Learn Python in 1 Day

By Krishna Rungta

Copyright 2018 - All Rights Reserved – Krishna Rungta

ALL RIGHTS RESERVED. No part of this publication may be reproduced or transmitted in any form whatsoever, electronic, or mechanical, including photocopying, recording, or by any informational storage or retrieval system without express written, dated and signed permission from the author.

Table Of Content

Chapter 1: Python Tutorial for Beginners: Learn Programming in 7 Days

1. Characteristics of Python
2. Python used for

Chapter 2: How to Install Python on Windows [Pycharm IDE]

1. Installing Python
2. Installing Pycharm

Chapter 3: Hello World: Create your First Python Program

Chapter 4: Pyth on Main Function with Examples: Understand __main__

Chapter 5: Python Variables: Declare, Concatenate, Global & Local

1. What is a Variable in Python?
2. How to Declare and use a Variable
3. Re-declare a Variable
4. Concatenate Variables
5. Local & Global Variables
6. Delete a variable

Chapter 6: Python Strings: Replace, Join, Split, Reverse, Uppercase & Lowercase

1. Accessing Values in Strings
2. Various String Operators
3. Some more examples
4. Python String replace() Method
5. Changing upper and lower case strings
6. Using ""join"" function for the string
7. Reversing String
8. Split Strings

Chapter 7: Python TUPLE - Pack, Unpack, Compare, Slicing, Delete, Key

1. Packing and Unpacking
2. Comparing tuples
3. Using tuples as keys in dictionaries
4. Deleting Tuples
5. Slicing of Tuple
6. Built-in functions with Tuple
7. Advantages of tuple over list

Chapter 8: Python Dictionary(Dict): Update, Cmp, Len, Sort, Copy, Items, str Example

1. Python Dictionary Methods
2. Python Dictionary in-built Functions

Chapter 9: Python Operators: Arithmetic, Logical, Comparison, Assignment, Bitwise &

1. Arithmetic Operators
2. Comparison Operators
3. Python Assignment Operators
4. Logical Operators
5. Membership Operators
6. Identity Operators
7. Operator precedence

Chapter 10: Python Functions Examples: Call, Indentation, Arguments & Return Values

1. How to define and call a function in Python
2. Significance of Indentation (Space) in Python
3. How Function Return Value?
4. Arguments in Functions

Chapter 11: Python IF, ELSE, ELIF, Nested IF & Switch Case Statement

1. What is If Statement? How to Use it?

2. What happen when ""if condition"" does not meet
3. How to use ""else condition""
4. When ""else condition"" does not work
5. How to use ""elif"" condition
6. How to execute conditional statement with minimal code
7. Nested IF Statement
8. Switch Statement

Chapter 12: Python For & While Loops: Enumerate, Break, Continue Statement

1. How to use ""While Loop""
2. How to use ""For Loop""
3. How to use For Loop for String
4. How to use break statements in For Loop
5. How to use ""continue statement"" in For Loop
6. How to use ""enumerate"" function for ""For Loop""
7. Pratical Example
8. How to use for loop to repeat the same statement over and again

Chapter 13: Python OOPs: Class, Object, Inheritance and Constructor with Example

1. How to define Python classes

2. How Inheritance works
3. Python Constructors

Chapter 14: Python Regex: re.match(), re.search(), re.findall() with Example

1. Regular Expression Syntax
2. Example of w+ and ^ Expression
3. Example of \s expression in re.split function
4. Using regular expression methods
5. Using re.match()
6. Finding Pattern in Text (re.search())
7. Using re.findall for text
8. Python Flags
9. Example of re.M or Multiline Flags

Chapter 15: Python DateTime, TimeDelta, Strftime(Format) with Examples

1. How to Use Date & DateTime Class
2. Print Date using date.today()
3. Python Current Date and Time: now() today()
4. How to Format Date and Time Output with Strftime()

5. How to use Timedelta Objects "
6. Python 2 Example "

Chapter 16: Python CALENDAR Tutorial with Example

Chapter 17: Python File Handling: Create, Open, Append, Read, Write

1. How to Create a Text File
2. How to Append Data to a File
3. How to Read a File
4. How to Read a File line by line
5. File Modes in Python

Chapter 18: Python Check If File or Directory Exists

1. os.path.exists()
2. os.path.isfile()
3. os.path.isdir()
4. pathlibPath.exists() For Python 3.4

Chapter 19: Python COPY File using shutil.copy(), shutil.copystat()

Chapter 20: Python Rename File and Directory using os.rename()

Chapter 21: Python ZIP file with Example

Chapter 22: Python Internet Access using Urllib.Request and urlopen()

1. How to Open URL using Urllib
2. How to get HTML file form URL in Python

Chapter 23: Python XML Parser Tutorial: Read xml file example(Minidom, ElementTree)

1. How to Parse XML using minidom
2. How to Create XML Node
3. How to Parse XML using ElementTree

Chapter 24: Python vs JAVA vs PHP vs PERL vs Ruby vs JavaScript vs C++ vs TCL

Chapter 1: Python Tutorial for Beginners: Learn Programming in 7 Days

Python is an object-oriented programming language created by Guido Rossum in 1989. It is ideally designed for rapid prototyping of complex applications. It has interfaces to many OS system calls and libraries and is extensible to C or C++. Many large companies use the Python programming language include NASA, Google, YouTube, BitTorrent, etc.

Python is widely used in Artificial Intelligence, Natural Language

Generation, Neural Networks and other advanced fields of Computer Science. Python had deep focus on code readability & this class will teach you python from basics.

Characteristics of Python

- It provides rich data types and easier to read syntax than any other programming languages
- It is a platform independent scripted language with full access to operating system API's
- Compared to other programming languages, it allows more run-time flexibility
- It includes the basic text manipulation facilities of Perl and Awk
- A module in Python may have one or more classes and free functions
- Libraries in Pythons are cross-platform compatible with Linux, MacIntosh, and Windows
- For building large applications, Python can be compiled to byte-code
- Python supports functional and structured programming as well as OOP

- It supports interactive mode that allows interacting Testing and debugging of snippets of code
- In Python, since there is no compilation step, editing, debugging and testing is fast.

Python used for

- Programming video games
- Artificial Intelligence algorithm
- Programming various scientific programs such as statistical models

Chapter 2: How to Install Python on Windows [Pycharm IDE]

PyCharm is a cross-platform editor developed by JetBrains. Pycharm provides all the tools you need for productive Python development.

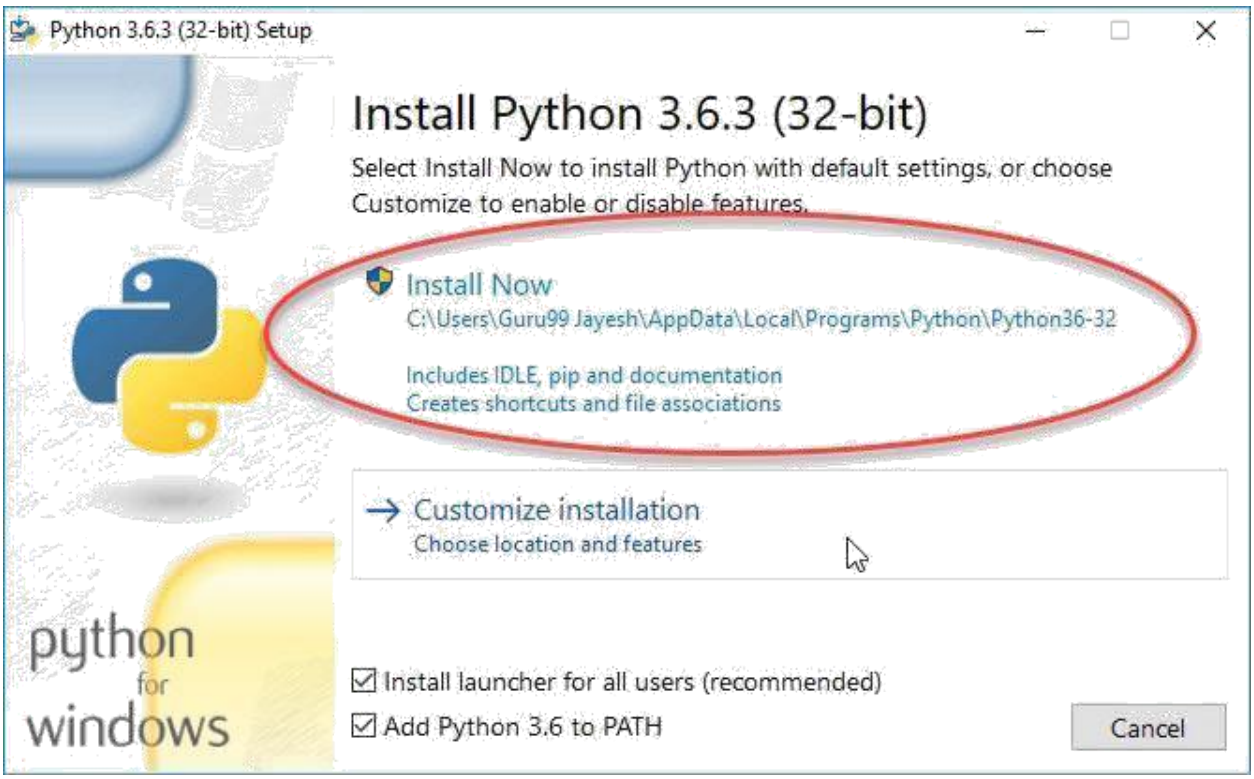
Below are the detailed steps for installing Python and PyCharm

Installing Python

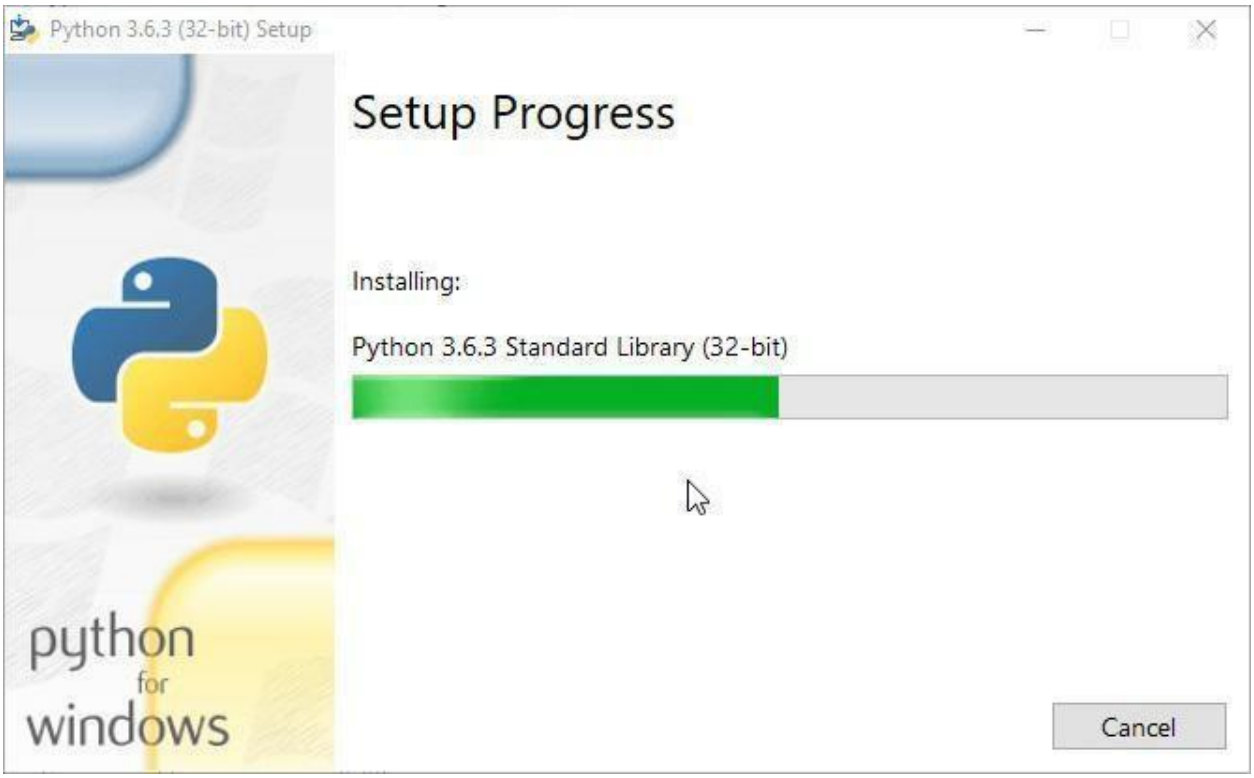
Step 1) To download and install Python visit the official website of Python http://www.python.org/downloads/ and choose your version. We have chosen Python version 3.6.3

Step 2) Once the download is complete, run the exe for install Python.

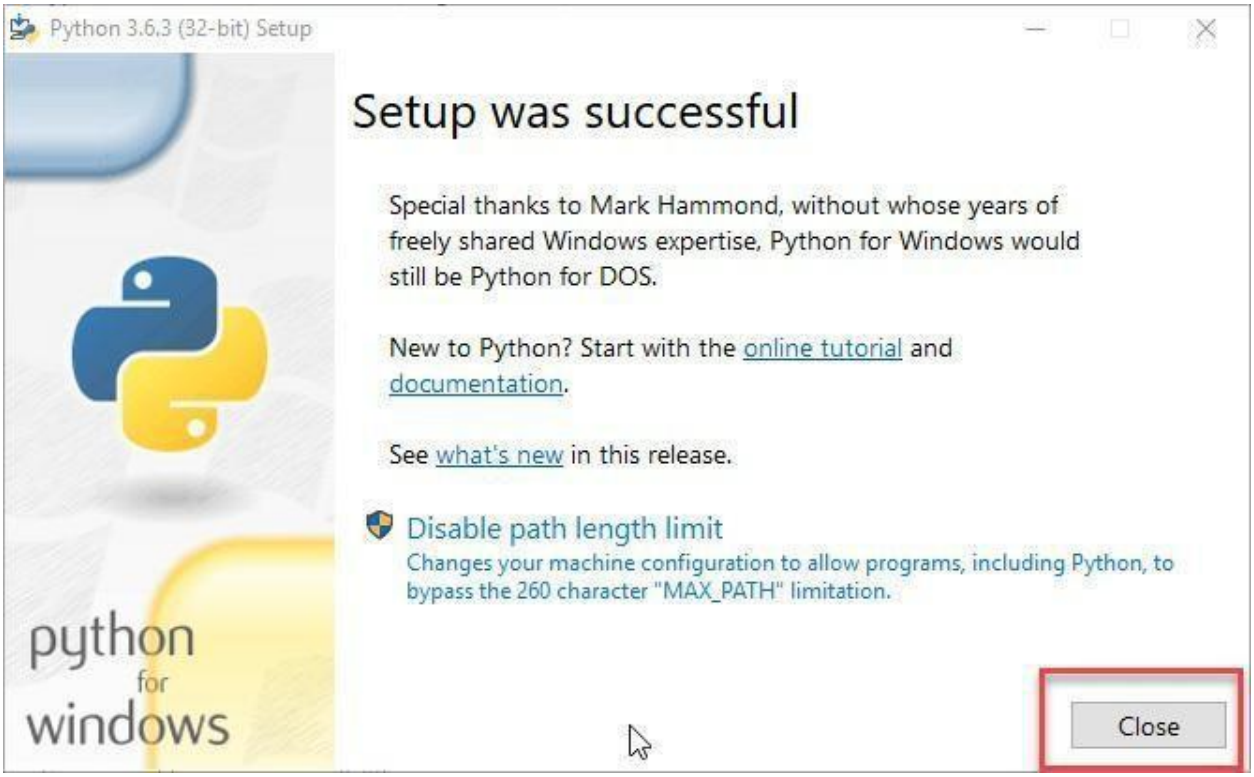
Now click on Install Now.



Step 3) You can see Python installing at this point.

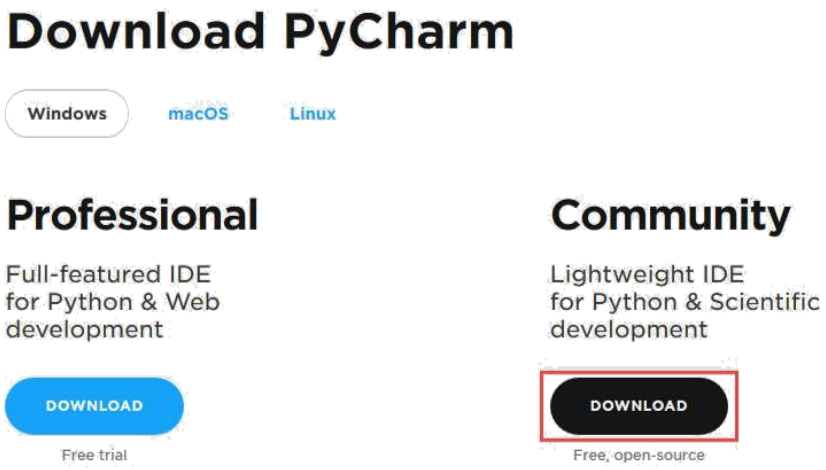


Step 4) When it finishes, you can see a screen that says the Setup was successful. Now click on "Close".

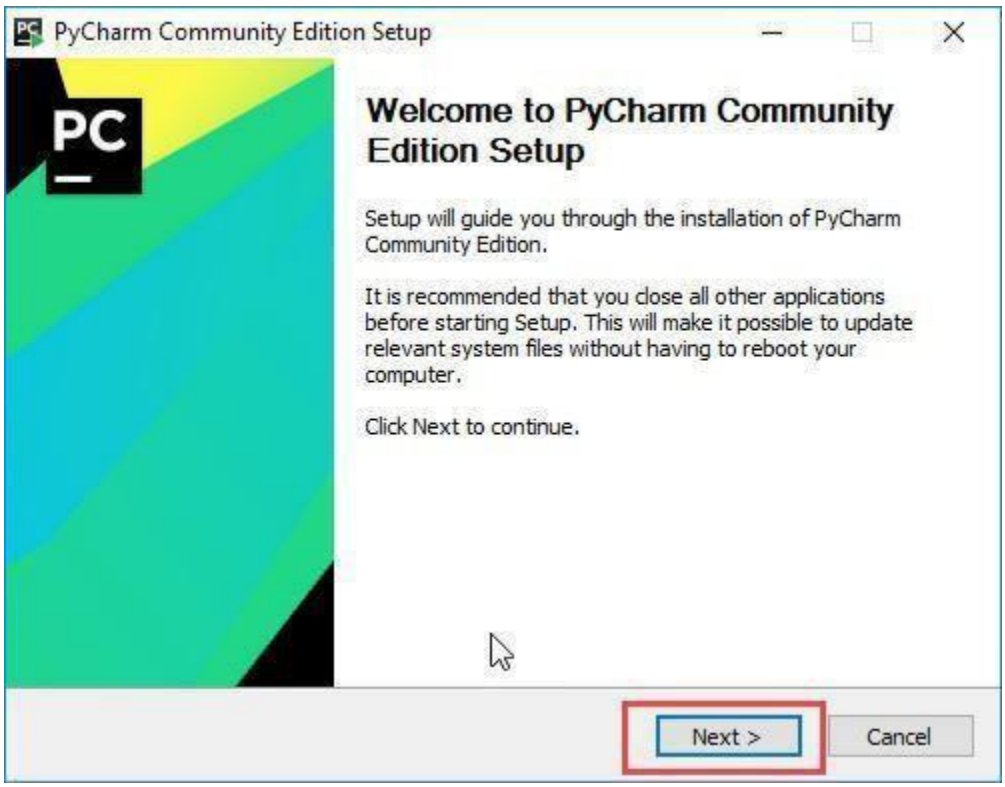


Installing Pycharm

Step 1) To download PyCharm visit the website https://www.jetbrains.com/pycharm/download/ and Click the "DOWNLOAD" link under the Community Section.

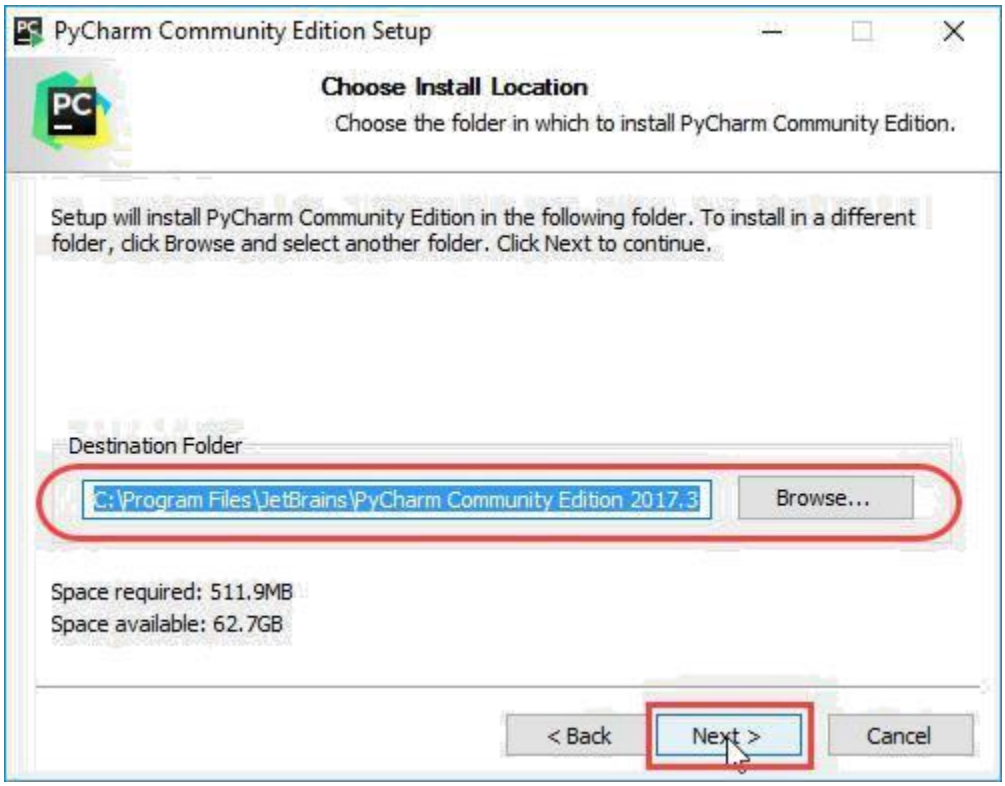


Step 2) Once the download is complete, run the exe for install PyCharm. The setup wizard should have started. Click "Next".

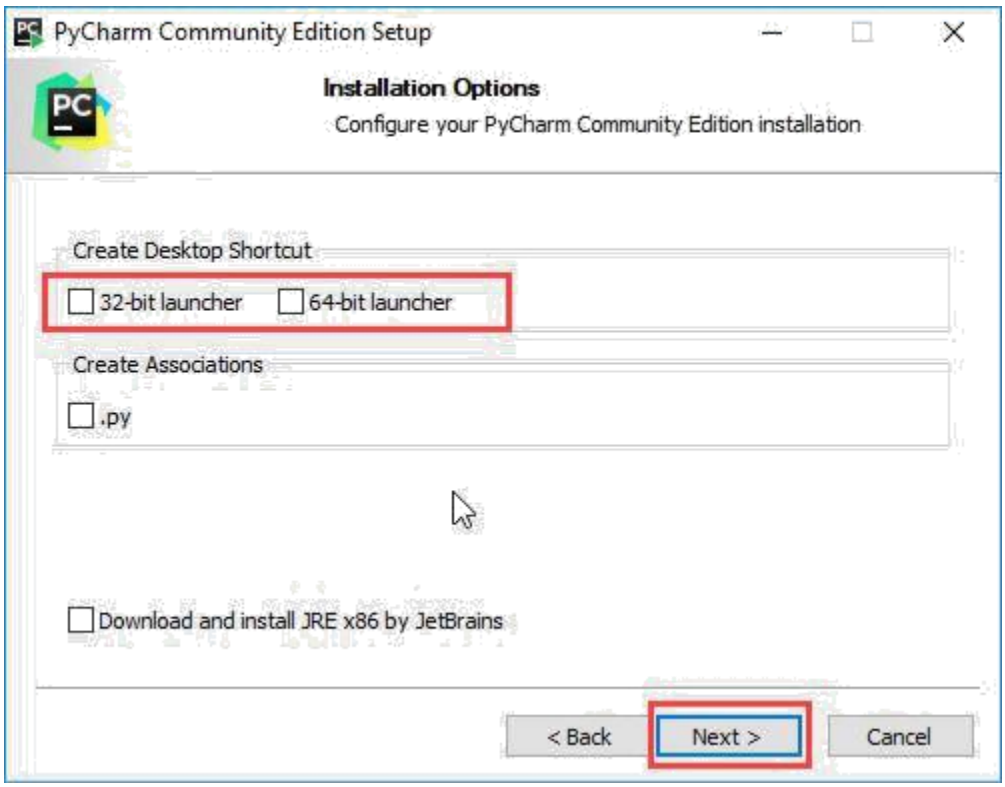


Step 3) On the next screen, Change the installation path if required.

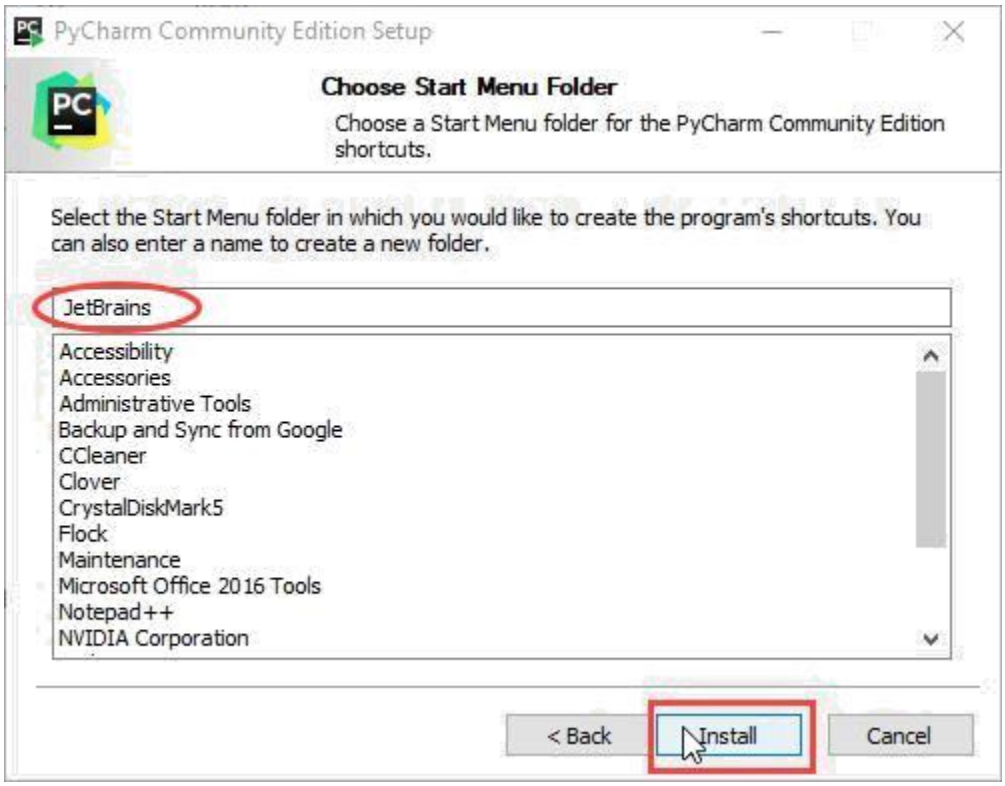
Click "Next".



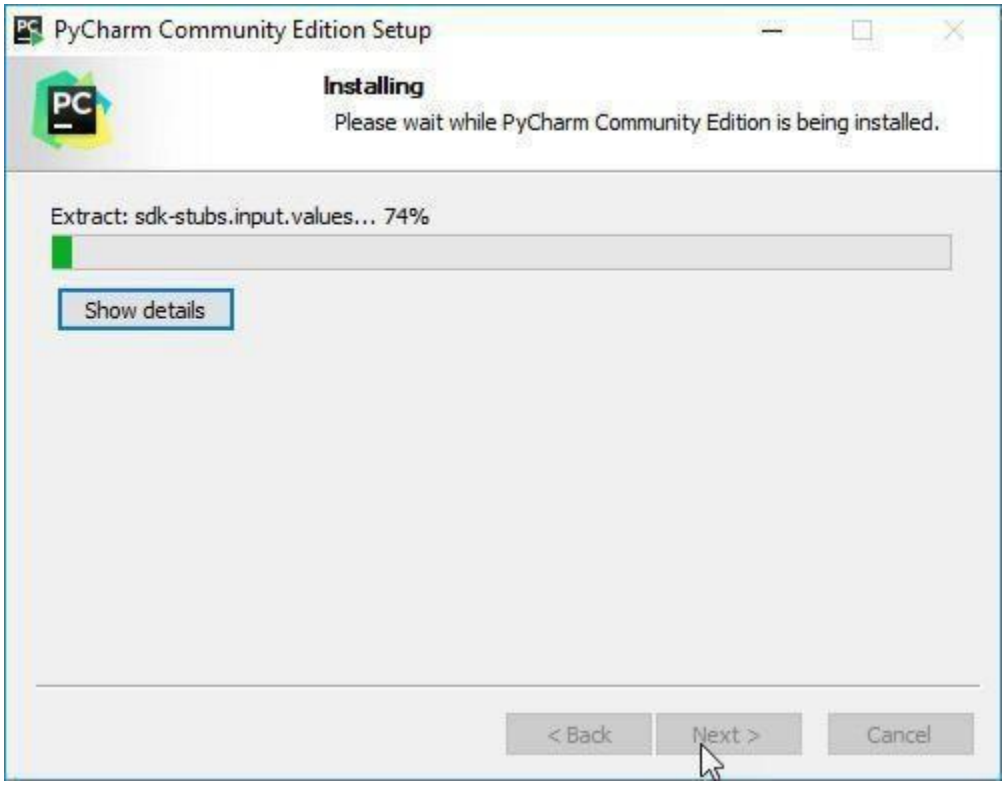
Step 4) On the next screen, you can create a desktop shortcut if you want and click on "Next".



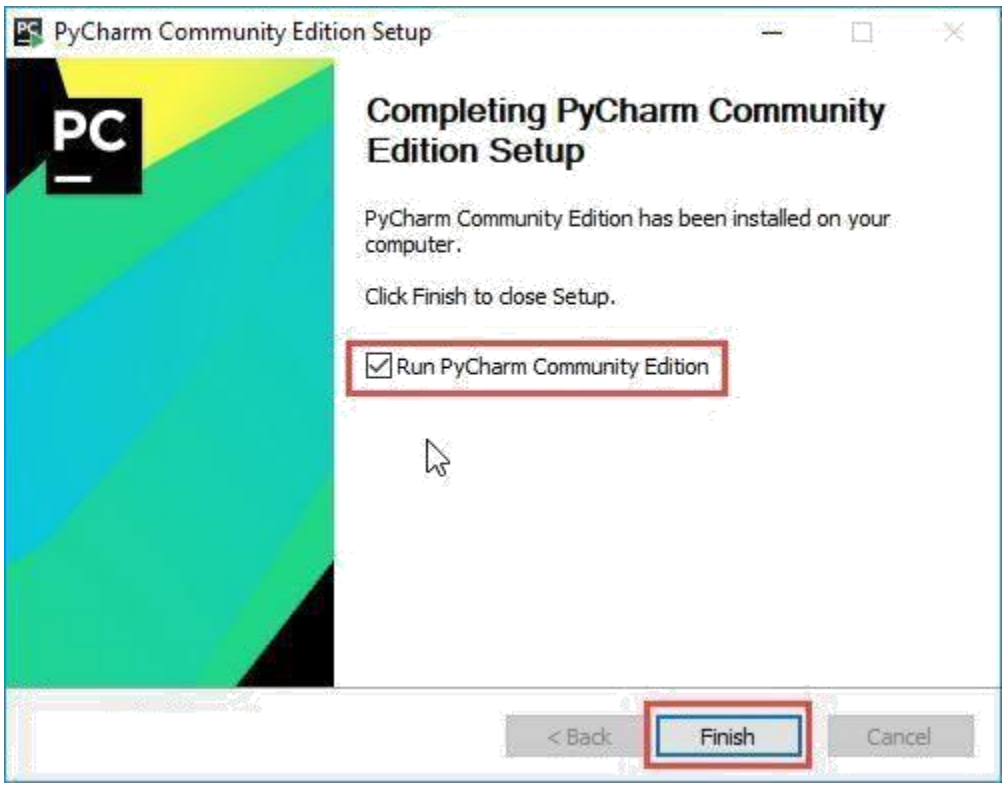
Step 5) Choose the start menu folder. Keep selected JetBrains and click on "Install".



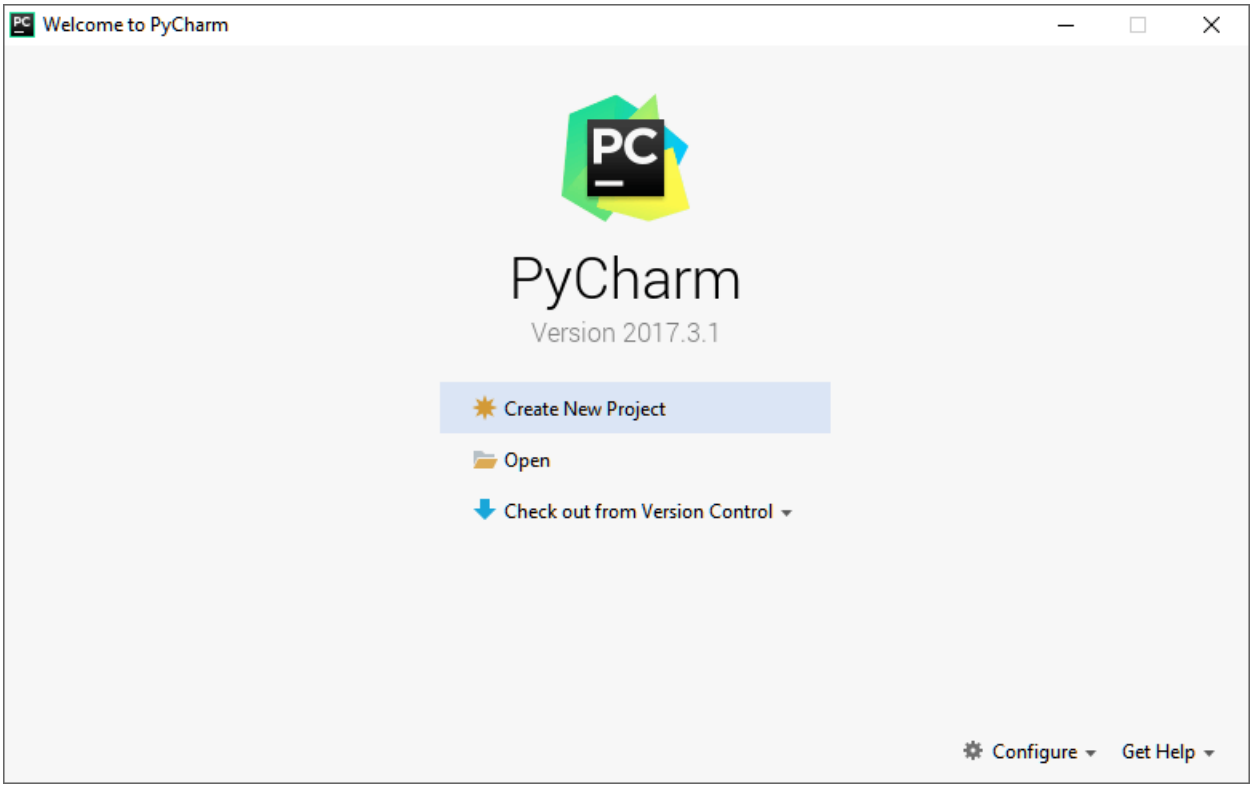
Step 6) Wait for the installation to finish.



Step 7) Once installation finished, you should receive a message screen that PyCharm is installed. If you want to go ahead and run it, click the "Run PyCharm Community Edition" box first and click "Finish".



Step 8) After you click on "Finish," the Following screen will appear.



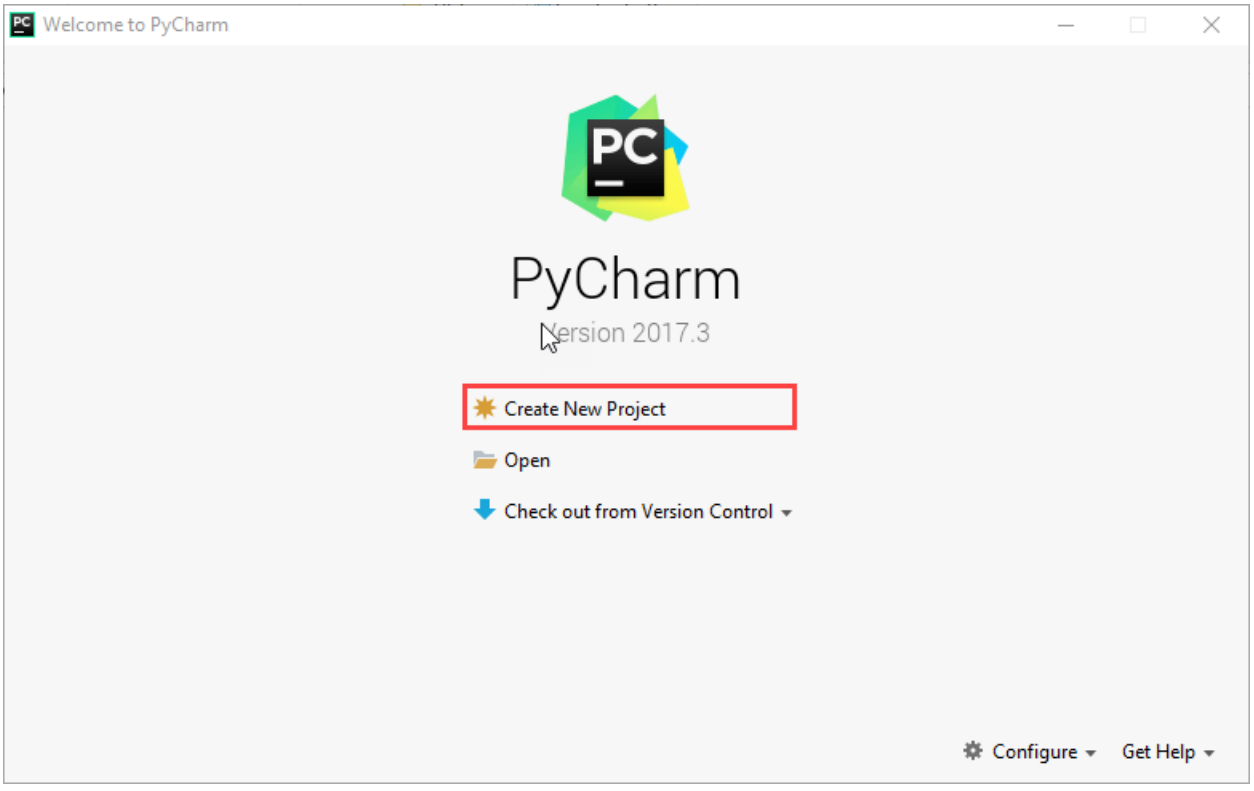
Chapter 3: Hello World: Create your First Python Program

In the last tutorial, we completed our Python installation and setup.

It's time to create your first program.

Creating First Program

Step 1) Open PyCharm Editor. You can see the introductory screen for PyCharm. To create a new project, click on "Create New Project".

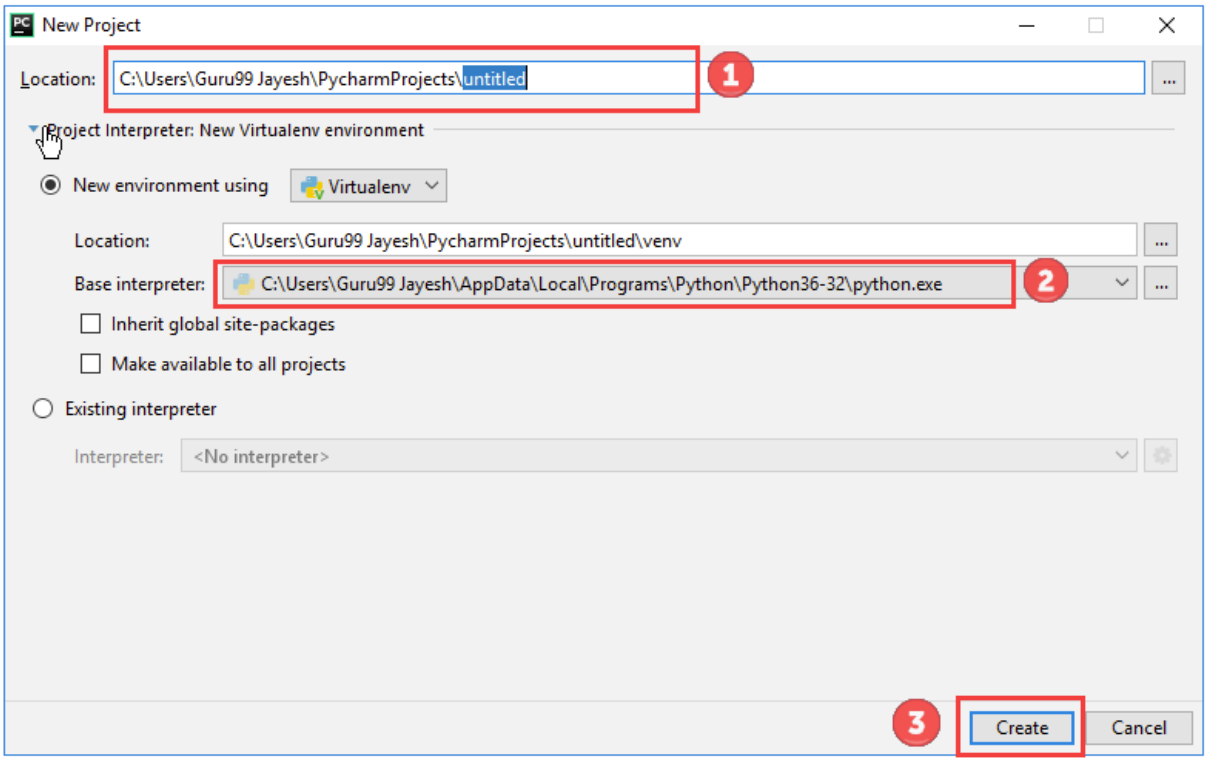


Step 2) You will need to select a location.

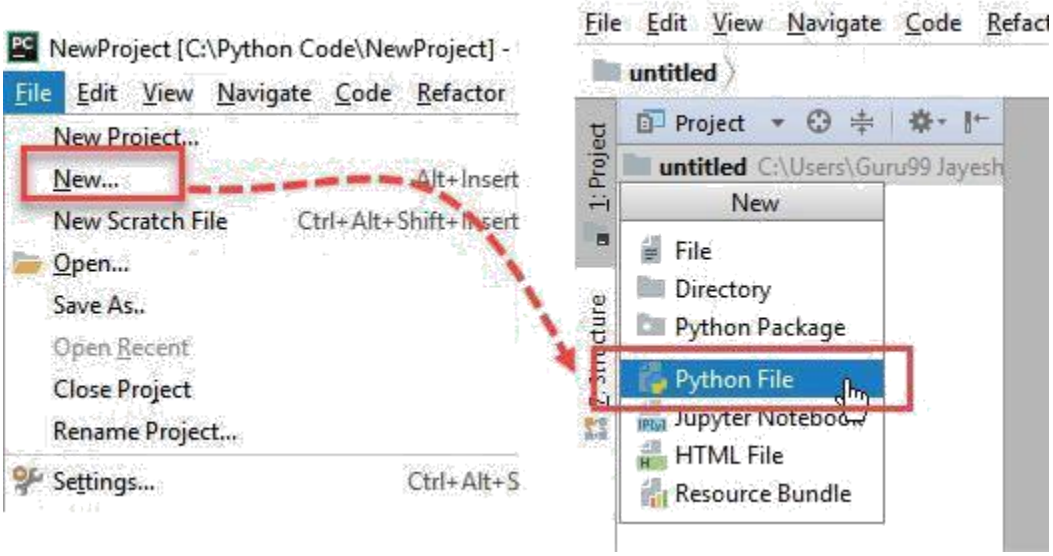
1. You can select the location where you want the project to be created. If you don't want to change location than keep it as it is but at least change the name from "untitled" to something more meaningful, like "FirstProject".

2. PyCharm should have found the Python interpreter you installed earlier.

3. Next Click the "Create" Button.



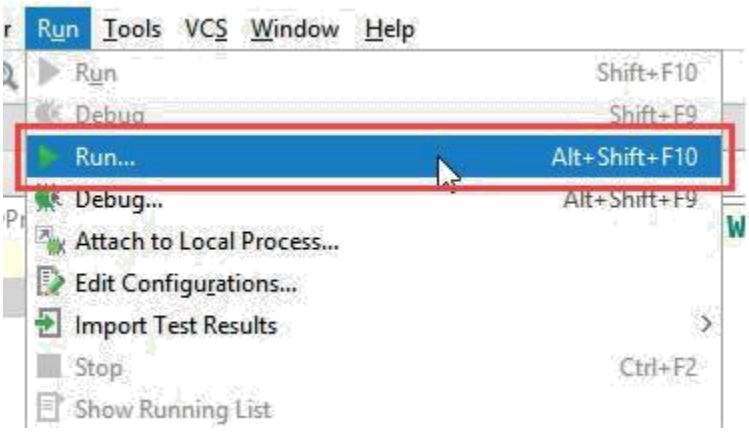
Step 3) Now Go up to the "File" menu and select "New". Next, select "Python File".



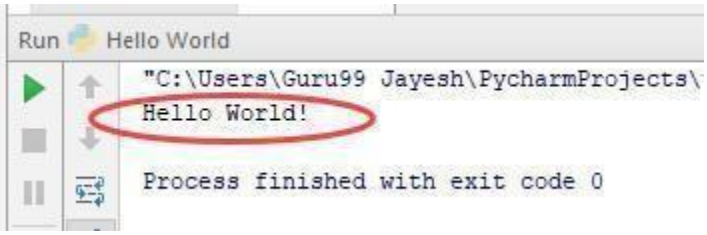
Step 4) A new pop up will appear. Now type the name of the file you want (Here we give "HelloWorld") and hit "OK".

Step 5) Now type a simple program - print ('Hello World!').

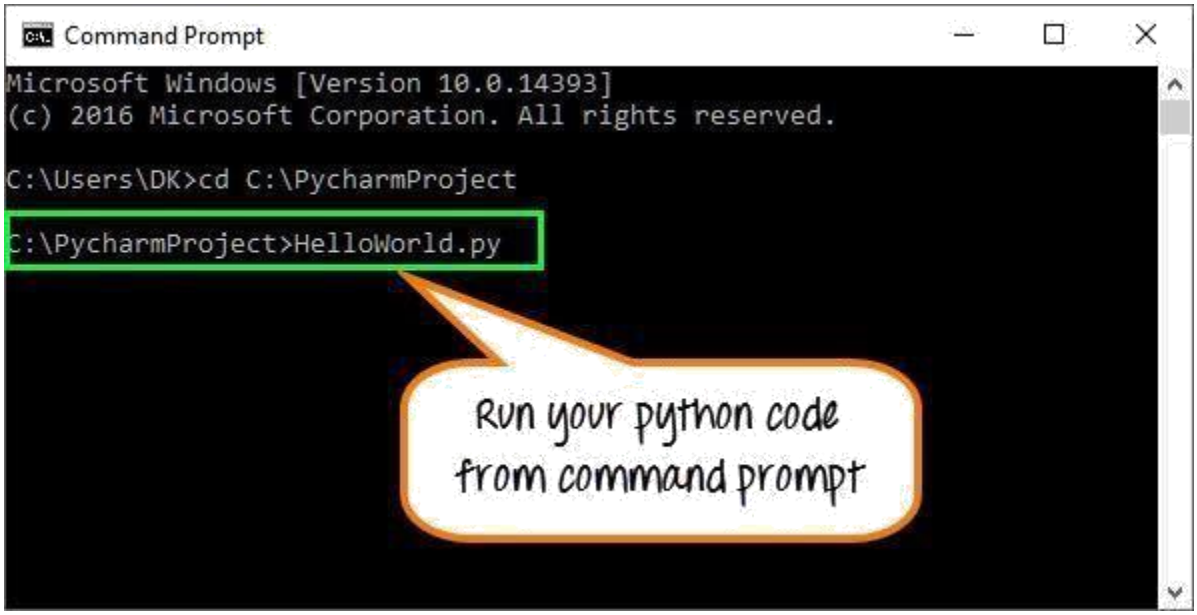
Step 6) Now Go up to the "Run" menu and select "Run" to run your program.



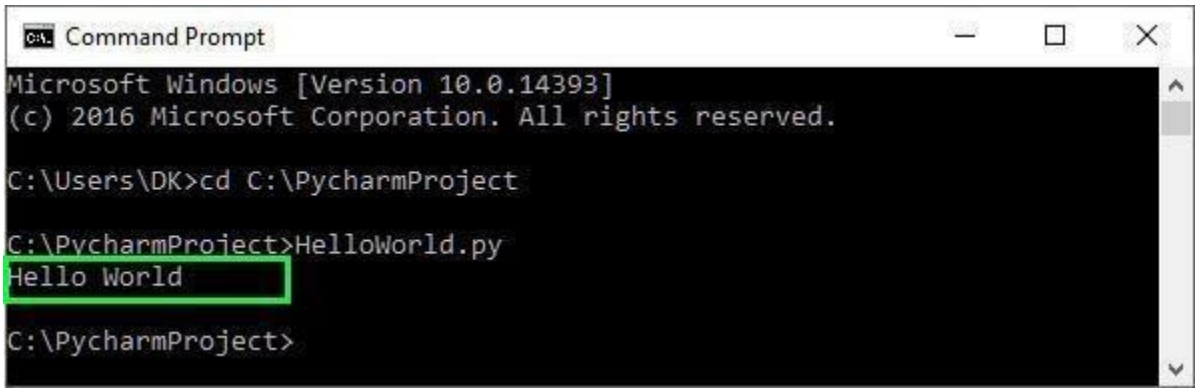
Step 7) You can see the output of your program at the bottom of the screen.



Step 8) Don't worry if you don't have Pycharm Editor installed, you can still run the code from the command prompt. Enter the correct path of a file in command prompt to run the program.



The output of the code would be



Step 9) If you are still not able to run the program, we have Python Editor for you.

Please run the given code at Python Online Editor

```
print("Hello World")
```

Buy Now $9.99