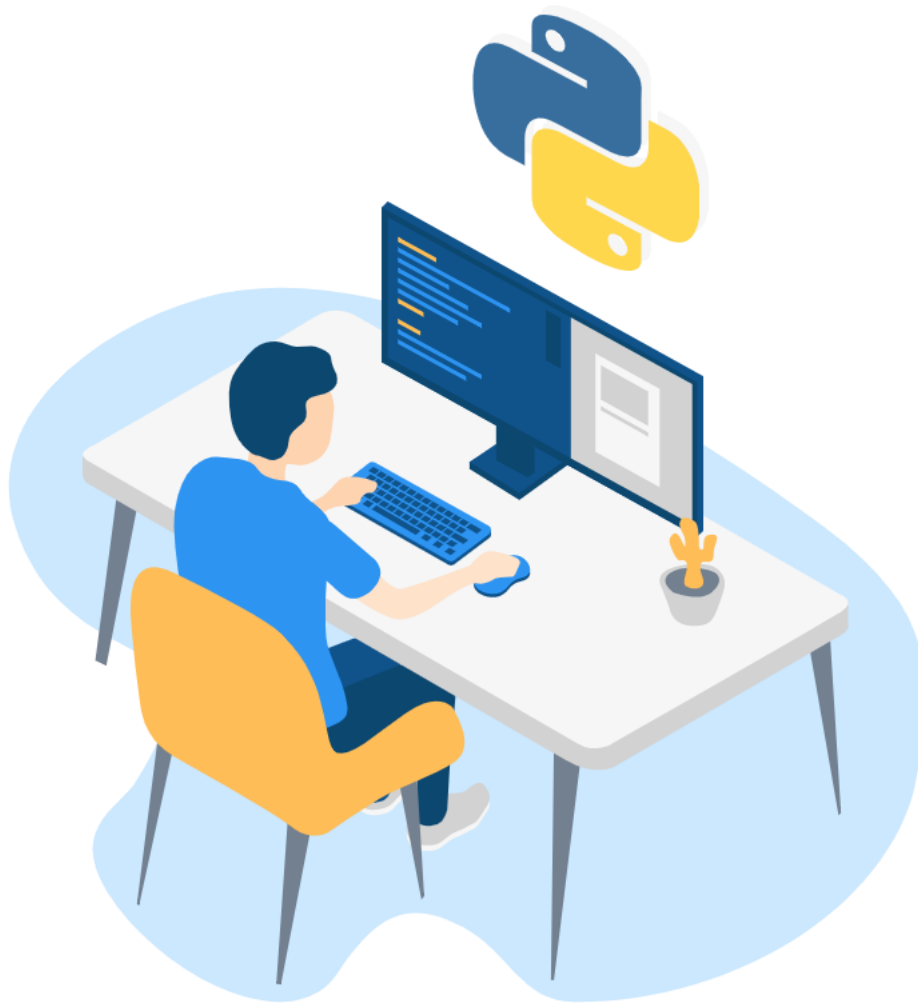


BAB 2

Struktur Dasar Bahasa Python



3.1 Tujuan

1. Dapat menulis, meng-compile dan menjalankan program python
2. Dapat menjelaskan struktur dasar bahasa python.
3. Dapat menjelaskan input output dan variabel

3.2 Pengantar

3.2.1 Bahasa Python

Python dikembangkan oleh Guido van Rossum pada tahun 1990 di CWI, Amsterdam sebagai kelanjutan dari bahasa pemrograman ABC. Versi terakhir yang dikeluarkan CWI adalah 1.2. Tahun 1995, Guido pindah ke CNRI sambil terus melanjutkan pengembangan Python. Versi terakhir yang dikeluarkan adalah 1.6. Tahun 2000, Guido dan para pengembang inti Python pindah ke BeOpen.com yang merupakan sebuah perusahaan komersial dan membentuk BeOpen PythonLabs. Python 2.0 dikeluarkan oleh BeOpen. Setelah mengeluarkan Python 2.0, Guido dan beberapa anggota tim PythonLabs pindah ke DigitalCreations.

Saat ini pengembangan Python terus dilakukan oleh sekumpulan pemrogram yang dikoordinir Guido dan Python Software Foundation. Python Software Foundation adalah sebuah organisasi non-profit yang dibentuk sebagai pemegang hak cipta intelektual Python sejak versi 2.1 dan dengan demikian mencegah Python dimiliki oleh perusahaan komersial.

Nama Python dipilih oleh Guido sebagai nama bahasa ciptaannya karena kecintaan Guido pada acara televisi Monty Python's Flying Circus. Oleh karena itu seringkali ungkapan-ungkapan khas dari acara tersebut seringkali muncul dalam korespondensi antar pengguna Python.

Python pun terus berkembang dalam penggunaannya, sehingga fitur-fitur baru dibutuhkan untuk dikembangkan. Versi 2.0 dirilis Oktober 2000 dengan beberapa pengembangan fitur termasuk *Garbage Collector* dan *Memory Management* yang juga menjadi fitur pada beberapa bahasa pemrograman modern lainnya, di antaranya Java dan C#.

Python 3.0 adalah versi perubahan mayor yang dirilis pada Desember 2008. Beberapa sintaksis/statement yang sebelumnya berjalan di versi 2.x, kini tidak lagi berjalan. Contohnya, fungsi print yang sebelumnya adalah statement di python 2.x, menjadi function di python 3.x.

Pada saat modul ini ditulis versi python paling baru adalah 3.8.5 yang dirilis pada 20 Juli 2020. Versi python tersebut yang akan kita pakai selama praktikum ini.

3.2.2 Librari pada Python

Penggunaan bahasa python semakin populer karena banyak librari yang tersedia. Librari ini menyediakan beberapa fungsionalitas baru dan membuat pengembang mudah dalam menciptakan suatu aplikasi. Beberapa librari tersebut antara lain

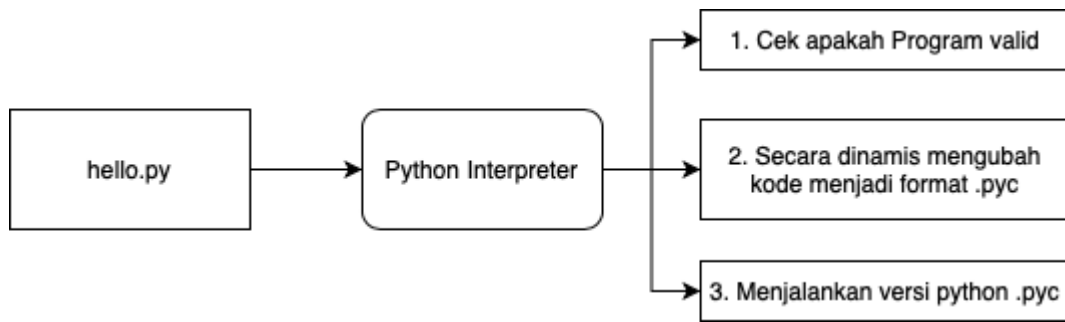
- Framework web seperti Django dan Flask
- Email client seperti smtplib
- Data visualisasi seperti Matplotlib dan PyOpenGL
- Machine learning seperti SKLearn dan Tensorflow

Jika kalian ingin belajar lebih jauh tentang librari yang tersedia pada python maka silakan kunjungi website <https://pymotw.com/3/> yang berisi berbagai macam librari yang dapat digunakan beserta penjelasan singkat dan cara penggunaannya.

3.2.3 Bagaimana Python Bekerja

Tidak seperti bahasa pemrograman lainnya, python bukan merupakan bahasa precompiled seperti halnya C++. Python sendiri termasuk dalam bahasa interpreter (penerjemah). Sebuah bahasa interpreter bekerja dengan melakukan eksekusi sejumlah kode yang ditulis dalam bahasa pemrograman tanpa perlu menyusunnya dalam bentuk bahasa mesin. Proses ini berbeda dengan bahasa compiler, dimana kode harus diubah terlebih dahulu ke dalam bahasa mesin sebelum dijalankan.

Interpreter pada python bekerja dengan mengubah kode bahasa python menjadi sebuah format baru yang lebih mudah dijalankan oleh mesin. Python menyimpan format baru ini dengan ekstensi .pyc (huruf c mengindikasikan format python yang telah dicompile) kemudian format baru tersebut dijalankan untuk menghasilkan suatu aplikasi. Ilustrasi proses ini dapat dilihat pada gambar 2.1



Gambar 2. 1 Proses Interpreter Python

Ada beberapa cara yang dapat dilakukan untuk menjalankan program python, beberapa diantaranya antara lain :

- Secara interaktif menggunakan python interpreter
- Disimpan dalam file .py kemudian dijalankan menggunakan command python
- Dijalankan melalui text editor / IDE (Integrated Development Environment)
- Dijalankan melalui IDE berbasis browser

Pada praktikum ini kita akan menggunakan Visual Studio Code dalam pengembangan program python

3.2.4 PyCharm

PyCharm merupakan IDE yang digunakan khusus untuk pengembangan aplikasi menggunakan bahasa Python. PyCharm dikembangkan oleh perusahaan bernama JetBrains, perusahaan yang juga membuat beberapa IDE lain seperti IntelliJ Idea dan Android Studio. Beberapa fitur dari PyCharm antara lain code analysis, graphical debugger, version control system, hingga support pengembangan web melalui Django.

PyCharm merupakan IDE yang dapat berjalan di Windows, macOS dan Linux. Tersedia dua versi yang dapat digunakan. Versi Professional dengan harga mulai \$199 pertahun dan versi Community yang dapat digunakan secara gratis.

3.2.5 Pengantar Python

Sebuah program Hello World pada python sangat sederhana jika dibandingkan dengan bahasa lain seperti C++ atau Java. Pada python kita cukup menulis satu baris kode seperti berikut

```
1. print('Hello World PTI')
```

Kita dapat menggunakan text editor atau IDE seperti Notepad, Visual Studio Code, Sublime dan PyCharm untuk membuat file python. Sebuah file python memiliki akhiran .py . Di dalam suatu file .py bisa memiliki satu atau ribuan baris kode python. Sebagai contoh kita dapat membuat sebuah file bernama main.py yang berisikan function print('Hello World PTI') di dalamnya.

Yang menjadi pertanyaan saat ini, darimana fungsi print() berasal?

Print() merupakan fungsi bawaan yang sudah tersedia pada python, berguna untuk mencetak output ke user. Fungsi bawaan merupakan fungsi yang sudah ada pada Python dan dikenali oleh interpreter sehingga kita tidak perlu membuat kode definisinya.

Sebuah fungsi Print() akan mencetak apapun yang ada diantara tanda kurung. Jika berisi String maka akan mencetak String. Jika berisi integer seperti angka 42 maka akan mencetak angka 42. Jika berisi angka desimal seperti 22.4 maka akan mencetak 22.4. Sehingga jika kita menjalankan main.py maka akan mencetak tulisan Hello World PTI.

3.2.6 Input, Output dan Variabel

Mari kita ubah kode program Hello World menjadi lebih interaktif seperti berikut ini

```
1. nama = input('Siapa nama kamu : ')
2. print ('hallo ', nama)
```

kode di atas jika dijalankan menampilkan output seperti berikut

```
Siapa nama kamu : arif
hallo arif
```

Pada baris 1 kita menggunakan fungsi bawaan yang bernama `input()`. Fungsi `input()` berguna untuk menerima inputan dari user. Sehingga jika baris 1 dijalankan maka akan menampilkan kata **Siapa nama kamu :** dan user harus menuliskan kata agar program dapat berjalan. Pada contoh diatas, user mengetik kata arif. Hasil dari inputan user akan disimpan dalam sebuah variabel bernama **nama**. Variabel merupakan suatu tempat pada memory yang digunakan untuk menyimpan suatu data. Pada contoh kode diatas, apapun yang diinputkan oleh user maka akan disimpan dalam sebuah variabel bernama **nama**. Kemudian pada baris ke-2 kita dapat menampilkan isi data variabel tersebut menggunakan fungsi `print()` yang diikuti dengan menuliskan nama variabelnya.

Isi dari suatu variabel dapat berubah-ubah sesuai dengan kegunaannya. Mari kita lihat kode berikut ini

```
1. nama = input('Siapa nama kamu : ')
2. print ('hallo ', nama)
3. nama = input ('Siapa nama Ayah kamu : ')
4. print ('Ayah kamu bernama ', nama)
```

Jika kita menginputkan kata arif dan deni maka hasil dari kode diatas akan seperti berikut

```
Siapa nama kamu : arif
hallo arif
Siapa nama Ayah kamu : deni
Ayah kamu bernama deni
```

Bisa kita lihat pada baris ke-2 dan ke-4, kita tetap memanggil variabel `nama` tapi jika kita lihat dari outputnya. Variabel `nama` memiliki isi yang berbeda. Hal ini terjadi karena bagian memori variabel `nama` yang sebelumnya berisi kata arif berubah menjadi deni.

Pada python, sebuah variabel memiliki sifat dynamic typing. Yaitu sebuah tipe variabel yang dapat berubah secara dinamis saat program berjalan. Sehingga kita tidak perlu memerlukan deklarasi variabel. Perhatikan isi variabel_satu pada kode berikut ini.

```
1. variabel_satu = 'Arif'
2. print(variabel_satu)
3. variabel_satu = 40
4. print(variabel_satu)
5. variabel_satu = True
6. print(variabel_satu)
```

Pada baris 1, variabel_satu berisi String 'Arif'. Pada baris 3, variabel_satu berisi integer 40. Pada baris ke-5, variabel_satu berisi True. Fleksibilitas variabel seperti ini yang merupakan salah satu keunggulan dari bahasa Python.

Penulisan variabel memiliki aturan sebagai berikut ini:

- Menggunakan huruf kecil (lowercase), gunakan format camelCase
- Gunakan kata deskriptif yang mewakili isi dari variabel. Hindari penamaan satu huruf seperti a dan b kecuali untuk penggunaan dalam looping.
- Jika menggunakan lebih dari satu kata, pisahkan menggunakan underscore (_)

Berikut ini merupakan contoh penamaan variabel yang benar pada python

- nama_saya, nama_kamu, user_name
- total, total_pengguna, nomor_rumah
- is_okay, is_correct

3.2.7 Operator Assignment

Perhatikan kode berikut ini, apa fungsi tanda = (sama dengan) yang berada diantara nama dan input()?

```
1. nama = input('Siapa nama kamu : ')
```

tanda = disebut dengan operator assignment. Tanda ini digunakan untuk memberi nilai pada suatu variabel. Pada contoh kode diatas maka, apapun hasil input dari user akan disimpan pada variabel nama.

3.2.8 Komentar

Komentar digunakan untuk menambahkan keterangan pada kode sehingga akan membantu siapapun yang membaca kode tersebut agar lebih mengerti. Suatu komentar pada kode tidak akan dieksekusi oleh interpreter. Pada Python, suatu komentar diawali dengan tanda #. Apapun yang berada setelah tanda # akan dianggap sebagai komenter dan tidak akan dieksekusi. Perhatikan kode berikut untuk penggunaan komentar pada Python

```
1. #mengambil inputan nama
2. nama = input('Siapa nama kamu : ')
3.
4. print ('hallo ', nama) #menampilkan variabel nama
```

3.3 Kegiatan Praktikum

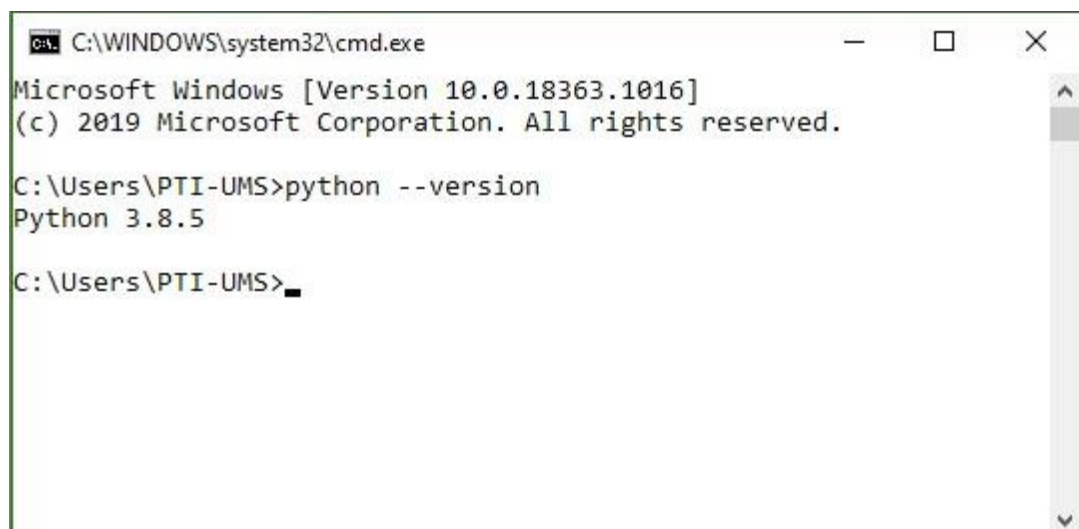
3.3.1 Kegiatan 1 : Menjalankan Python mode interaktif

1. Unduh python dari <https://www.python.org/downloads/windows/> . Pada saat modul ini ditulis versi paling baru adalah 3.8.5
2. Setelah file selesai diunduh kemudian lakukan instalasi. Pastikan mencentang **Add Python 3.8 to PATH** untuk menambahkan python dalam Environment Variables.



Gambar 3. 1 Instalasi Python

3. Buka command prompt kemudian ketik **python --version** untuk mengecek apakah python sudah terinstall dengan benar



Gambar 3.2 Cek apakah python sudah terinstal

4. Untuk masuk ke mode interaktif ketikkan **python** pada cmd, kemudian ketik **print("Hello World PTI")** untuk menampilkan tulisan **Hello World PTI**

```

C:\WINDOWS\system32\cmd.exe - python
Microsoft Windows [Version 10.0.18363.1016]
(c) 2019 Microsoft Corporation. All rights reserved.

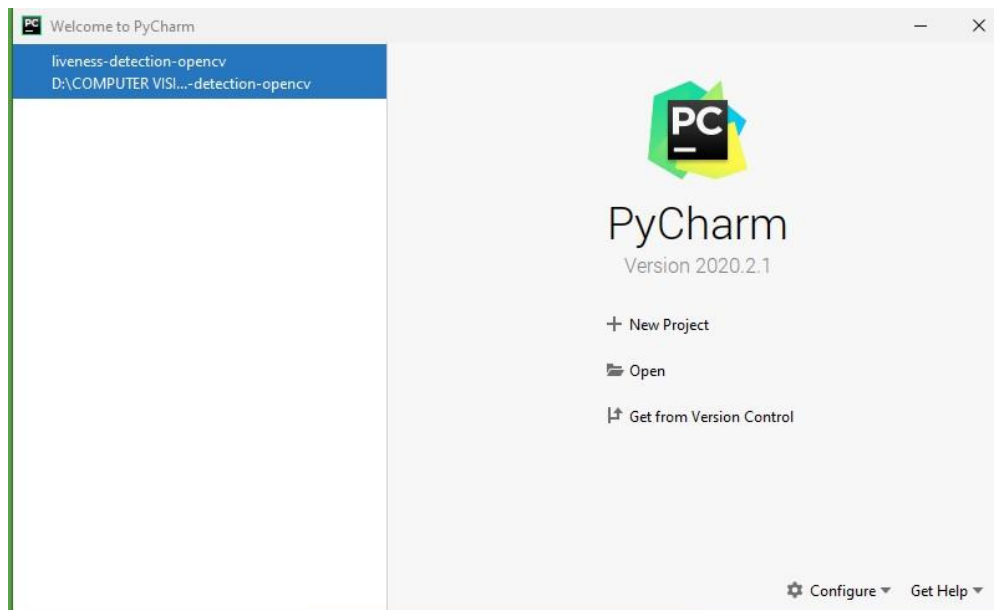
C:\Users\PTI-UMS>python
Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:43:08) [MSC v.1926 32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello World PTI")
Hello World PTI
>>> █

```

Gambar 3.3 Python mode interaktif

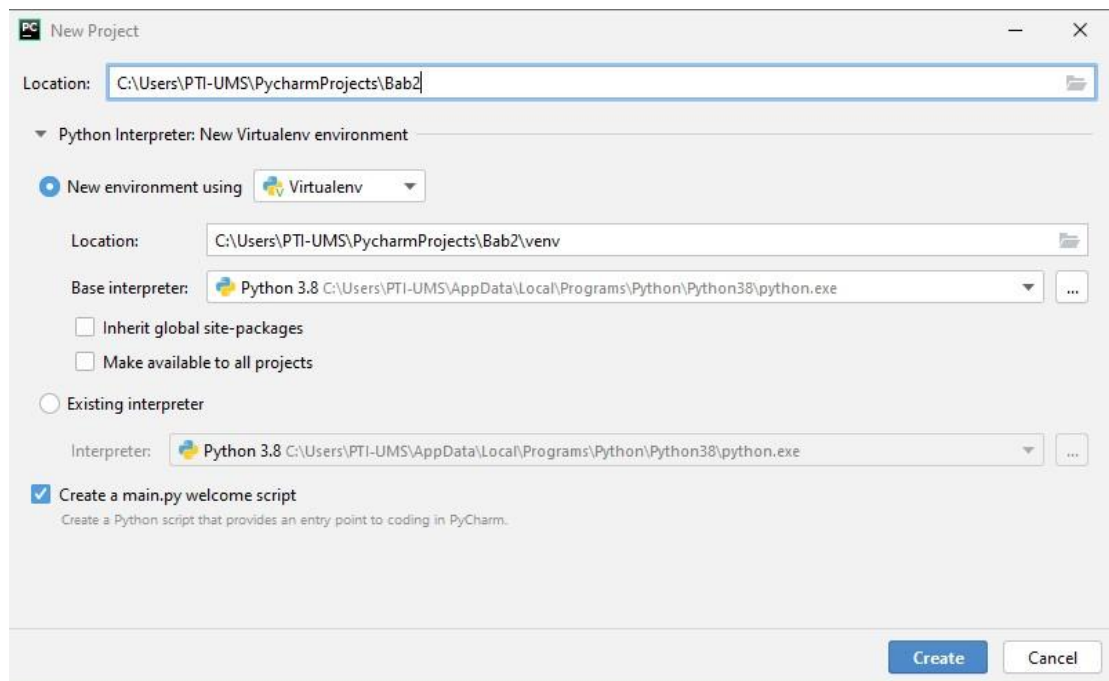
3.3.2 Kegiatan 2 : Menjalankan Python menggunakan IDE PyCharm

1. Unduh PyCharm dari <https://www.jetbrains.com/pycharm/download/> . Terdapat dua pilihan download yaitu versi Professional dan Community. Untuk praktikum ini silakan download versi **Community**. Setelah terdownload kemudian install PyCharm pada komputer.
2. Jalankan PyCharm kemudian pilih New Project



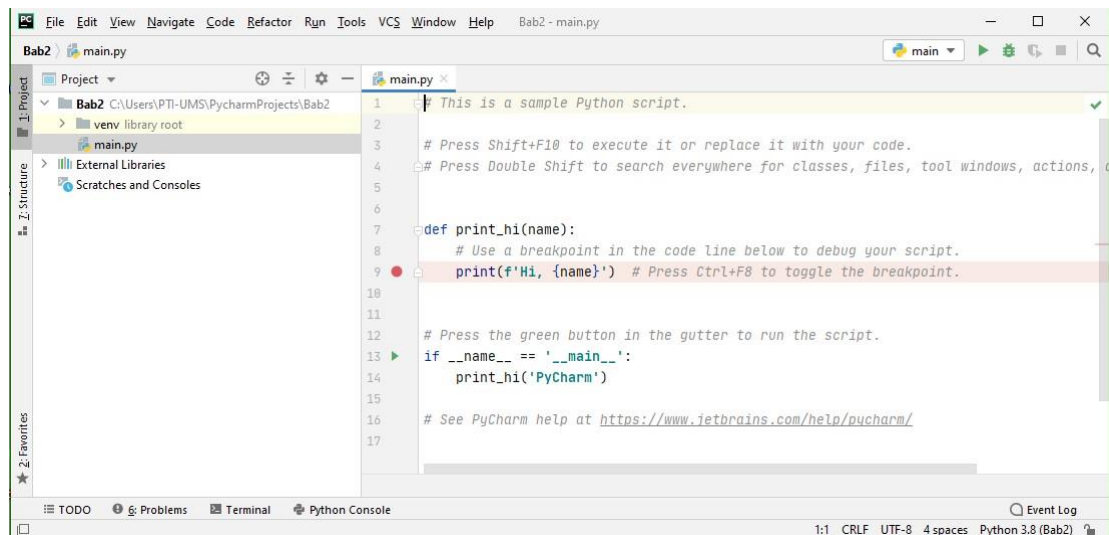
Gambar 3.4 Tampilan Welcome PyCharm

3. Sesuaikan Location pada folder yang akan digunakan sebagai tempat penyimpanan project, kemudian klik **Create**



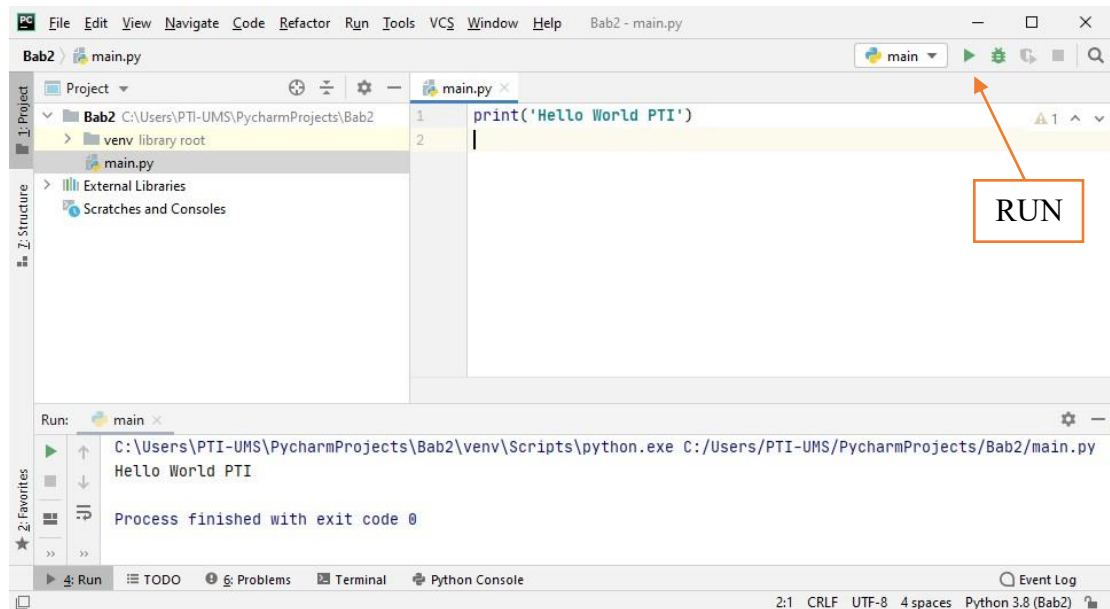
Gambar 3.5 Tampilan New Project

4. Gambar 2.7 merupakan tampilan UI dari PyCharm



Gambar 3.6 Tampilan UI PyCharm

5. Hapus semua kode pada main.py kemudian ketik `print('Hello World PTI')`, kemudian klik logo **RUN** untuk menjalankan kode tersebut



Gambar 3.7 Run Kode Python

3.3.3 Kegiatan 3 : Penggunaan Input Output

1. Buat sebuah file program baru kemudian tulis kode program berikut ini

```
1. #mengambil inputan nama
2. nama = input('Siapa nama kamu : ')
3.
4. #mengambil inputan nama ayah
5. umur = input ('Berapa umur kamu: ')
6.
7. #mengambil inputan alamat
8. alamat = input ('Dimana alamatmu : ')
9.
10. #menampilkan variabel
11. print('hallo ', nama)
12. print('umur kamu ', umur)
13. print('alamat kamu ', alamat)
```

2. Amati hasilnya kemudian tulis analisis singkat mengenai kegiatan 2 ini

3.3.4 Kegiatan 4 : Penggunaan Variabel

1. Buat sebuah file program baru kemudian tulis kode program berikut ini

```
1. print("Menghitung Luas dan Keliling Persegi")
2. #mengambil inputan panjang
3. panjang = input('Masukkan nilai Panjang: ')
4.
5. #mengambil inputan lebar
6. lebar = input ('Masukkan nilai Lebar: ')
7.
8. #menghitung luas dan keliling
9. luas = int(panjang) * int(lebar)
10. keliling = 2 * (int(panjang) + int(lebar))
11.
12. #menampilkan hasil
13. print ('Luas Persegi adalah ', luas , ' sedangkan kelilingnya adalah ',
    keliling)
```

2. Amati hasilnya kemudian tulis analisis singkat mengenai kegiatan 3 ini

3.4 Tugas

1. Buatlah program untuk menampilkan data diri anda dengan menggunakan variabel untuk menempatkan data nama, nim, dan angkatan.
2. Buatlah program untuk melakukan perhitungan akar pangkat tiga dari sebuah angka, dengan angka dan hasilnya disimpan dalam suatu variabel.