

Tutorial Setup Chatbot Alkitabiah di OS Windows

Prerequisite

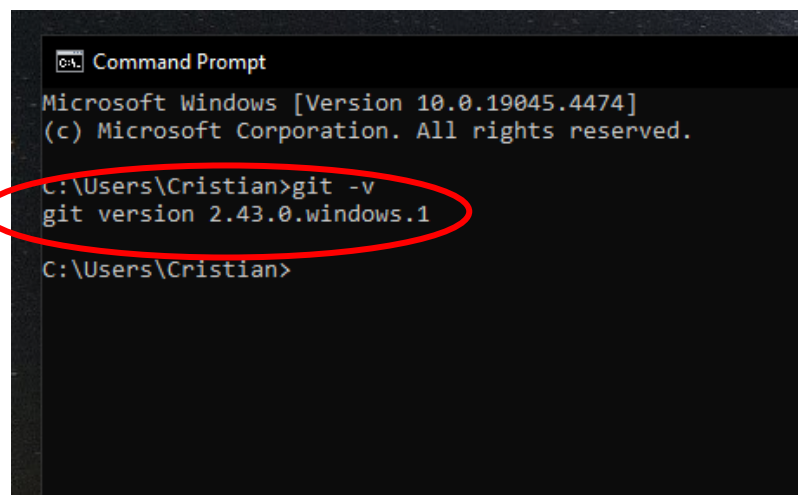
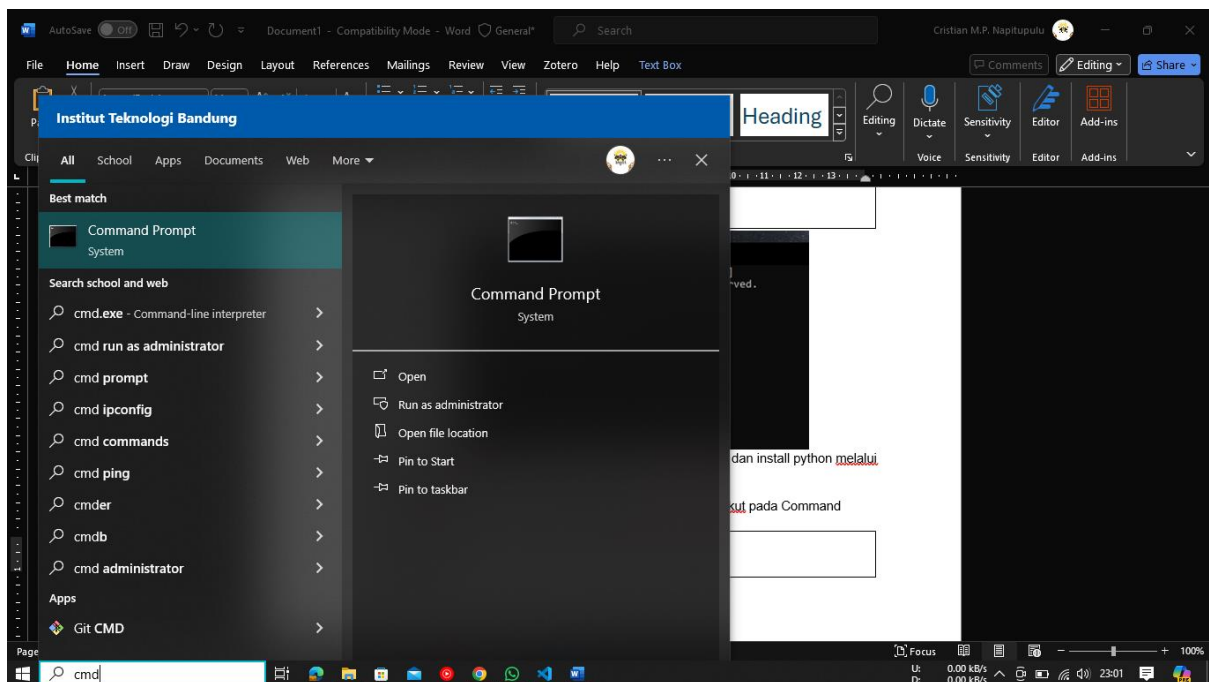
Note: Lakukan dalam kondisi terhubung internet.

1. Windows sudah terinstall git. Unduh dan install git melalui link berikut.

<https://git-scm.com/download/win>

Periksa instalasi git dengan memasukkan perintah berikut pada Command Prompt:

```
git -v
```

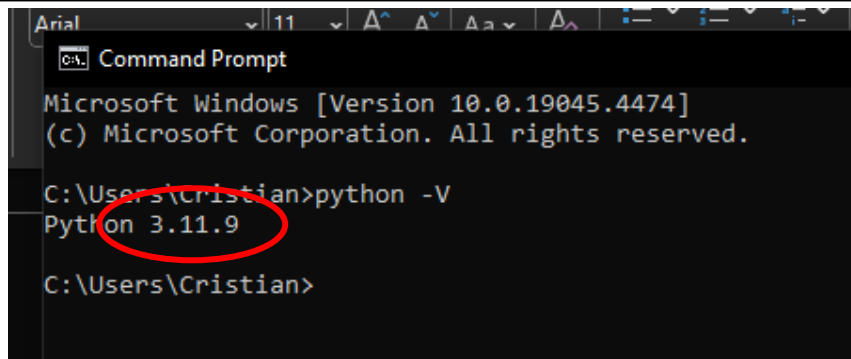


2. Windows sudah terinstall python minimal versi 3.10. Unduh dan install python melalui link berikut.

<https://www.python.org/downloads/windows/>

Periksa instalasi python dengan memasukkan perintah berikut pada Command Prompt dan pastikan bahwa versi python sudah minimal 3.10:

```
python -V
```



```
Command Prompt
Microsoft Windows [Version 10.0.19045.4474]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Cristian>python -V
Python 3.11.9

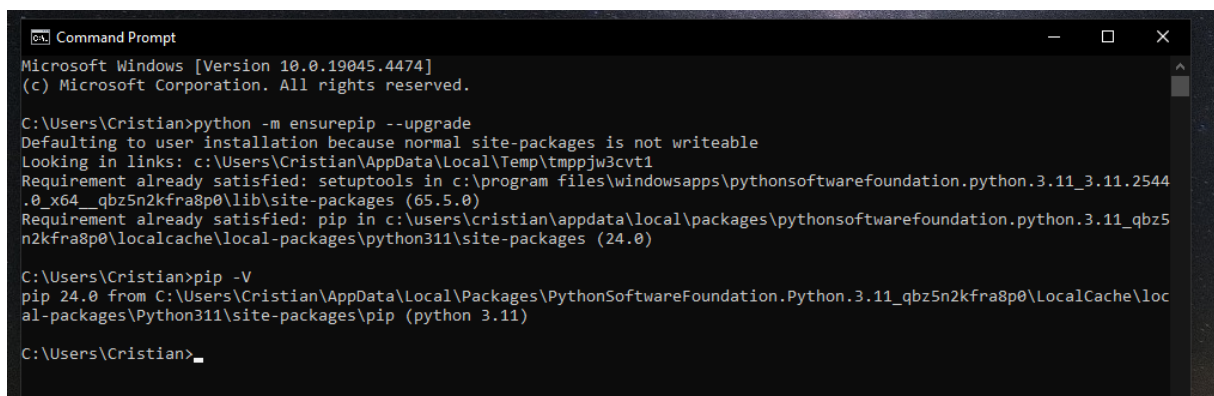
C:\Users\Cristian>
```

3. Windows sudah terinstal python pip. Install python pip dengan memasukkan perintah berikut pada Command Prompt:

```
python -m ensurepip --upgrade
```

Periksa instalasi python pip dengan memasukkan perintah berikut pada Command Prompt:

```
pip -V
```



```
Command Prompt
Microsoft Windows [Version 10.0.19045.4474]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Cristian>python -m ensurepip --upgrade
Defaulting to user installation because normal site-packages is not writeable
Looking in links: c:\Users\Cristian\AppData\Local\Temp\tmppjw3cvt1
Requirement already satisfied: setuptools in c:\program files\windowsapps\pythonsoftwarefoundation.python.3.11_3.11.2544
_x64_qbz5n2kfra8p0\lib\site-packages (65.5.0)
Requirement already satisfied: pip in c:\users\cristian\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5
n2kfra8p0\localcache\local-packages\python311\site-packages (24.0)

C:\Users\Cristian>pip -V
pip 24.0 from C:\Users\Cristian\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.11_qbz5n2kfra8p0\LocalCache\loc
al-packages\Python311\site-packages\pip (python 3.11)

C:\Users\Cristian>
```

4. Windows sudah terinstal modul virtualenv dari pip. Install modul virtualenv dengan memasukkan perintah berikut pada Command Prompt:

```
pip install virtualenv
```

Periksa instalasi virtualenv dengan memasukkan perintah berikut pada Command Prompt:

```
python -m virtualenv --version
```

```
Command Prompt
Microsoft Windows [Version 10.0.19045.4474]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Cristian>pip install virtualenv
Requirement already satisfied: virtualenv in c:\users\cristian\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (20.26.2)
Requirement already satisfied: distlib<1,>=0.3.7 in c:\users\cristian\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from virtualenv) (0.3.8)
Requirement already satisfied: filelock<4,>=3.12.2 in c:\users\cristian\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from virtualenv) (3.13.4)
Requirement already satisfied: platformdirs<5,>=3.9.1 in c:\users\cristian\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from virtualenv) (4.1.0)

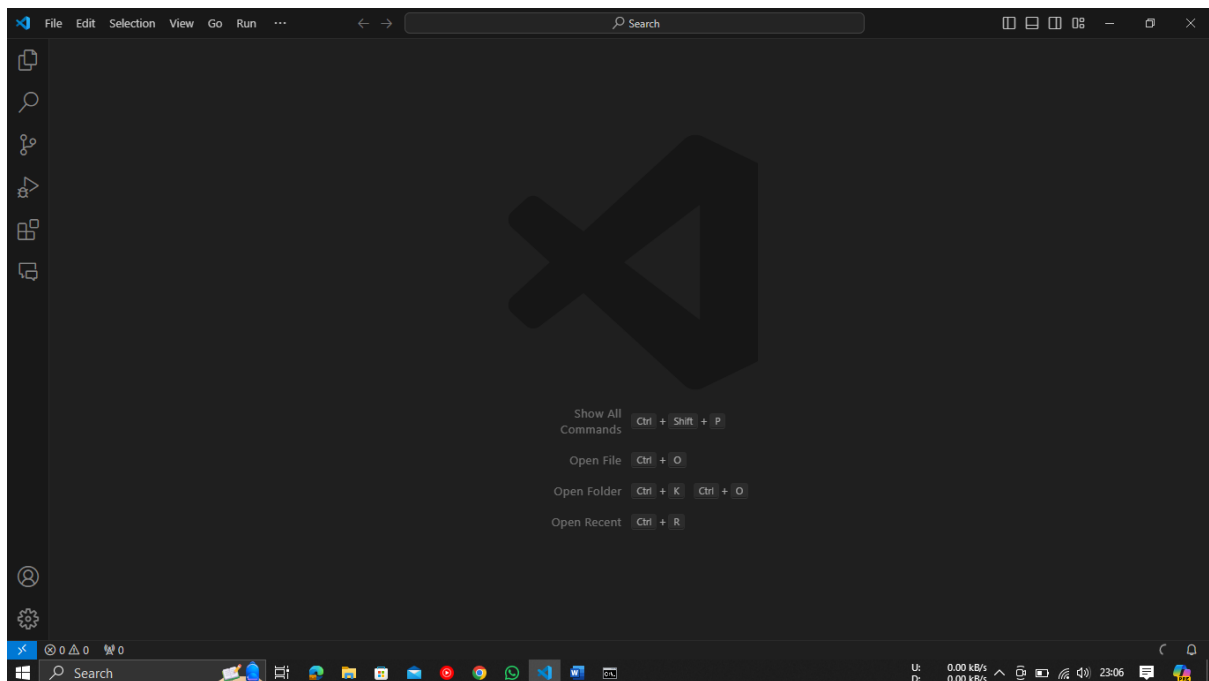
C:\Users\Cristian>python -m virtualenv --version
virtualenv 20.26.2 from C:\Users\Cristian\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.11_qbz5n2kfra8p0\LocalCache\local-packages\Python311\site-packages\virtualenv\__init__.py

C:\Users\Cristian>
```

5. Sudah menginstall software Visual Studio Code (opsional, dapat diganti dengan text editor lain). Unduh dan install VS Code melalui link berikut.

<https://code.visualstudio.com/Download>

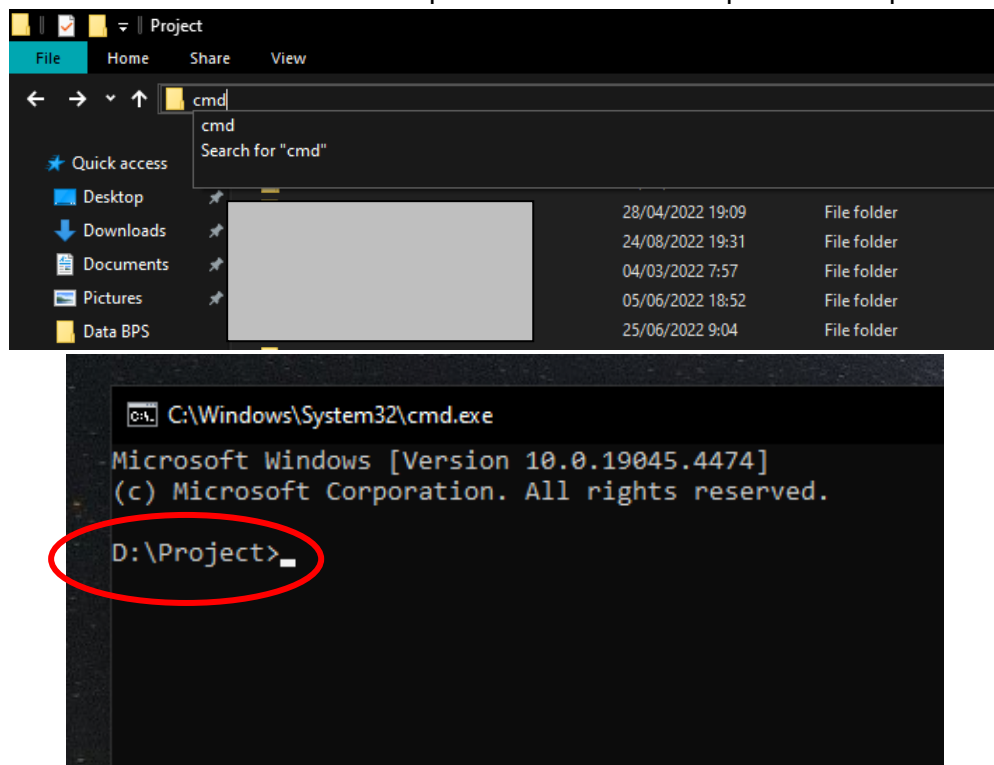
Jika VS Code sudah terinstal, buka aplikasi VS Code hingga terlihat tampilan berikut.



6. Selesai.

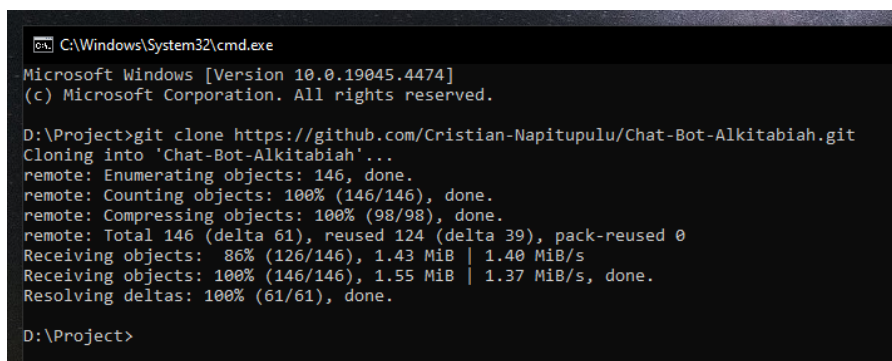
Clone Repository Chat Bot Alkitabiah

1. Pada file explorer, masuk ke folder tujuan untuk menyimpan repository Chat Bot Alkitabiah dan buka Command Prompt melalui address bar pada File Explorer.



2. Clone repository Chat Bot Alkitabiah dengan memasukkan perintah berikut pada Command Prompt:

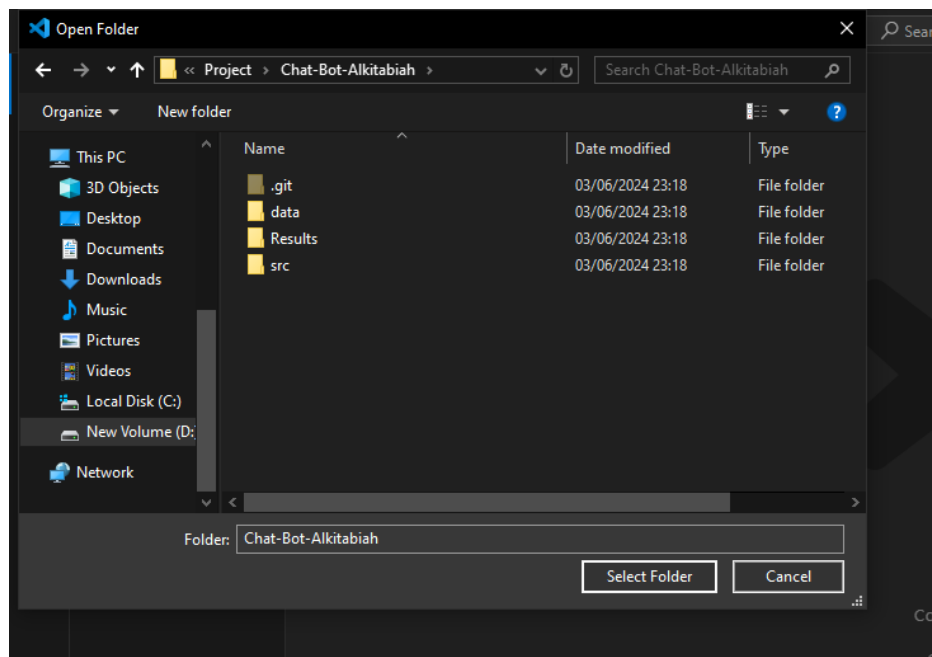
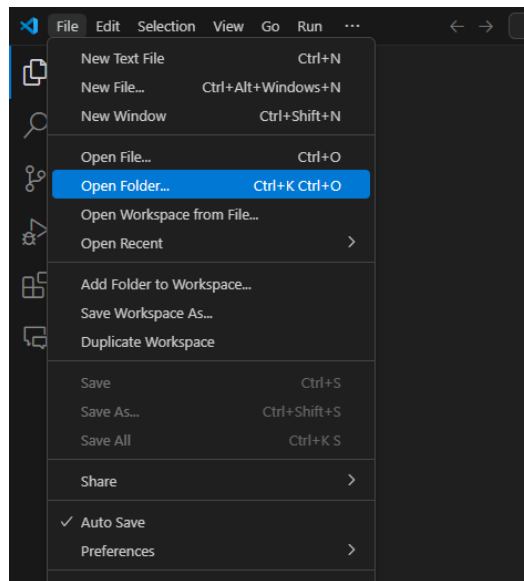
```
git clone https://github.com/Cristian-Napitupulu/Chat-Bot-Alkitabiah.git
```

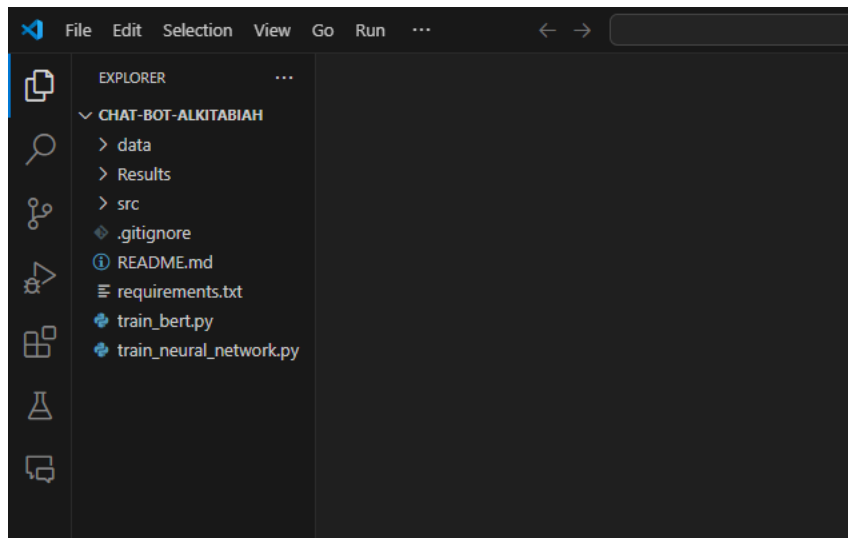


3. Selesai.

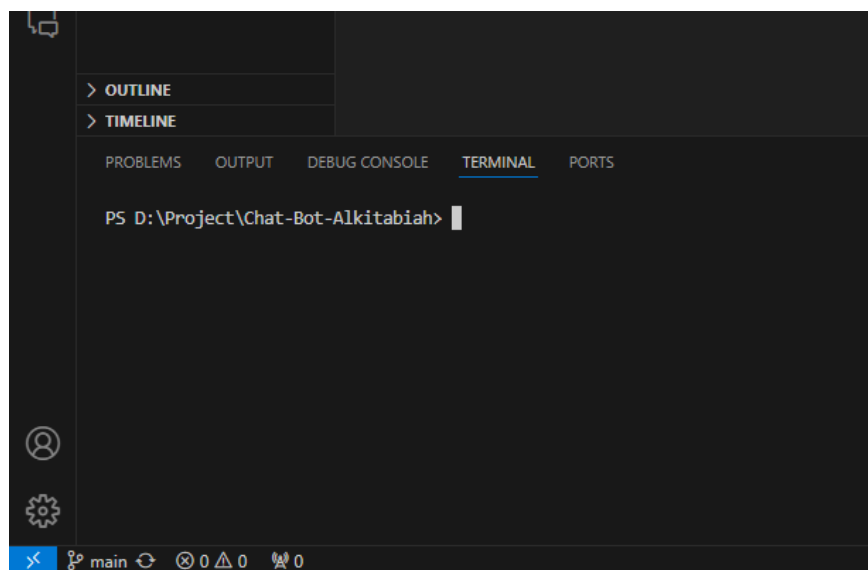
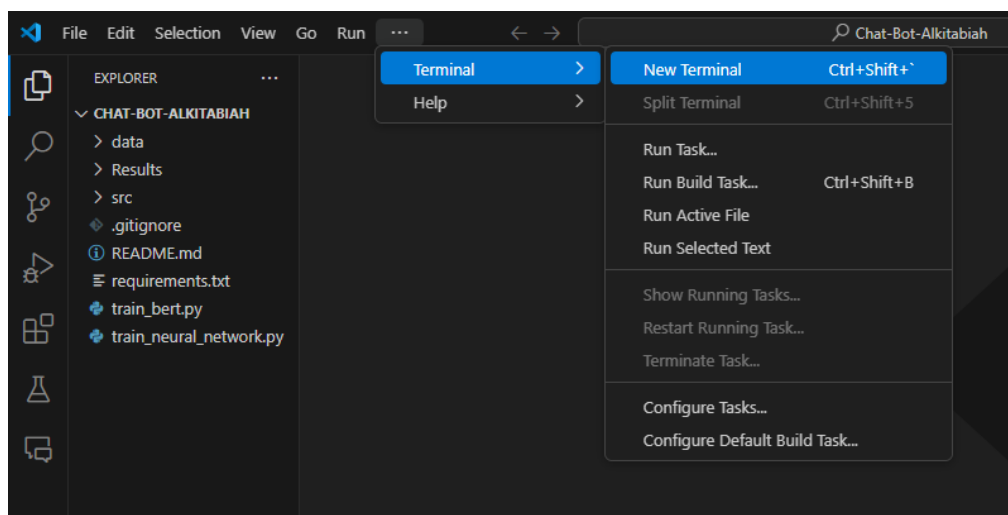
Setup Environment Chat Bot Alkitabiah

1. Masuk ke aplikasi VS Code lalu klik File>Open Folder. Lalu buka folder Chat-Bot-Alkitabiah yang sudah di clone sebelumnya.

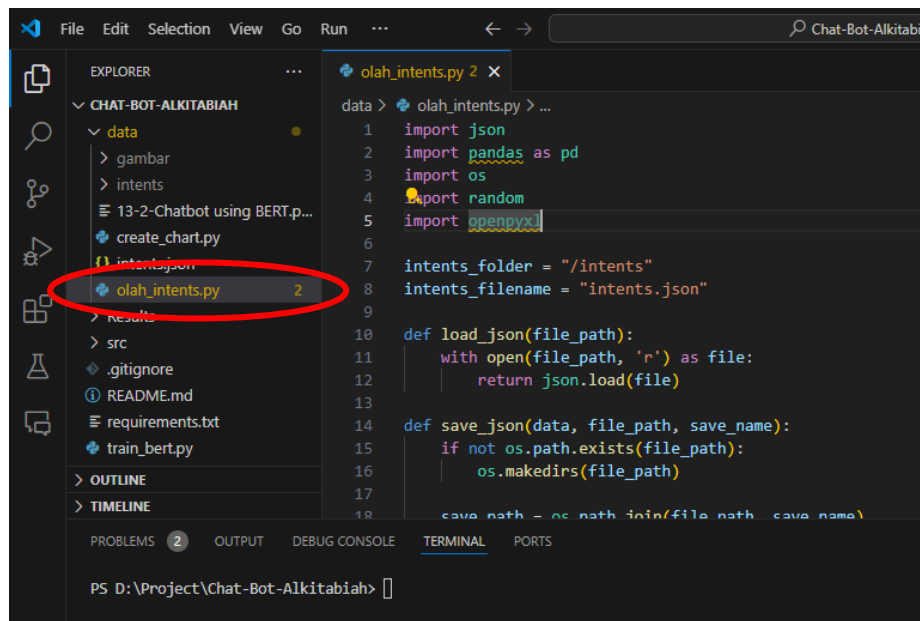




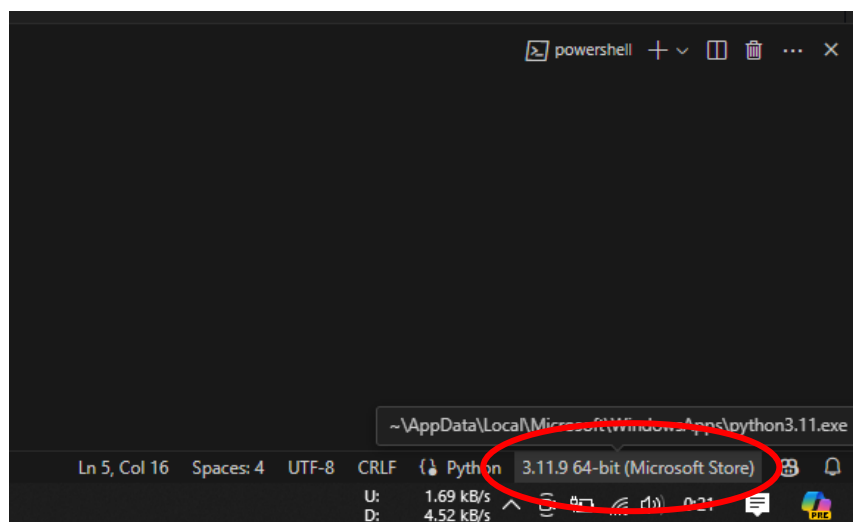
2. Buka terminal baru pada aplikasi VS Code



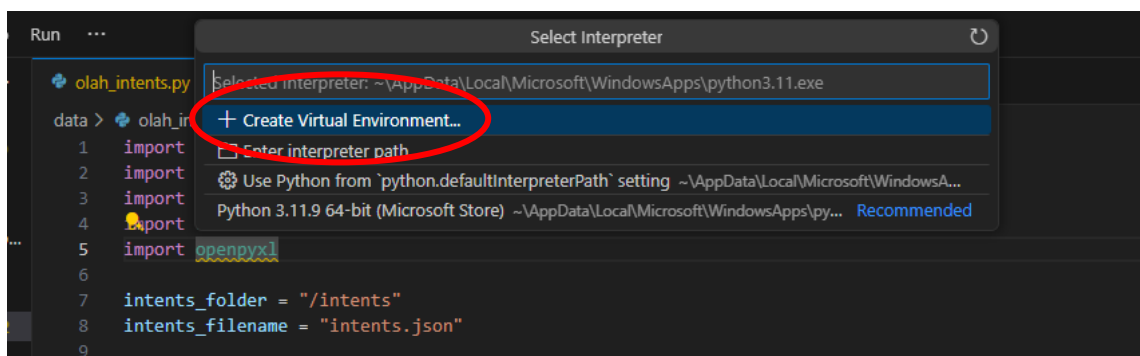
3. Buka file /data/olah_intents.py



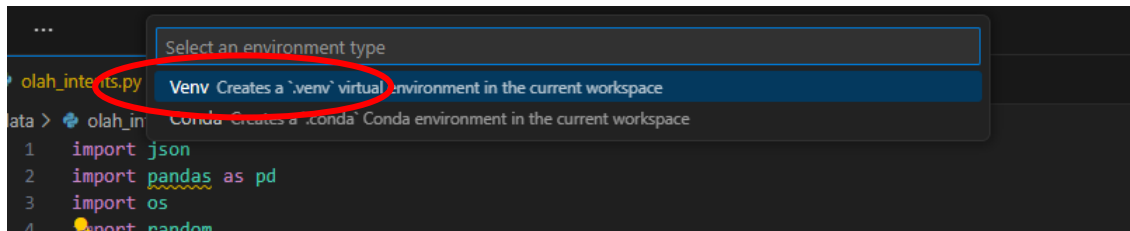
4. Kemudian, ubah python interpreter untuk Chat Bot Alkitabiah dengan cara klik tombol di pojok kanan bawah.



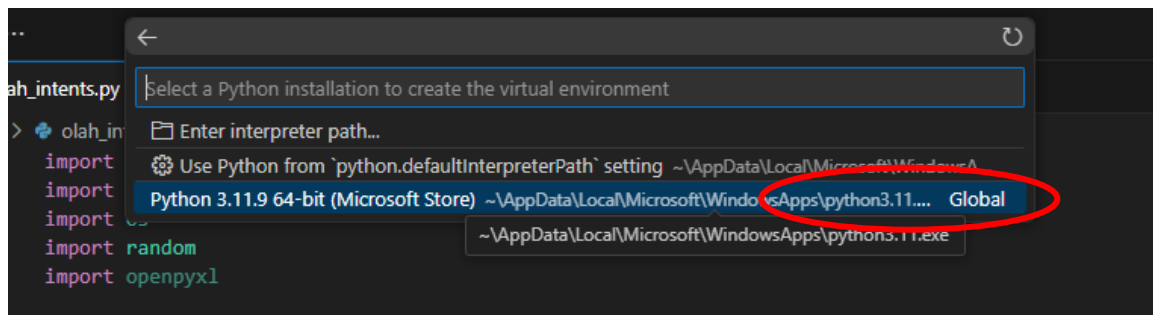
Kemudian pilih opsi "Create Virtual Environment"



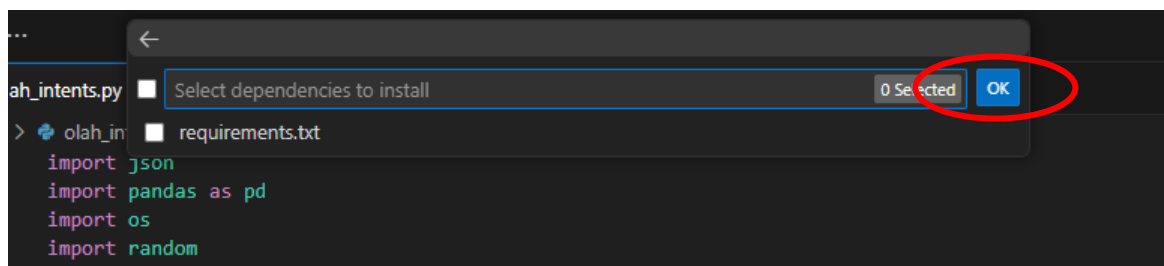
Lanjutkan dengan memilih opsi “venv”



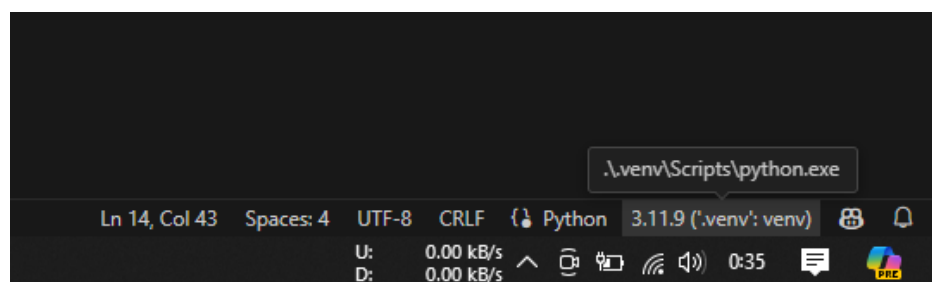
Kemudian pilih python Global yang ada pada komputer sebagai base.



Lanjutkan dengan klik “OK” **TANPA** mencentang apa pun pada tampilan ini.

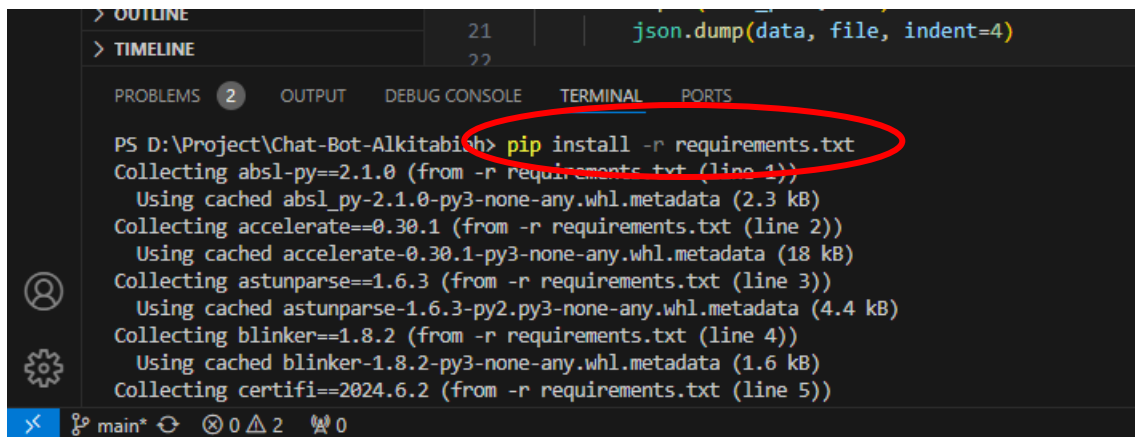


5. Tutup aplikasi VS Code dan buka kembali. Pastikan bahwa python interpreter yang terpilih berasal dari “.venv”



6. Selanjutnya install semua modul yang diperlukan oleh Chat Bot Alkitabiah dengan memasukkan perintah berikut pada terminal VS Code:

```
pip install -r requirements.txt
```

The screenshot shows a VS Code interface with a terminal window open. The terminal title is 'main*'. The command prompt is 'PS D:\Project\Chat-Bot-Alkitabiah>'. The command entered is 'pip install -r requirements.txt', which is circled in red. The output shows the installation of several packages from a requirements file, including absl-py, accelerate, astunparse, blinker, and certifi. The status bar at the bottom indicates 'main*' with 0 errors, 2 warnings, and 0 info messages.

```
PS D:\Project\Chat-Bot-Alkitabiah> pip install -r requirements.txt
Collecting absl-py==2.1.0 (from -r requirements.txt (line 1))
  Using cached absl_py-2.1.0-py3-none-any.whl.metadata (2.3 kB)
Collecting accelerate==0.30.1 (from -r requirements.txt (line 2))
  Using cached accelerate-0.30.1-py3-none-any.whl.metadata (18 kB)
Collecting astunparse==1.6.3 (from -r requirements.txt (line 3))
  Using cached astunparse-1.6.3-py2.py3-none-any.whl.metadata (4.4 kB)
Collecting blinker==1.8.2 (from -r requirements.txt (line 4))
  Using cached blinker-1.8.2-py3-none-any.whl.metadata (1.6 kB)
Collecting certifi==2024.6.2 (from -r requirements.txt (line 5))
```

7. Selesai.

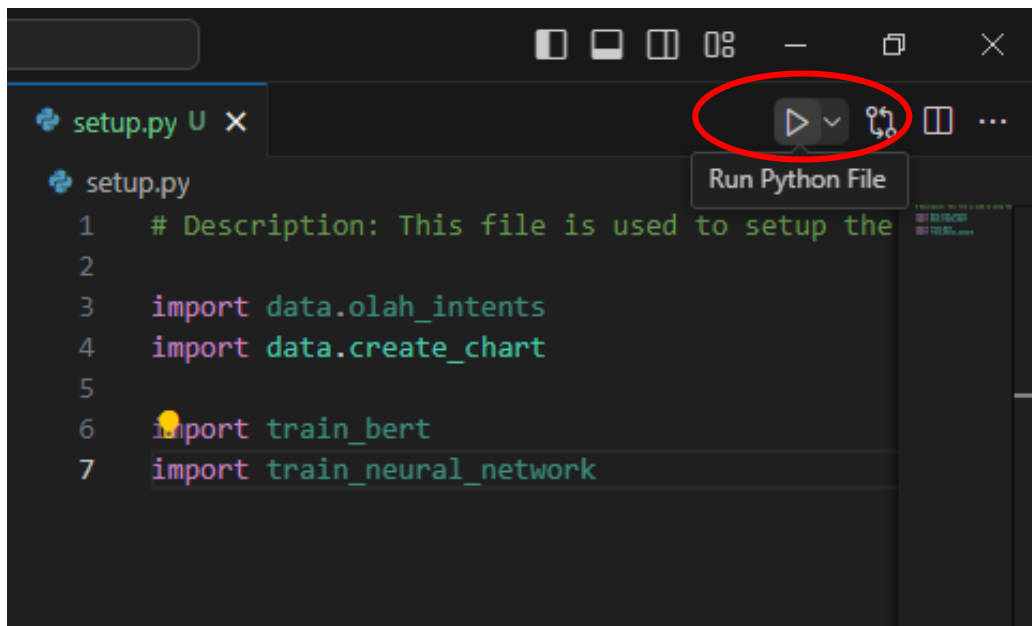
Menjalankan program Chat Bot Alkitabiah

Program Chat Bot Alkitabiah terbagi atas beberapa bagian:

- a) Program untuk mengolah dataset, terdiri atas:
 - i) `olah_intents.py`
 - ii) `create_chart.py`
- b) Program untuk training model, terdiri atas:
 - i) `train_bert.py`
 - ii) `train_neural_network.py`
- c) Program untuk deploy model dan user interface (UI), terdiri atas:
 - i) `bert.py`
Hanya deploy model BERT dengan beberapa sampel uji coba
 - ii) `neural_network.py`
Hanya deploy model Neural Network dengan beberapa sampel uji coba
 - iii) `app.py`
Untuk menjalankan website sebagai UI dan bisa memilih model dan perangkat proses yang digunakan

Sebelum bisa menggunakan Chat Bot Alkitabiah sepenuhnya, program bagian (a) dan (b) harus dijalankan terlebih dahulu secara berurutan.

Untuk mempermudah proses inialisasi, buka program `setup.py` dan jalankan program tersebut dengan meng-klik tombol “play” di pojok kanan atas pada aplikasi VS Code.



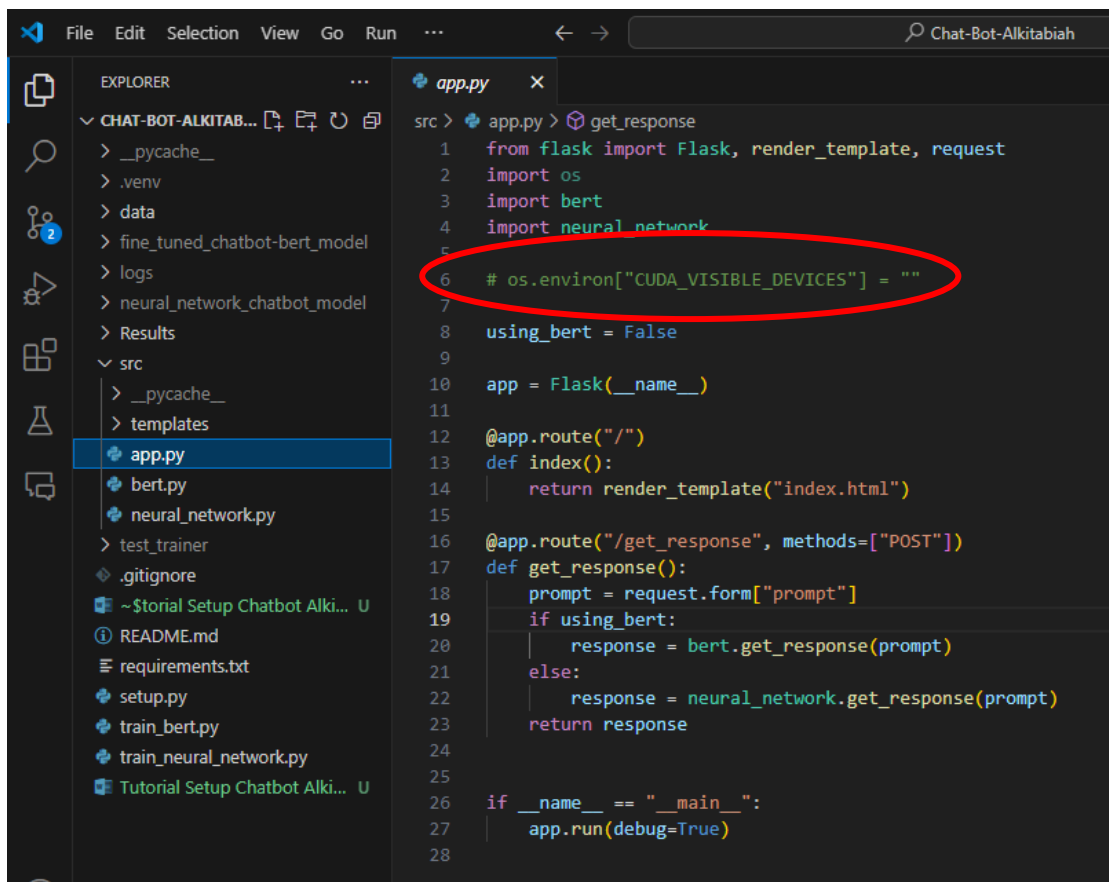
Setelah selesai menjalankan program `setup.py`, silahkan mencoba program bagian (c).

Pada tutorial ini, akan dijelaskan lebih lanjut mengenai program “app.py”.

1. Untuk mengatur penggunaan CPU atau GPU pada model, berikan/hilangkan “comment” pada baris:

```
# os.environ["CUDA_VISIBLE_DEVICES"] = ""
```

Menghilangkan “comment” akan mengaktifkan baris tersebut yang berfungsi untuk “menyembunyikan” GPU dari program sehingga program akan berjalan pada CPU. Sebaliknya, jika “comment” diberikan, maka akan menonaktifkan baris tersebut sehingga GPU “terlihat” oleh program. Program akan berjalan pada GPU selama perangkat komputer yang digunakan juga memiliki GPU dan driver GPU yang terinstall.



```
File Edit Selection View Go Run ... Chat-Bot-Alkitabiah
EXPLORER
CHAT-BOT-ALKITAB...
  > __pycache__
  > .venv
  > data
  > fine_tuned_chatbot-bert_model
  > logs
  > neural_network_chatbot_model
  > Results
  > src
    > __pycache__
    > templates
    app.py
    bert.py
    neural_network.py
  > test_trainer
  .gitignore
  ~$torial Setup Chatbot Alki... U
  README.md
  requirements.txt
  setup.py
  train_bert.py
  train_neural_network.py
  Tutorial Setup Chatbot Alki... U
src > app.py > get_response
1 from flask import Flask, render_template, request
2 import os
3 import bert
4 import neural_network
5
6 # os.environ["CUDA_VISIBLE_DEVICES"] = ""
7
8 using_bert = False
9
10 app = Flask(__name__)
11
12 @app.route("/")
13 def index():
14     return render_template("index.html")
15
16 @app.route("/get_response", methods=["POST"])
17 def get_response():
18     prompt = request.form["prompt"]
19     if using_bert:
20         response = bert.get_response(prompt)
21     else:
22         response = neural_network.get_response(prompt)
23     return response
24
25
26 if __name__ == "__main__":
27     app.run(debug=True)
28
```

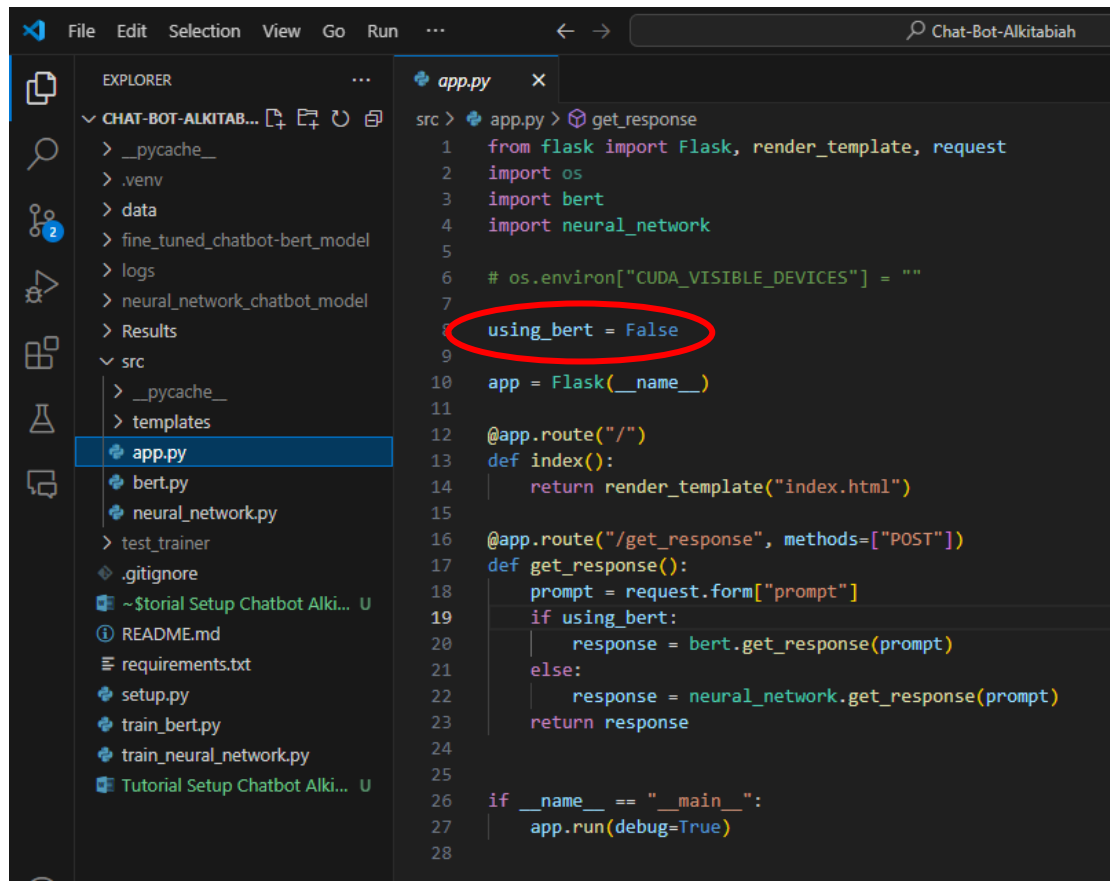
2. Untuk mengatur model yang akan digunakan (BERT atau Neural Network) ubah nilai pada baris:

```
using_bert = False
```

Nilai True akan memberitahu program untuk menggunakan model BERT

(Bidirectional Encoder Representations from Transformers).

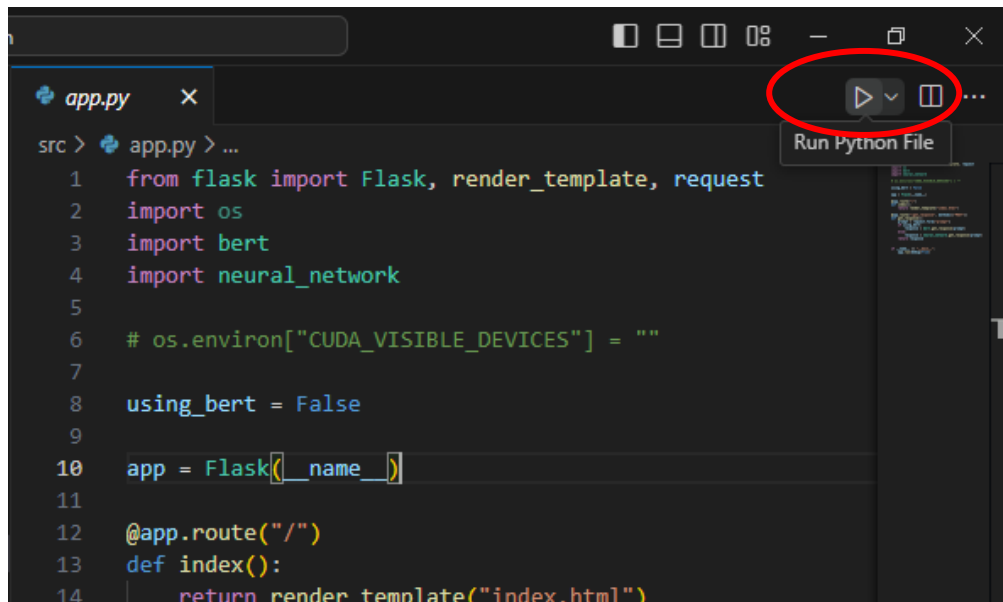
Sebaliknya, nilai False akan memberitahu program untuk menggunakan model Neural Network.



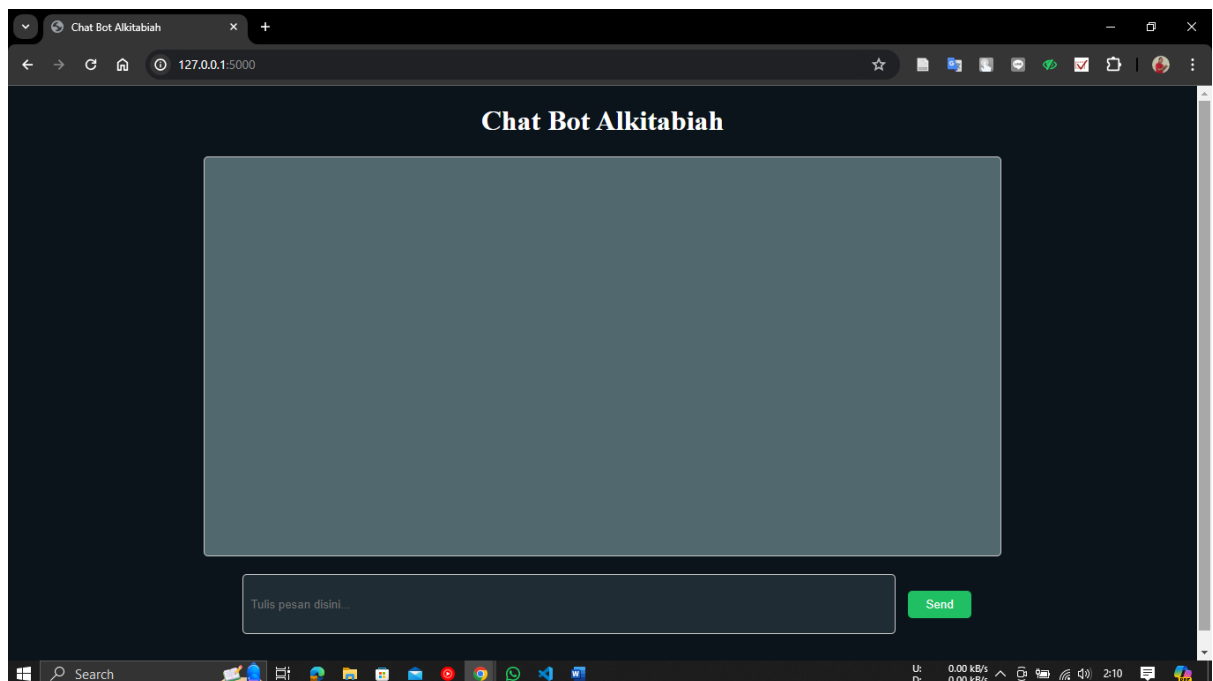
The screenshot shows a code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with folders like __pycache__, .venv, data, fine_tuned_chatbot-bert_model, logs, neural_network_chatbot_model, Results, and src. The src folder contains app.py, bert.py, and neural_network.py. The app.py file is selected and its content is displayed in the code editor. The code is a Flask application that uses either BERT or a Neural Network for text processing. The variable `using_bert` is set to `False`, which is highlighted with a red circle. The code also includes a `get_response` function that takes a prompt and returns a response based on the selected model.

```
1 from flask import Flask, render_template, request
2 import os
3 import bert
4 import neural_network
5
6 # os.environ["CUDA_VISIBLE_DEVICES"] = ""
7
8 using_bert = False
9
10 app = Flask(__name__)
11
12 @app.route("/")
13 def index():
14     return render_template("index.html")
15
16 @app.route("/get_response", methods=["POST"])
17 def get_response():
18     prompt = request.form["prompt"]
19     if using_bert:
20         response = bert.get_response(prompt)
21     else:
22         response = neural_network.get_response(prompt)
23     return response
24
25
26 if __name__ == "__main__":
27     app.run(debug=True)
28
```

3. Untuk menjalankan program "app.py" klik tombol play di pojok kanan atas pada aplikasi VS Code.



4. Tampilan UI dapat dilihat dengan membuka browser dan memasukkan alamat: <http://127.0.0.1:5000>



5. Selesai.