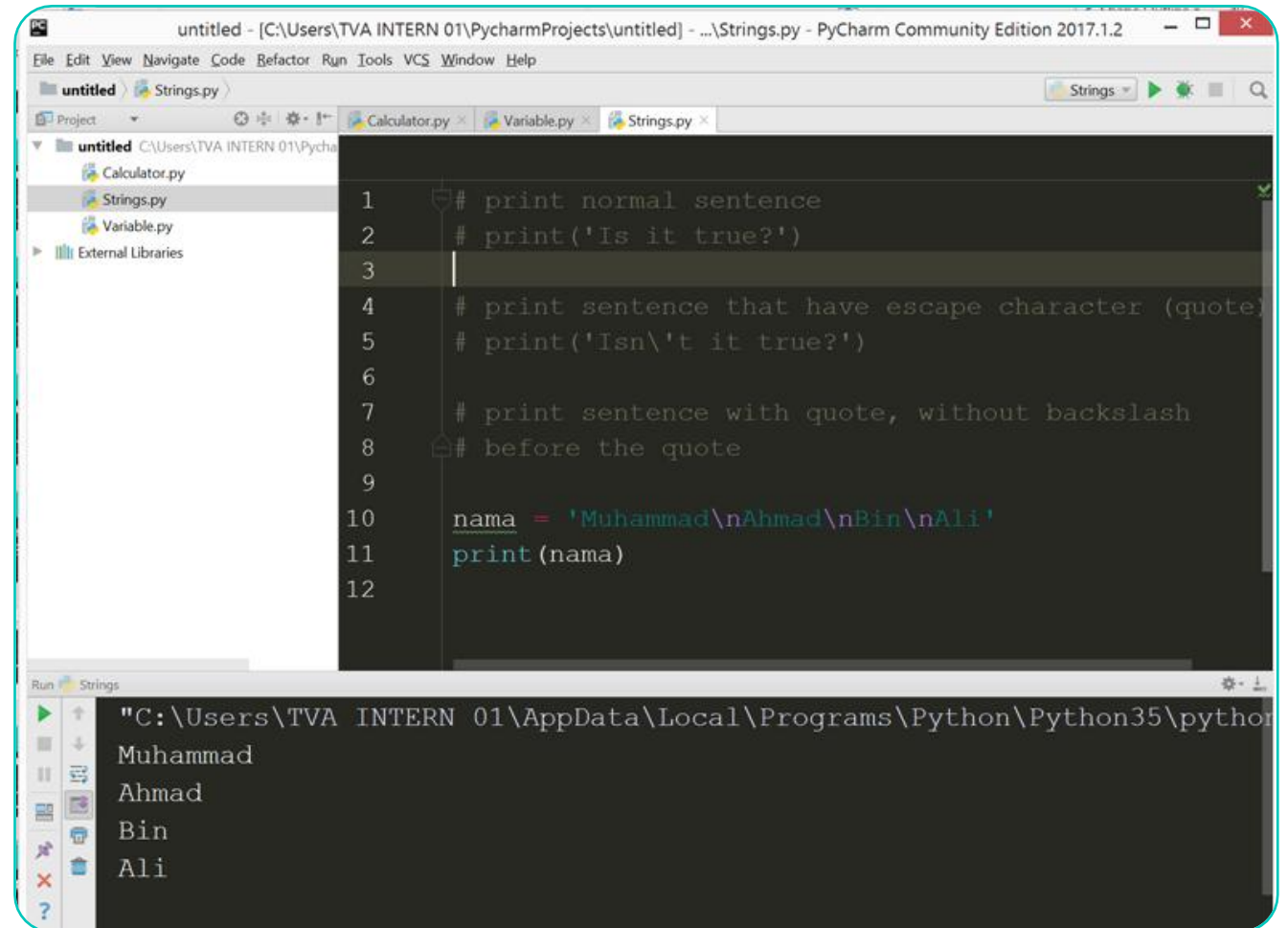
```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\python
Is it true?
Isn't it true?
Process finished with exit code 0
```

The status bar at the bottom indicates "PEP 8: blank line at end of file" and "9:1 n/a UTF-8".

Strings in Python (Cont.)

- Previous example just show how to print the output in one line. How to about to print an output that has multiple lines?
- Using '\n' to print a new line
- For example:

```
nama =  
'Muhammad\nAhmad\nBin\nAli'  
  
print(nama)
```
- Above example shows how to print multiple line using '\n' character.



The screenshot shows the PyCharm Community Edition 2017.1.2 interface. The main editor window displays a Python script named `Strings.py` with the following code:

```
1 # print normal sentence  
2 # print('Is it true?')  
3  
4 # print sentence that have escape character (quote)  
5 # print('Isn\'t it true?')  
6  
7 # print sentence with quote, without backslash  
8 # before the quote  
9  
10 nama = 'Muhammad\nAhmad\nBin\nAli'  
11 print(nama)  
12
```

The Run window at the bottom shows the output of the script:

```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\python  
Muhammad  
Ahmad  
Bin  
Ali
```

Strings in Python (Cont.)

- Strings also can do concatenate by using plus symbol

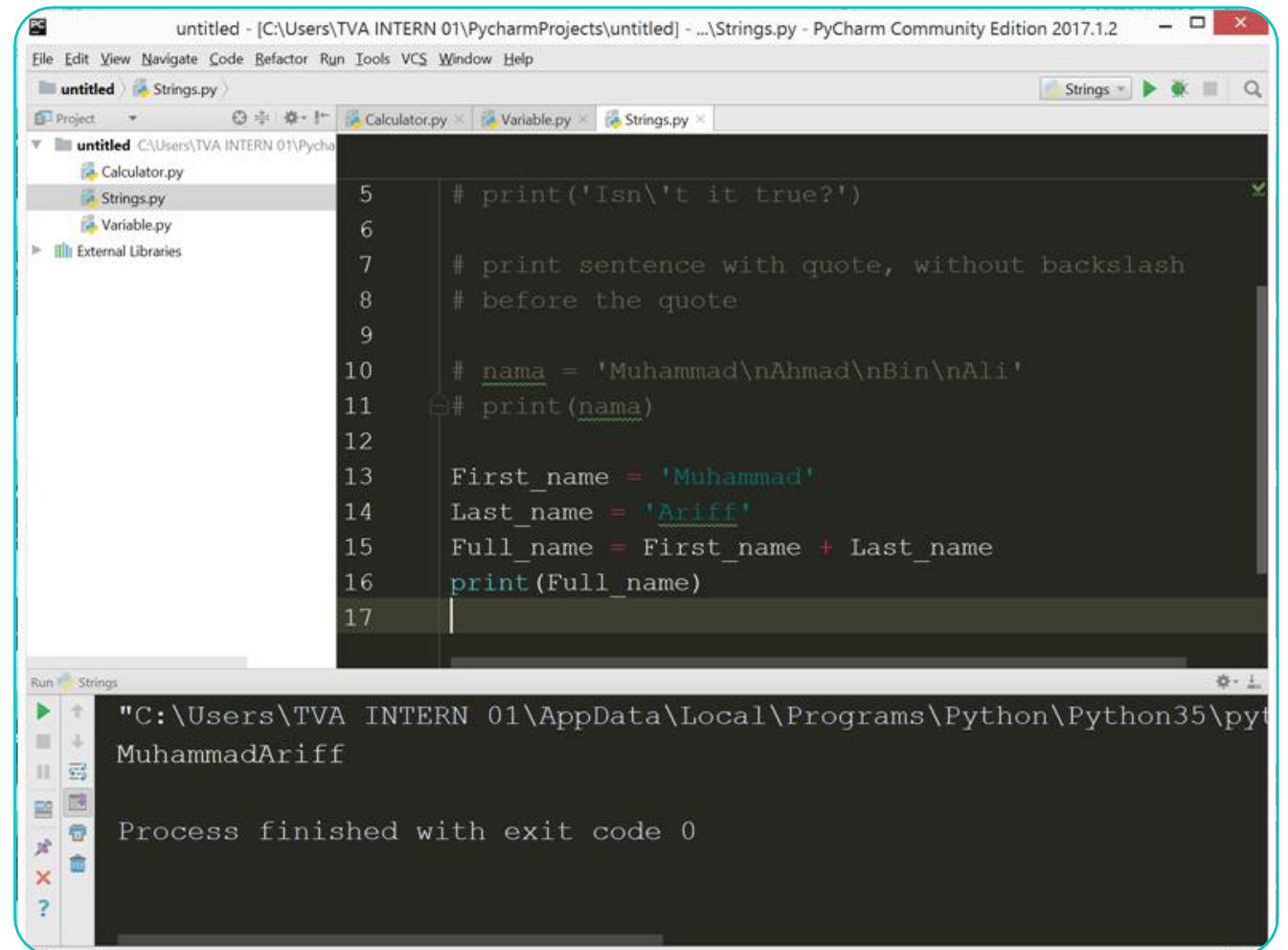
- For example:

First_name = 'Muhammad'

Last_name = 'Ariff'

Full_name = First_name +
Last_name

print(Full_name)



The screenshot shows the PyCharm IDE interface. The top window displays a Python script named 'Strings.py' with the following code:

```
5 # print('Isn\'t it true?')
6
7 # print sentence with quote, without backslash
8 # before the quote
9
10 # nama = 'Muhammad\nAhmad\nBin\nAli'
11 # print(nama)
12
13 First_name = 'Muhammad'
14 Last_name = 'Ariff'
15 Full_name = First_name + Last_name
16 print(Full_name)
17
```

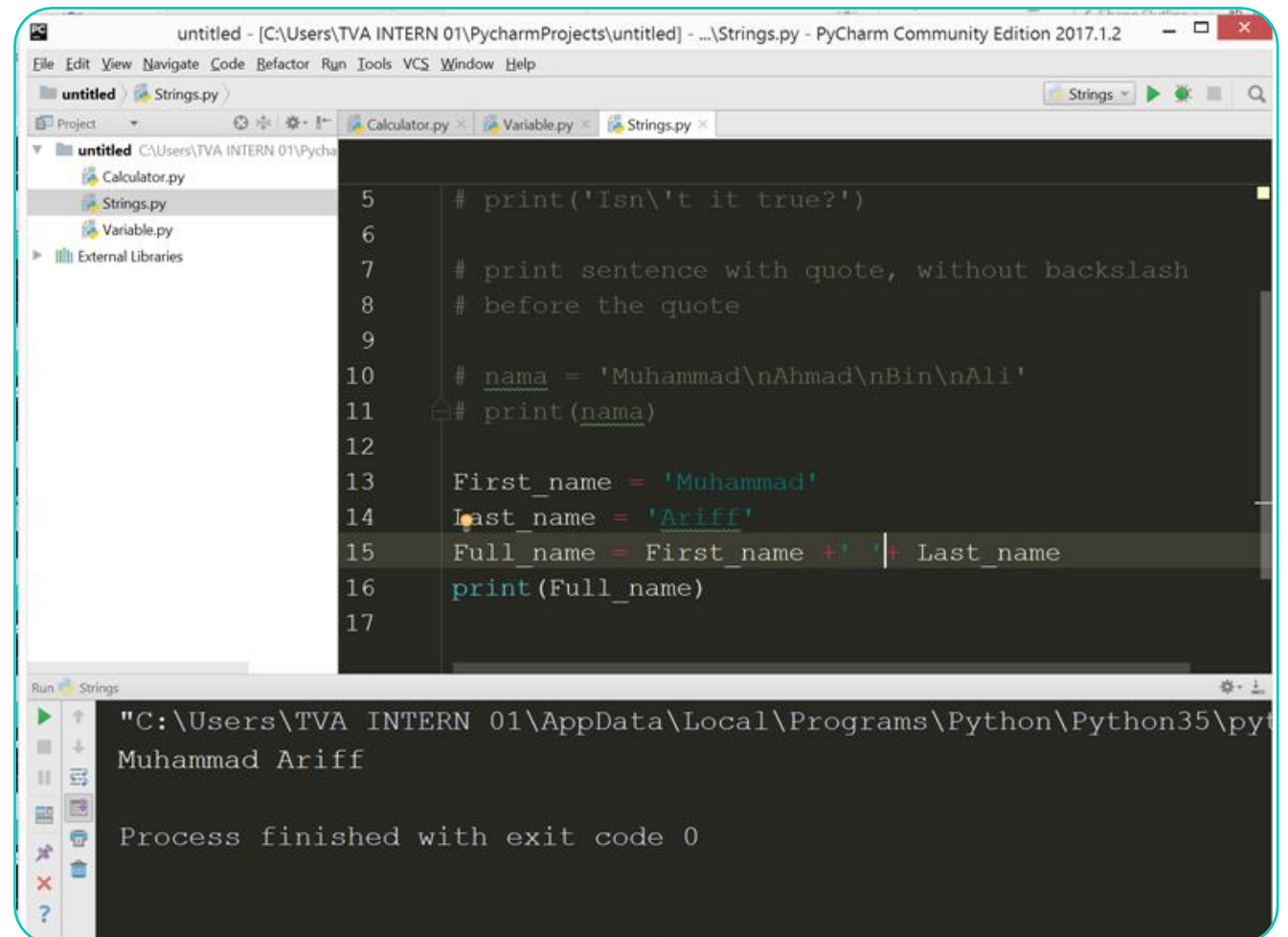
The bottom window shows the output of the script execution:

```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\python.exe" C:\Users\TVA INTERN 01\PycharmProjects\untitled\Strings.py
MuhammadAriff

Process finished with exit code 0
```

Strings in Python (Cont.)

- Previous example shows that we can combine two variables of strings. But the result does not have space between them.
- To include it, just add space between them.
- Example:
`Full_name = First_name + ' ' + last_name`



The screenshot shows the PyCharm IDE interface. The top toolbar includes menus like File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. The left sidebar shows a project tree with files: untitled, Calculator.py, Strings.py, Variable.py, and External Libraries. The main editor window displays the following Python code:

```
5 # print('Isn\'t it true?')
6
7 # print sentence with quote, without backslash
8 # before the quote
9
10 # nama = 'Muhammad\nAhmad\nBin\nAli'
11 # print(nama)
12
13 First_name = 'Muhammad'
14 last_name = 'Ariff'
15 Full_name = First_name + ' ' + Last_name
16 print(Full_name)
17
```

Below the editor, the Run console shows the output of the script:

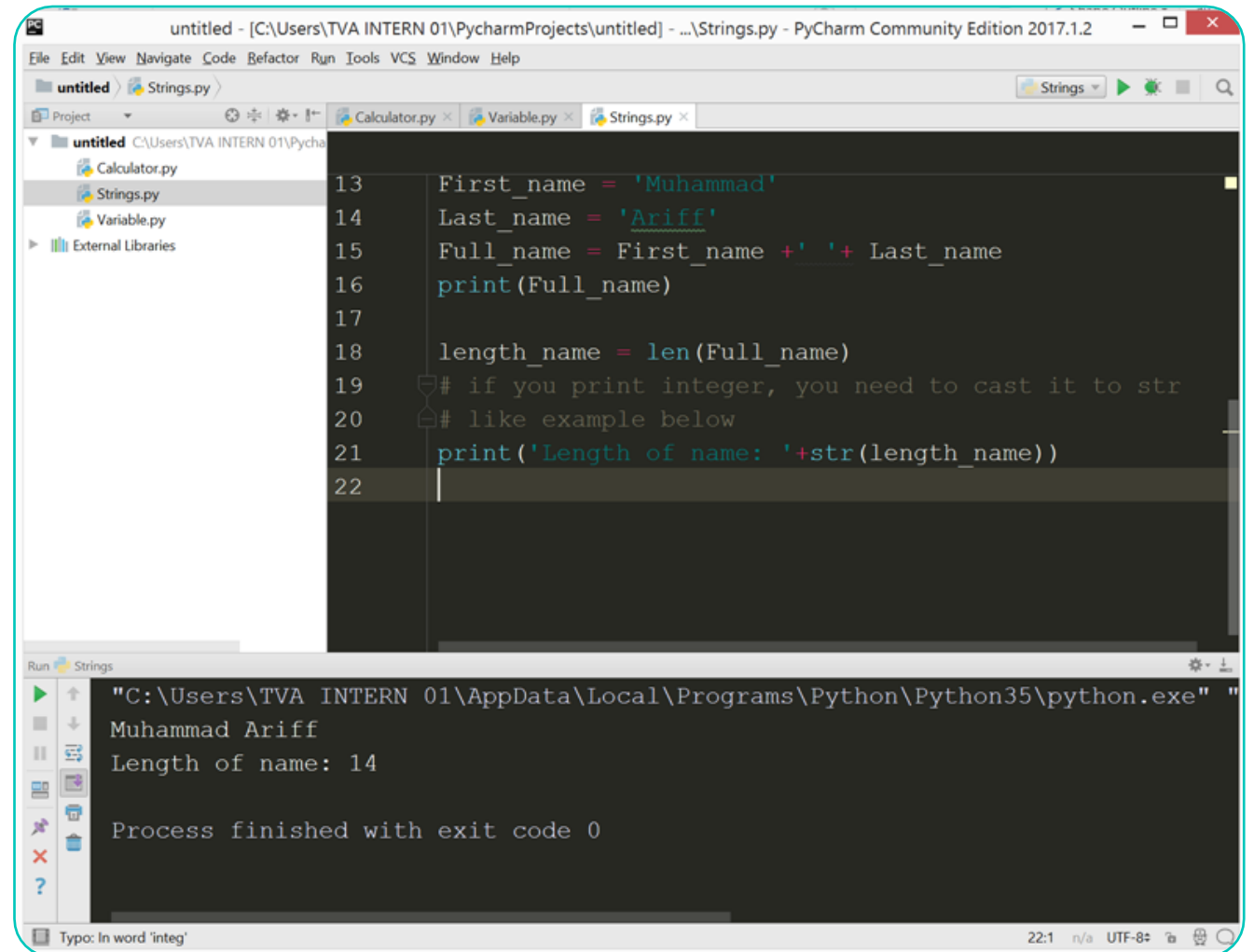
```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\pyt
Muhammad Ariff

Process finished with exit code 0
```


Strings in Python (Cont.)

- So you want to calculate how many characters in your name?
- Using function named `len`
- Example:

```
Length_name =  
len(full_name)
```
- The value length is including space



The screenshot shows the PyCharm IDE interface. The main editor window displays a Python script named `Strings.py` with the following code:

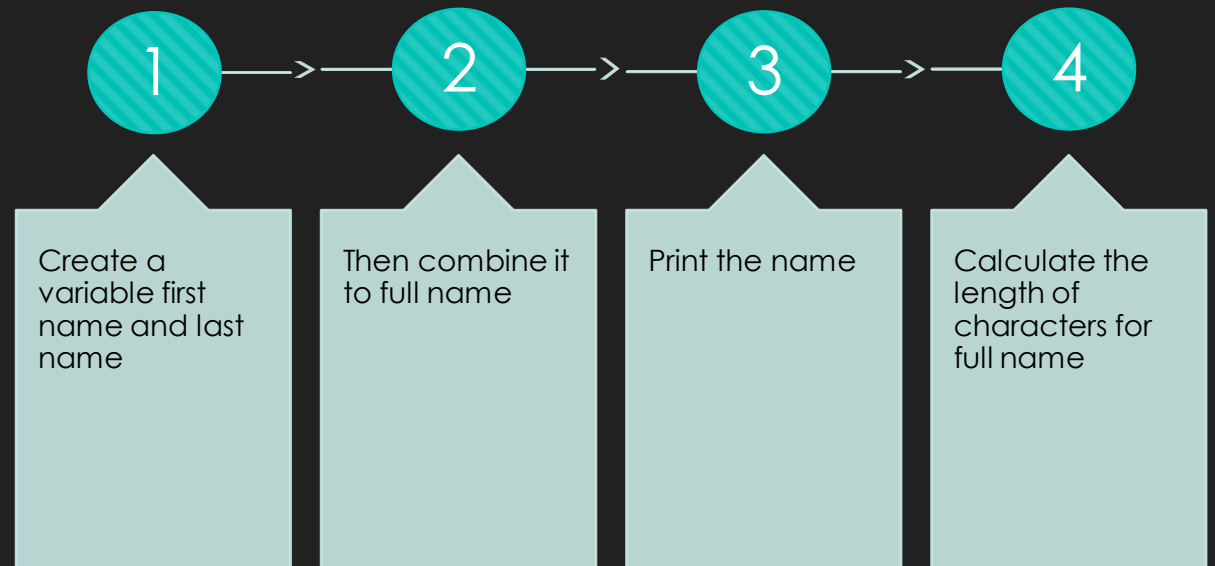
```
13 First_name = 'Muhammad'  
14 Last_name = 'Ariff'  
15 Full_name = First_name + ' ' + Last_name  
16 print(Full_name)  
17  
18 length_name = len(Full_name)  
19 # if you print integer, you need to cast it to str  
20 # like example below  
21 print('Length of name: ' + str(length_name))  
22
```

The bottom panel shows the output of the script execution:

```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\python.exe" "  
Muhammad Ariff  
Length of name: 14  
  
Process finished with exit code 0
```

The status bar at the bottom indicates a typo in the word 'integ' and the time is 22:1.

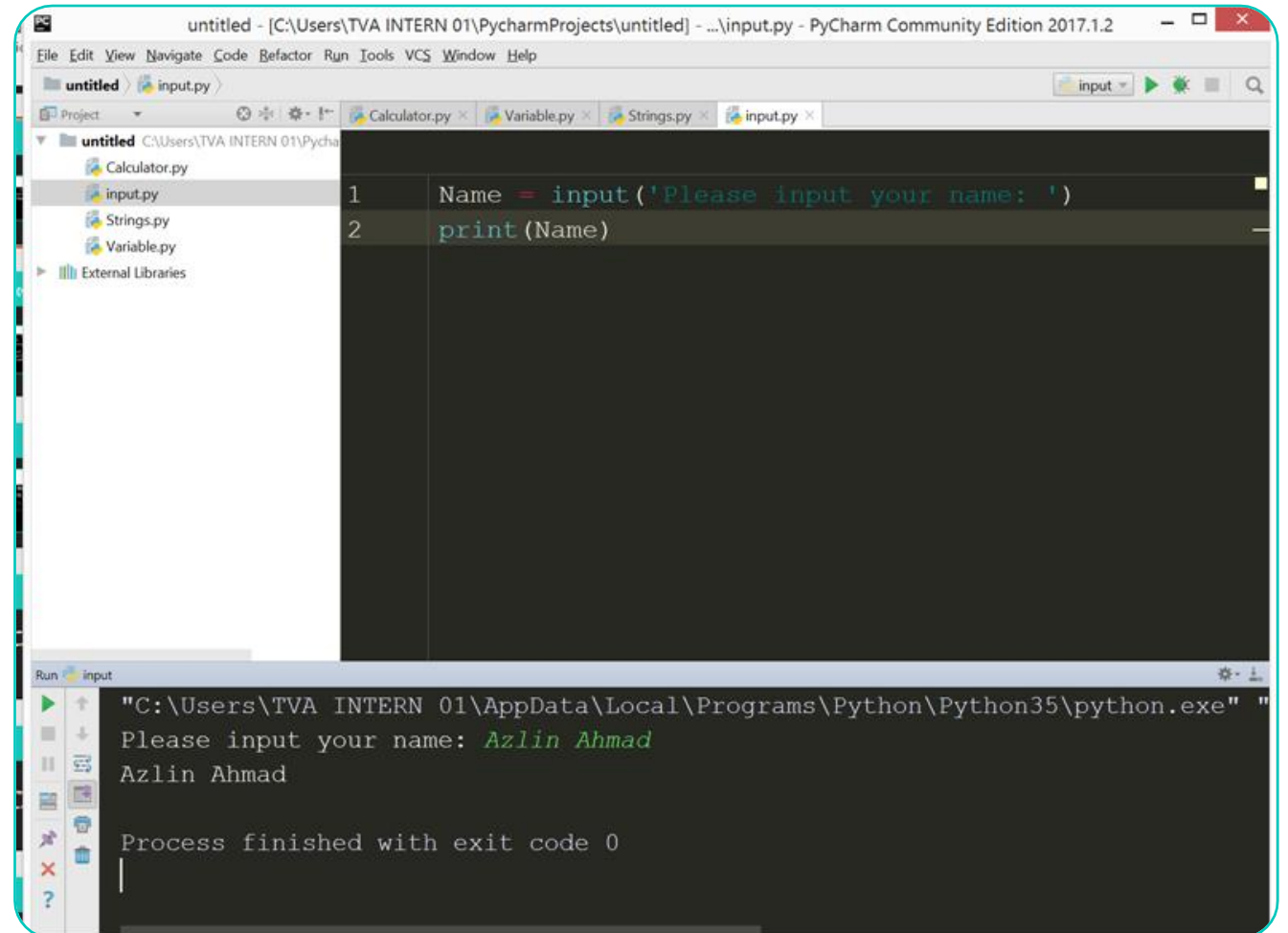
Exercise for Strings



Taking input from user

- Sometimes you need to make a program that need to take an input from the user.
- For that purpose, you cannot simply assign the value, since it depends on the user to key in.
- Example of taking an input using python:

```
Name = input('Please input your name: ')
```
- We use *input* function to take the input. You can write any strings between the bracket and quotes.



The screenshot shows the PyCharm Community Edition 2017.1.2 interface. The main editor window displays a Python file named `input.py` with the following code:

```
1 Name = input('Please input your name: ')  
2 print(Name)
```

The left sidebar shows the project structure with files `Calculator.py`, `input.py`, `Strings.py`, and `Variable.py`. The bottom panel shows the Run output for the `input.py` script:

```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\python.exe" "  
Please input your name: Azlin Ahmad  
Azlin Ahmad  
  
Process finished with exit code 0
```

Taking input from user (Cont.)

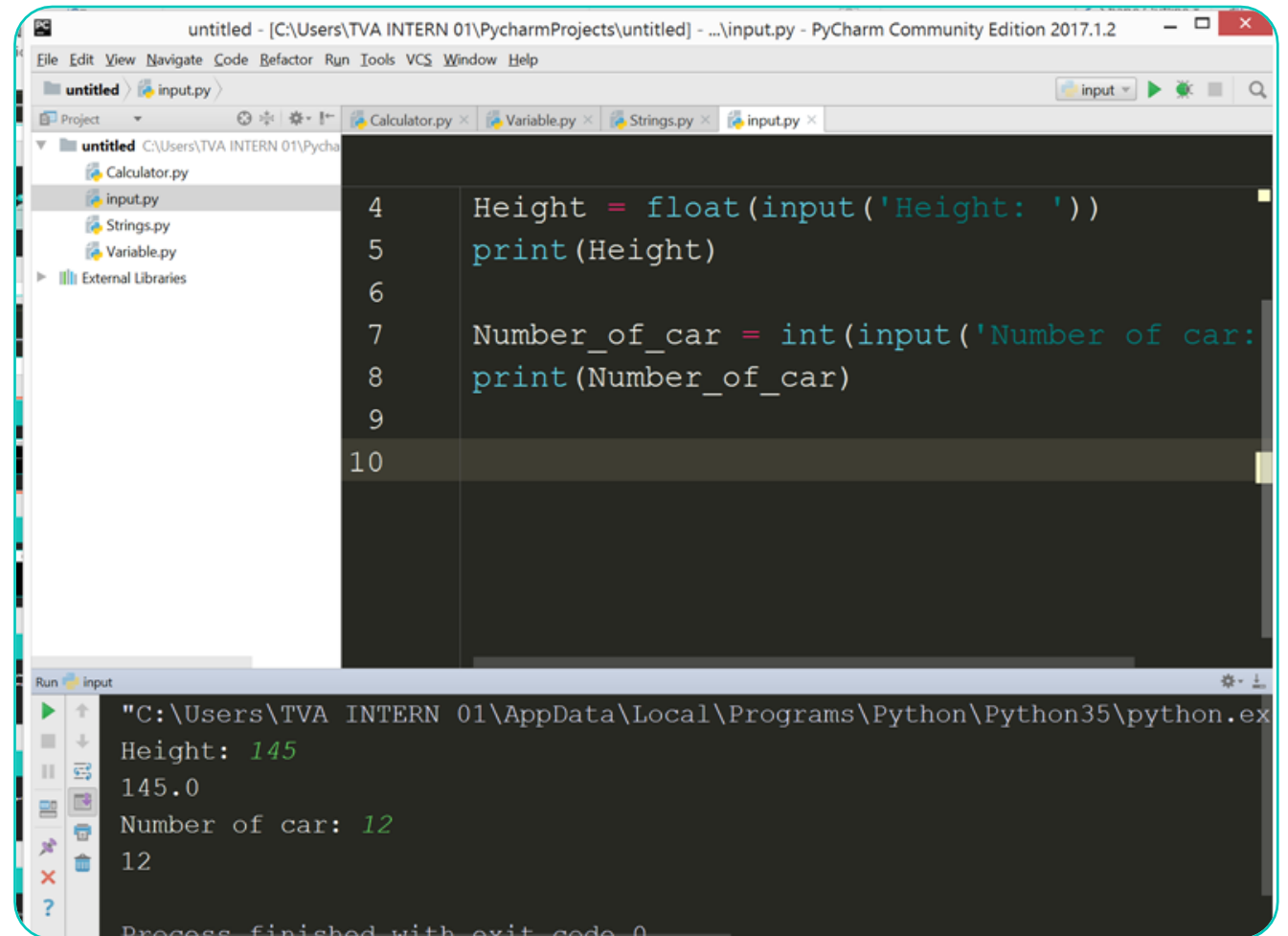
- But if you want to take an input *integer* or *float*, then we need to casting (transform) it into *integer* or *float*.

- By default, input function will take the input as a *String*.

- Example:

```
Height = float(input('Height: '))
```

```
Number_of_car =  
int(input('Number of car: '))
```



The screenshot shows the PyCharm IDE interface. The top pane displays a Python script named `input.py` with the following code:

```
4 Height = float(input('Height: '))  
5 print(Height)  
6  
7 Number_of_car = int(input('Number of car: '))  
8 print(Number_of_car)  
9  
10
```

The bottom pane shows the output of the program execution. It displays the prompts and user inputs:

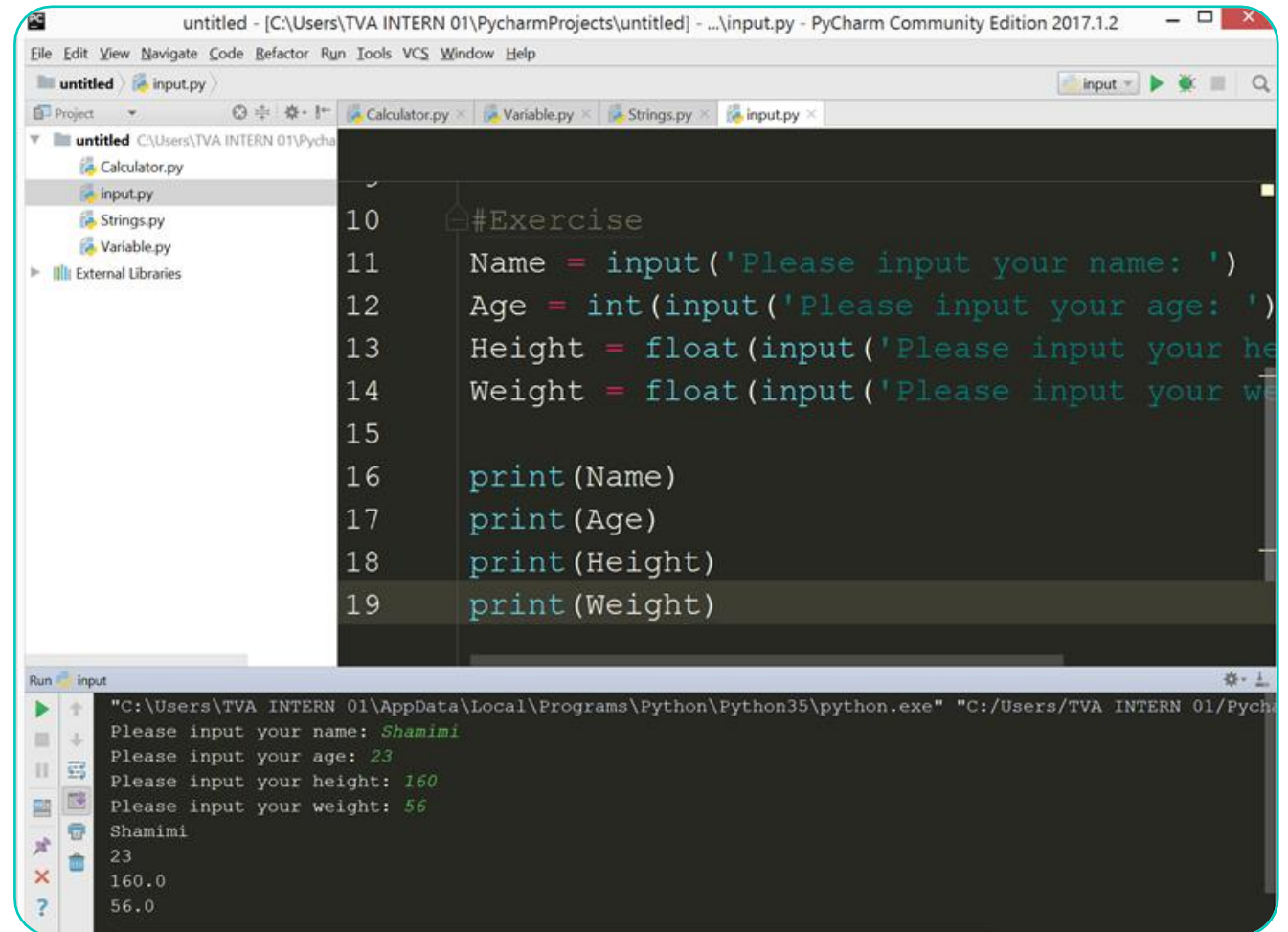
```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\python.exe"  
Height: 145  
145.0  
Number of car: 12  
12  
Process finished with exit code 0
```

Exercise for Input

- Make a program that taking input for:
 - Name
 - Age
 - Height
 - Weight
- Display all the information like the output in the picture

```
"C:\Users\TVA INTERN 01\AppData\Local  
Please input your name: Shamimi  
Please input your age: 23  
Please input your height: 160  
Please input your weight: 56  
Shamimi  
23  
160.0  
56.0
```

Answer for Input Exercise



The image shows a PyCharm IDE window titled "untitled - [C:\Users\TVA INTERN 01\PycharmProjects\untitled] - ...input.py - PyCharm Community Edition 2017.1.2". The main editor displays a Python script for an input exercise. The script prompts the user for their name, age, height, and weight, and then prints the entered values. The Run console at the bottom shows the output of the script, indicating that the program executed successfully with the provided input values.

```
10 #Exercise
11 Name = input('Please input your name: ')
12 Age = int(input('Please input your age: '))
13 Height = float(input('Please input your height: '))
14 Weight = float(input('Please input your weight: '))
15
16 print(Name)
17 print(Age)
18 print(Height)
19 print(Weight)
```

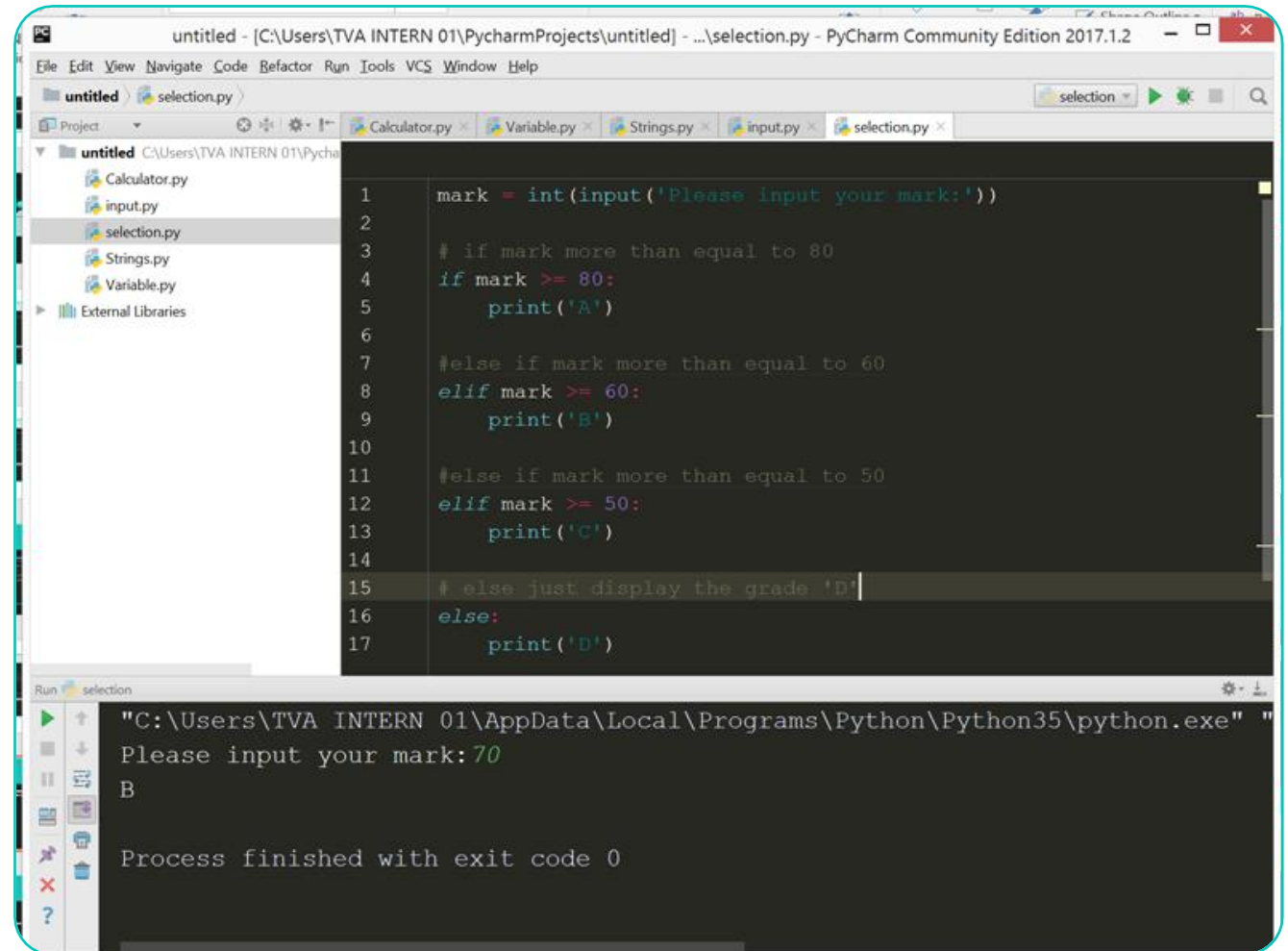
Run input

```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\python.exe" "C:/Users/TVA INTERN 01/PycharmProjects/untitled/input.py"
Please input your name: Shamimi
Please input your age: 23
Please input your height: 160
Please input your weight: 56
Shamimi
23
160.0
56.0
```

Selection in Python

- Selection is a situation where we need to choose or select something based on circumstances.
- For example, **grade A** will be given to the student **IF** the student's mark get **more than 80%**
- Example in Python:

```
if mark >= 80:  
    grade = 'A'  
elif mark >= 60:  
    grade = 'B'  
elif mark >= 50:  
    grade = 'C'  
else:  
    grade = 'D'
```



The screenshot displays the PyCharm Community Edition 2017.1.2 interface. The main editor window shows a Python file named 'selection.py' with the following code:

```
1 mark = int(input('Please input your mark:'))  
2  
3 # if mark more than equal to 80  
4 if mark >= 80:  
5     print('A')  
6  
7 #else if mark more than equal to 60  
8 elif mark >= 60:  
9     print('B')  
10  
11 #else if mark more than equal to 50  
12 elif mark >= 50:  
13     print('C')  
14  
15 # else just display the grade 'D'  
16 else:  
17     print('D')
```

The left sidebar shows the project structure with files: Calculator.py, input.py, selection.py, Strings.py, and Variable.py. The bottom console window shows the execution output:

```
"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\python.exe" "  
Please input your mark:70  
B  
Process finished with exit code 0
```

Looping in Python

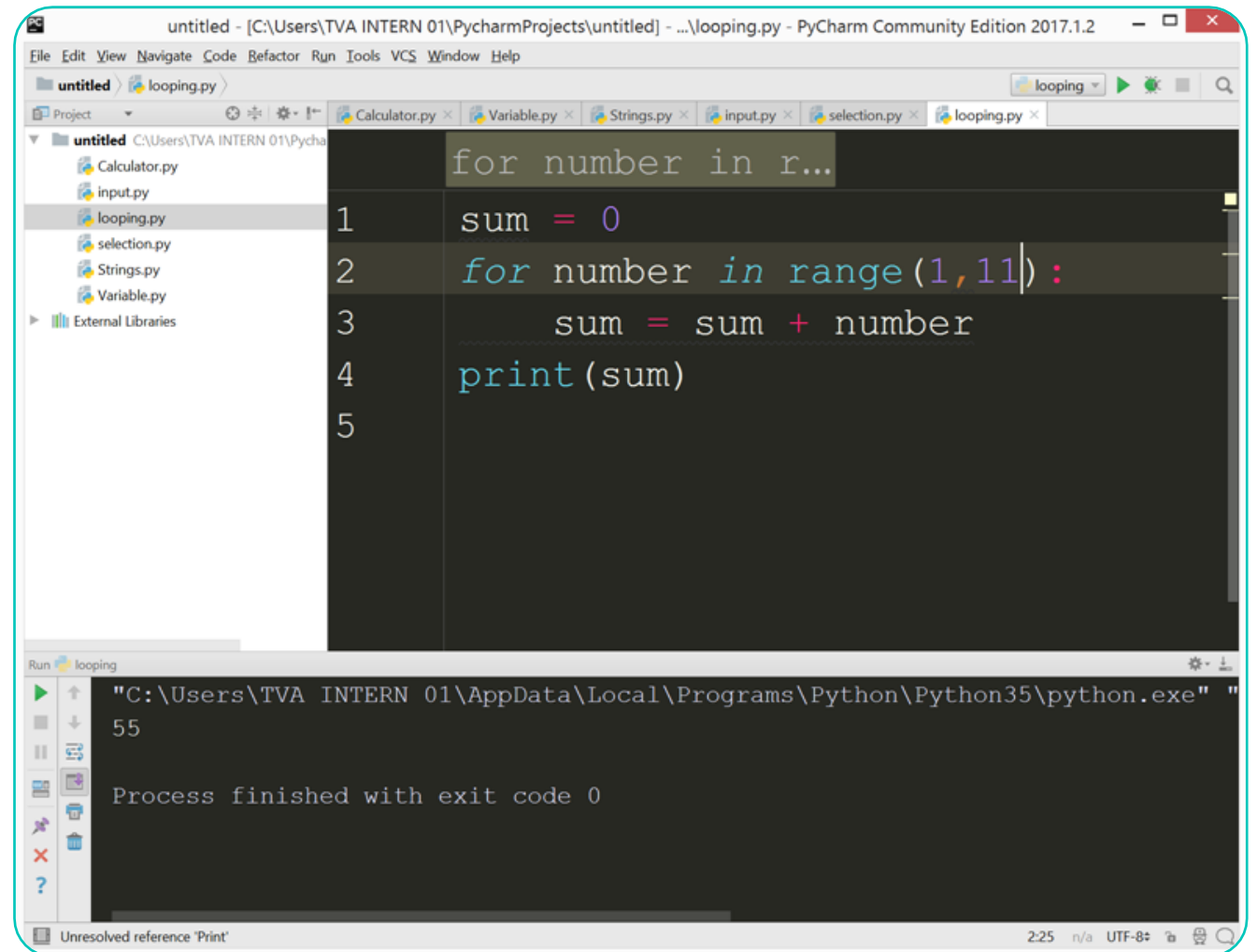
- Selection is a situation where we need to repeat the same task for specific times or infinity
- For example, you want to add numbers from 1 to 10
- Example in Python:

```
sum = 0
```

```
for number in range(1,11):
```

```
    sum = sum + number
```

```
print(sum)
```



The screenshot shows the PyCharm IDE interface. The top toolbar includes buttons for running (a green play icon) and debugging (a green bug icon). The main editor window displays a Python script in a dark theme. The script is as follows:

```
1 sum = 0
2 for number in range(1,11):
3     sum = sum + number
4 print(sum)
5
```

The left sidebar shows a project view with files: Calculator.py, input.py, looping.py (selected), selection.py, Strings.py, and Variable.py. Below the editor is a 'Run' console window. It shows the command executed: `"C:\Users\TVA INTERN 01\AppData\Local\Programs\Python\Python35\python.exe"`, the output `55`, and the message `Process finished with exit code 0`. The status bar at the bottom indicates an 'Unresolved reference 'Print'' and the time is 2:25.

Conclusion

- Python is easier in doing calculation, display output and taking an input
- Use *print()* to display any output
- Use variable to store any value, it is easier to use that variable again

Resources

- <http://interactivepython.org/courselib/static/pythonds/Introduction/WhatIsProgramming.html>
- <https://docs.python.org/3/tutorial/index.html>