## Python pipenv

```
If this Python Tutorial saves you
hours of work, please whitelist it in
      your ad blocker 📵 and
            Donate Now
(https://www.pythontutorial.net/donation/)
  to help us \heartsuit pay for the web
 hosting fee and CDN to keep the
```

website running. **Summary**: in this tutorial, you'll learn how to configure a project with a new virtual environment using the Python pipenv tool.

## Creating a new project

First, create a new project folder e.g., crawler .

Second, navigate to the crawler folder and install the requests package using the pipenv command:

```
pipenv install requests
```

## Output:

```
Creating a Pipfile for this project...
Installing requests...
Adding requests to Pipfile's [packages]...
Installation Succeeded
Pipfile.lock not found, creating...
```

And you'll see that pipenv created two new files called Pipfile and Pipfile.lock . On top of that, it installed a virtual environment.

If you look at the project folder, you won't see the virtual environment folder.

To find the location of the virtual environment, you use the following command:

```
pipenv --venv
```

It'll return something like this on Windows:

```
C:\Users\<username>\.virtualenvs\crawler-7nwusESR
```

Note that the <username> is the username that you use to login to the Windows.

Third, create a new file called app.py in the project folder and add the following code to the file:

```
import requests

response = requests.get('https://www.python.org/')
print(response.status_code)
```

In this code, we imported the requests third-party module, use the get() function to make an HTTP request to the URL <a href="https://www.python.org/">https://www.python.org/</a> and display the status code ( 200 ).

Fourth, run the app.py file from the terminal by using the python command:

```
python app.py
```

It'll show the following error:

```
ModuleNotFoundError: No module named 'requests'
```

The reason is that Python couldn't locate the new virtual environment. To fix this, you need to activate the virtual environment.

Fifth, use the following command to activate the new virtual environment:

```
pipenv shell
```

If you run the app.py now, it should work correctly:

```
python app.py
```

Output:

200

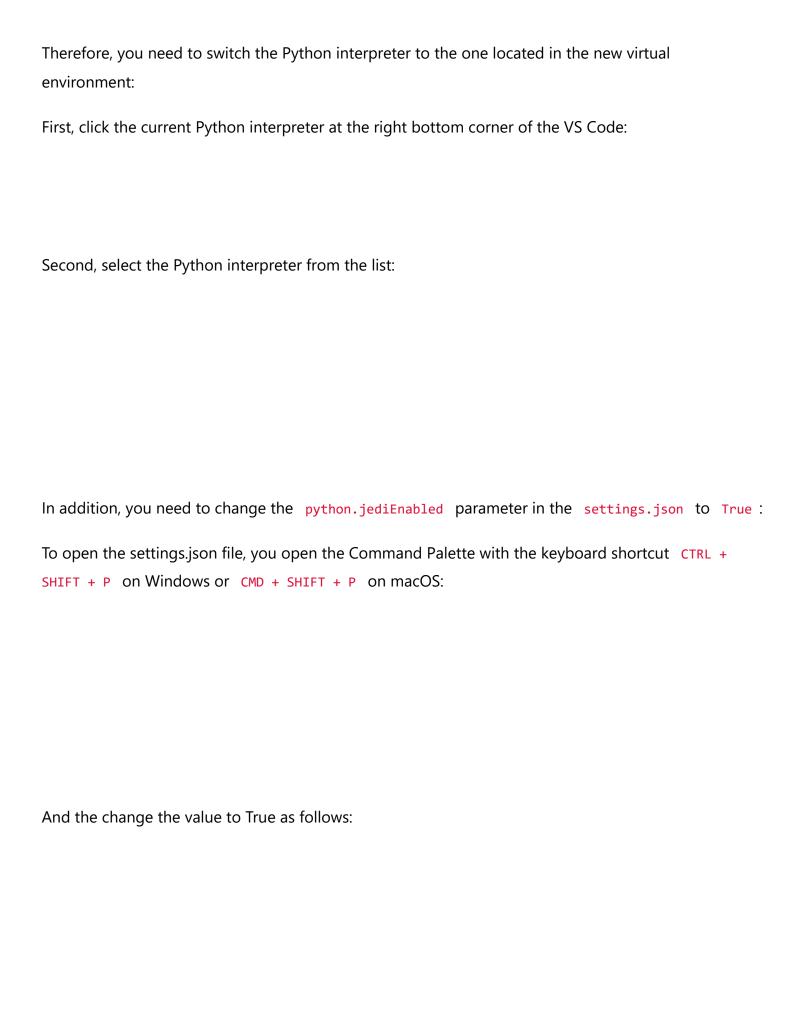
The status code 200 means the HTTP request has been succeeded.

Sixth, use the exit command to deactivate the virtual environment:

exit

## Resolving the Unresolved Import Warning in VS Code

If you're using VS Code, you may receive the unresolved import warning. The reason is that the VS code doesn't know which Python interpreter to use.



After that, you should	save the file and re	estart the VS Code	for the change to t	ake effect.