WIJDAN SKILLS

ENTERPRISE

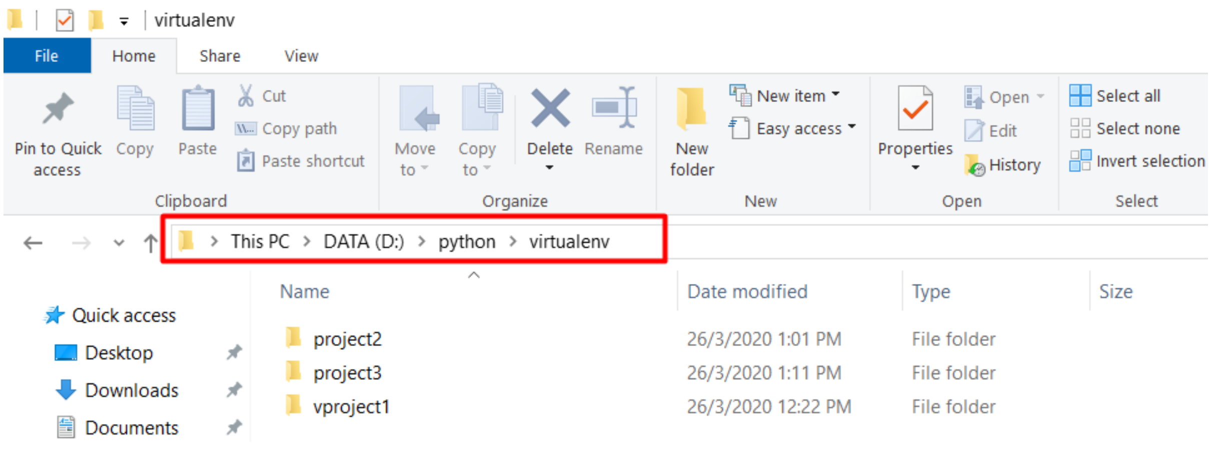
LAB 1 : Getting Started With Web Development

LAB 1: GETTING STARTED WITH WEB DEVELOPMENT

Objective:
In this lab we are going to create a virtual environment. So we will install the flask package inside this virtual environment. As we do not install into our main python system, so every project is not going to be affected with the packages updates etc.

Steps:

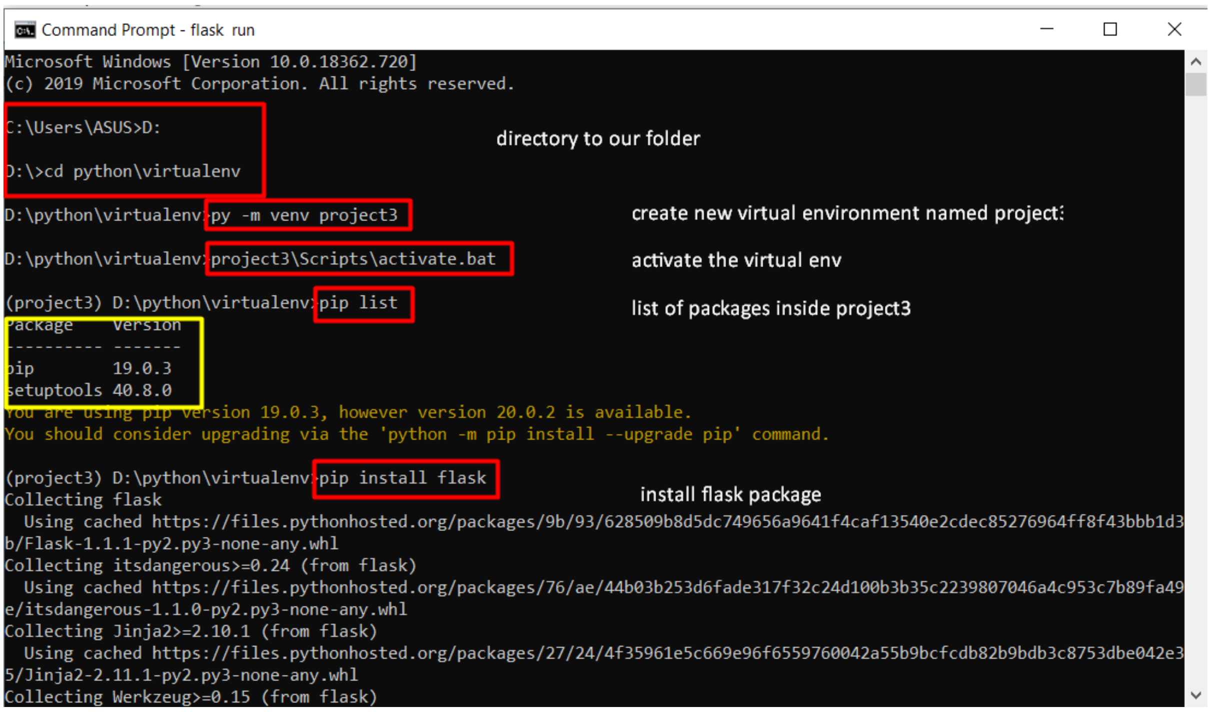
1. Create a new folder to save all our virtual project.



This PC > DATA (D:) > python > virtualenv

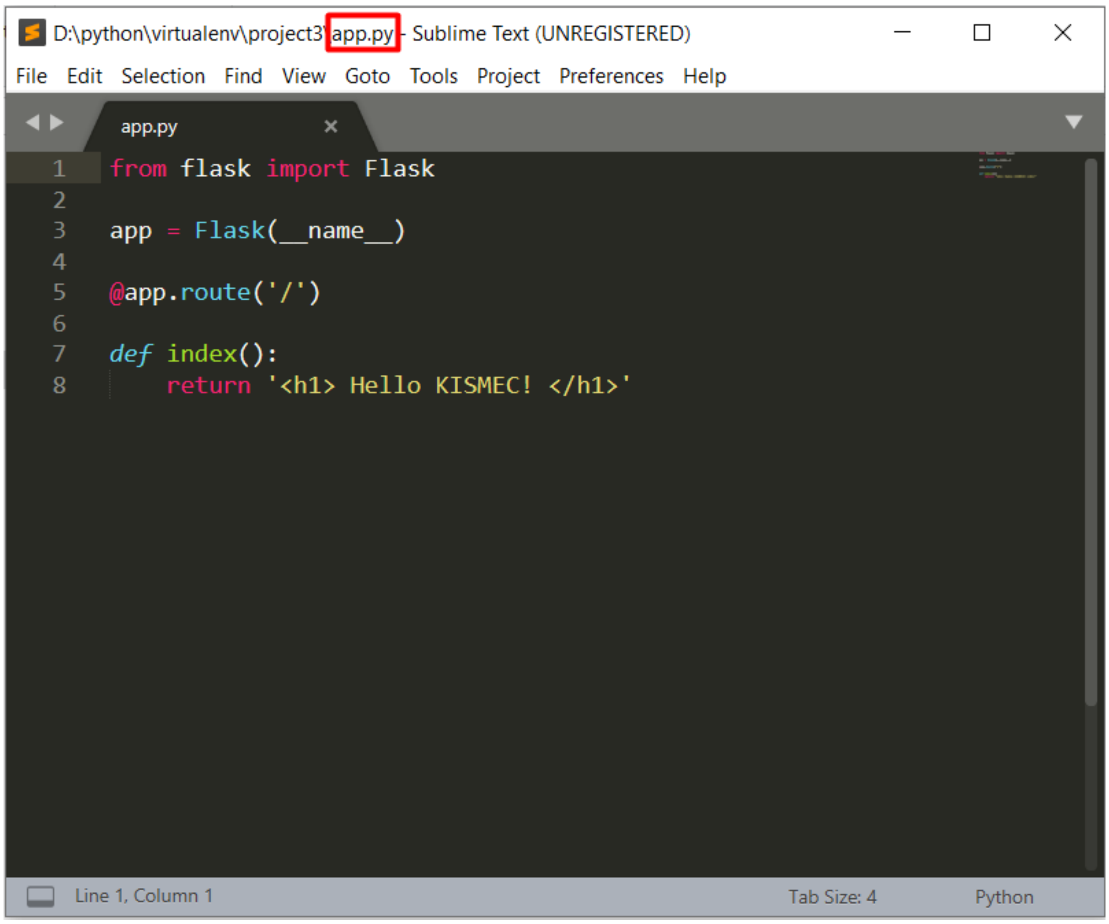
Name	Date modified	Type	Size
project2	26/3/2020 1:01 PM	File folder	
project3	26/3/2020 1:11 PM	File folder	
vproject1	26/3/2020 12:22 PM	File folder	

2. Create virtual environment, activate it and install flask inside


Command Prompt - flask run
directory to our folder
create new virtual environment named project3
activate the virtual env
list of packages inside project3
install flask package

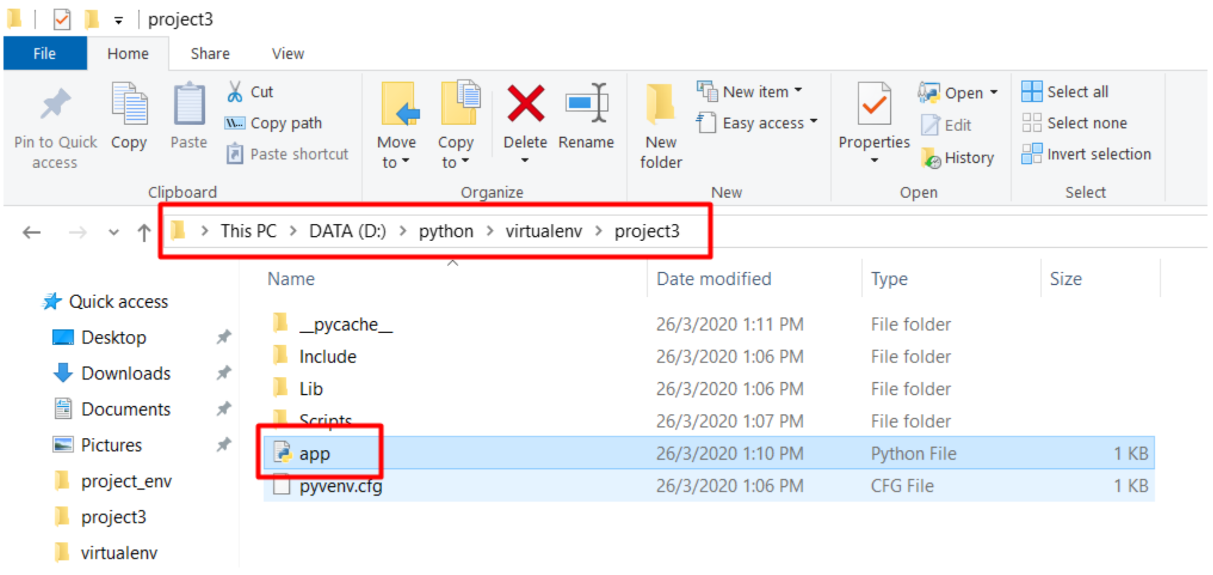
```
Microsoft Windows [Version 10.0.18362.720]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\ASUS>D:

D:\>cd python\virtualenv

D:\python\virtualenv>py -m venv project3

D:\python\virtualenv>project3\Scripts\activate.bat

(project3) D:\python\virtualenv>pip list
Package    Version
---------- -------
pip        19.0.3
setuptools 40.8.0
You are using pip version 19.0.3, however version 20.0.2 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

(project3) D:\python\virtualenv>pip install flask
Collecting flask
  Using cached https://files.pythonhosted.org/packages/9b/93/628509b8d5dc749656a9641f4caf13540e2cdec85276964ff8f43bbb1d3
b/Flask-1.1.1-py2.py3-none-any.whl
Collecting itsdangerous>=0.24 (from flask)
  Using cached https://files.pythonhosted.org/packages/76/ae/44b03b253d6fade317f32c24d100b3b35c2239807046a4c953c7b89fa49
e/itsdangerous-1.1.0-py2.py3-none-any.whl
Collecting Jinja2>=2.10.1 (from flask)
  Using cached https://files.pythonhosted.org/packages/27/24/4f35961e5c669e96f6559760042a55b9bcfcdb82b9bdb3c8753dbe042e3
5/Jinja2-2.11.1-py2.py3-none-any.whl
Collecting Werkzeug>=0.15 (from flask)
```

3. Code our first web app and saves as app.py, type all files inside project3 folder.


D:\python\virtualenv\project3\app.py - Sublime Text (UNREGISTERED)

```python
from flask import Flask

app = Flask(__name__)

@app.route('/')

def index():
    return '<h1> Hello KISMEC! </h1>'
```

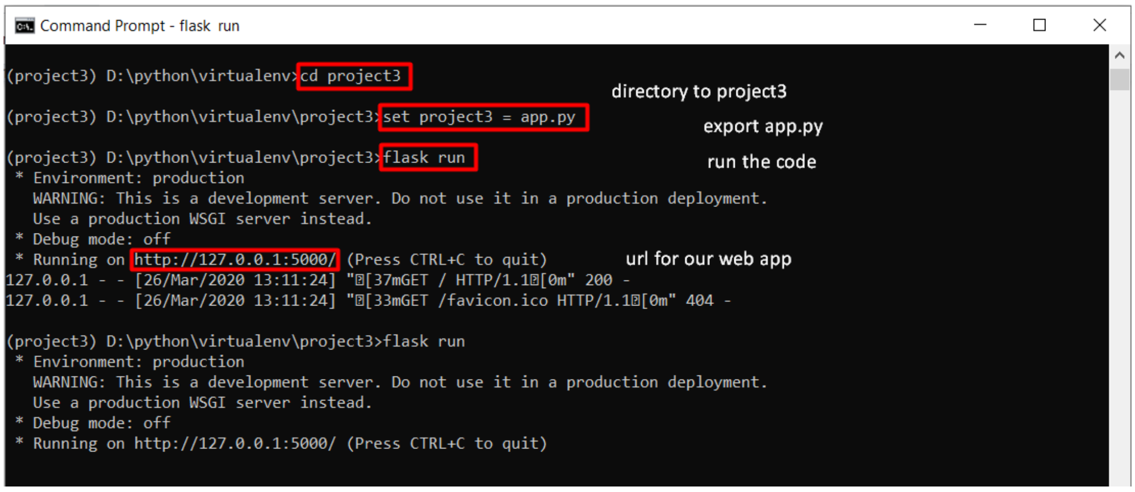
Line 1, Column 1   Tab Size: 4   Python



This PC > DATA (D:) > python > virtualenv > project3

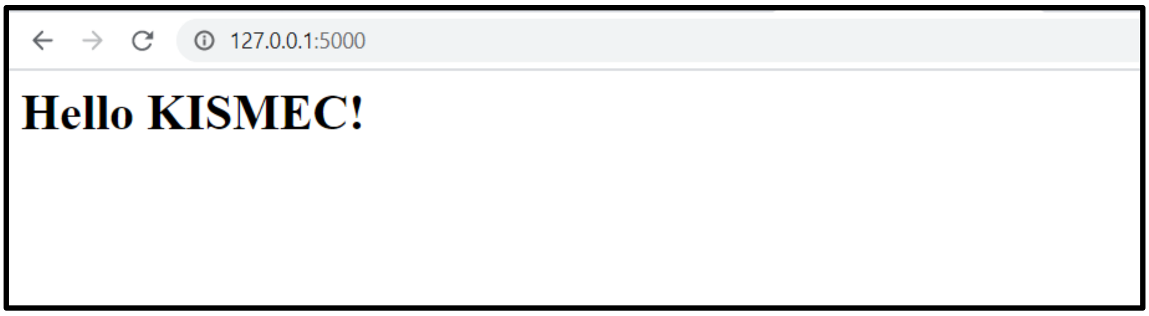
Name	Date modified	Type	Size
__pycache__	26/3/2020 1:11 PM	File folder	
Include	26/3/2020 1:06 PM	File folder	
Lib	26/3/2020 1:06 PM	File folder	
Scripts	26/3/2020 1:07 PM	File folder	
app	26/3/2020 1:10 PM	Python File	1 KB
pyvenv.cfg	26/3/2020 1:06 PM	CFG File	1 KB

4. Export and run
   * change set project3 = app.py to set FLASK_APP = app.py



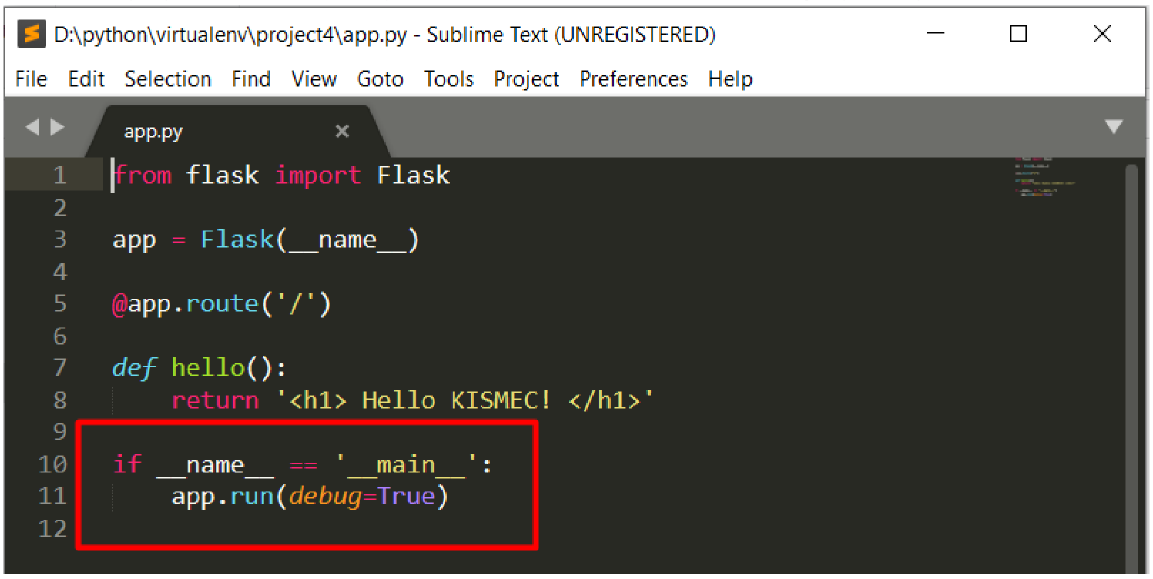
```
(project3) D:\python\virtualenv>cd project3

(project3) D:\python\virtualenv\project3>set project3 = app.py

(project3) D:\python\virtualenv\project3>flask run
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [26/Mar/2020 13:11:24] "␛[37mGET / HTTP/1.1␛[0m" 200 -
127.0.0.1 - - [26/Mar/2020 13:11:24] "␛[33mGET /favicon.ico HTTP/1.1␛[0m" 404 -

(project3) D:\python\virtualenv\project3>flask run
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

5. Go to http://127.0.0.1:5000/ in your browser



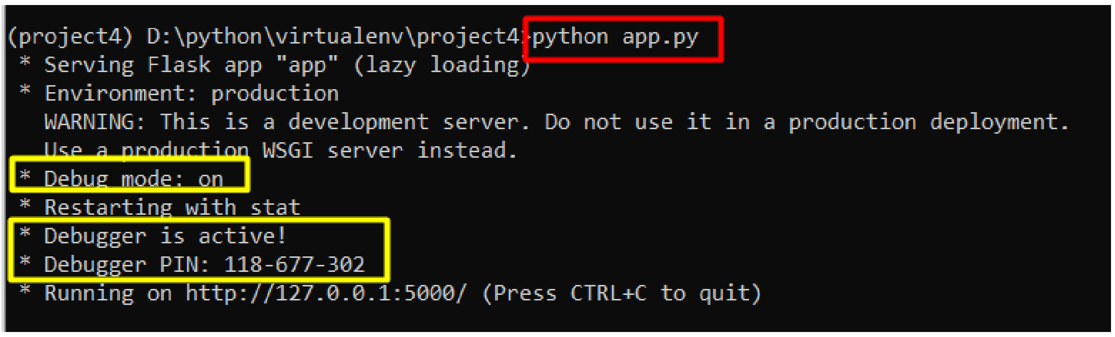
Hello KISMEC!

6. To run in debug mode, append this code. In debug mode you don't need to stop if you change your code. Save your new codes and refresh the page only.


D:\python\virtualenv\project4\app.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
app.py

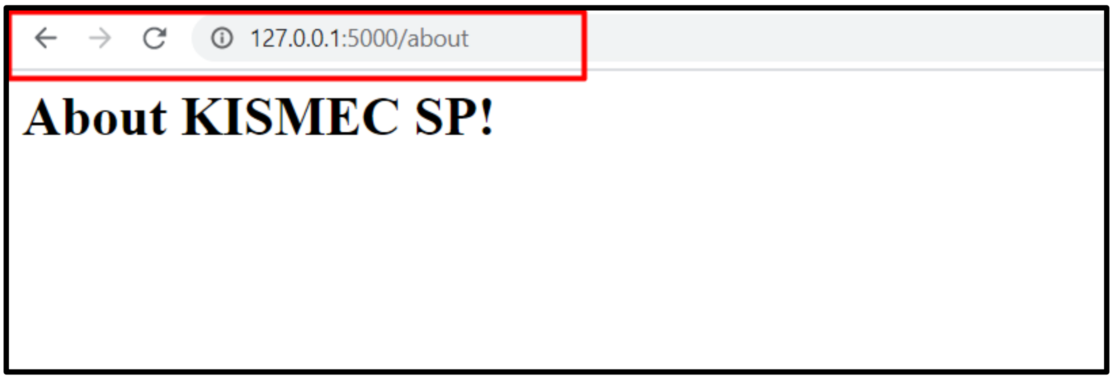
```python
from flask import Flask

app = Flask(__name__)

@app.route('/')

def hello():
    return '<h1> Hello KISMEC! </h1>'

if __name__ == '__main__':
    app.run(debug=True)
```

7. Run your code - python app.py



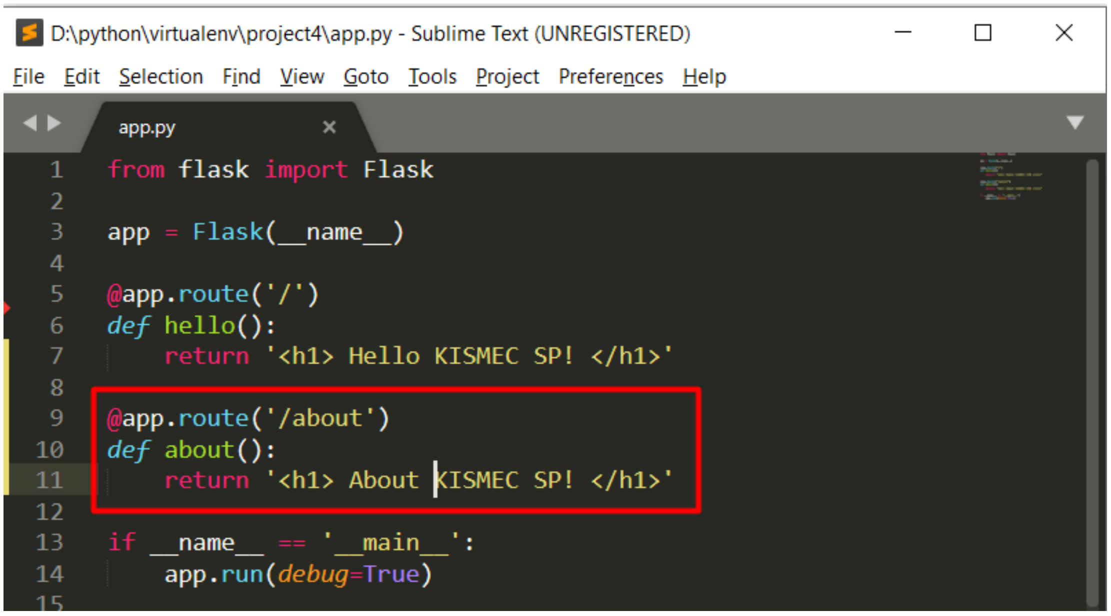
```
(project4) D:\python\virtualenv\project4>python app.py
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 118-677-302
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

8. We want to create page with a route



127.0.0.1:5000/about

About KISMEC SP!

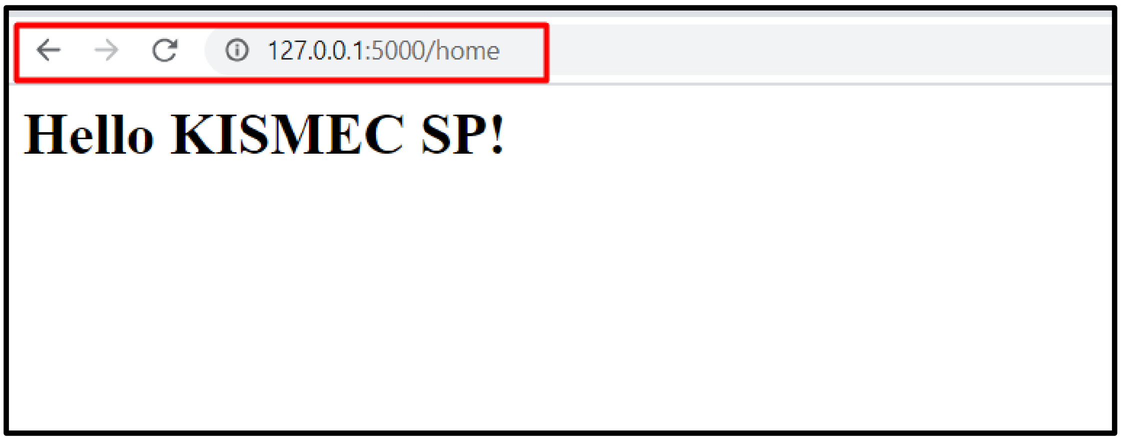


D:\python\virtualenv\project4\app.py - Sublime Text (UNREGISTERED)

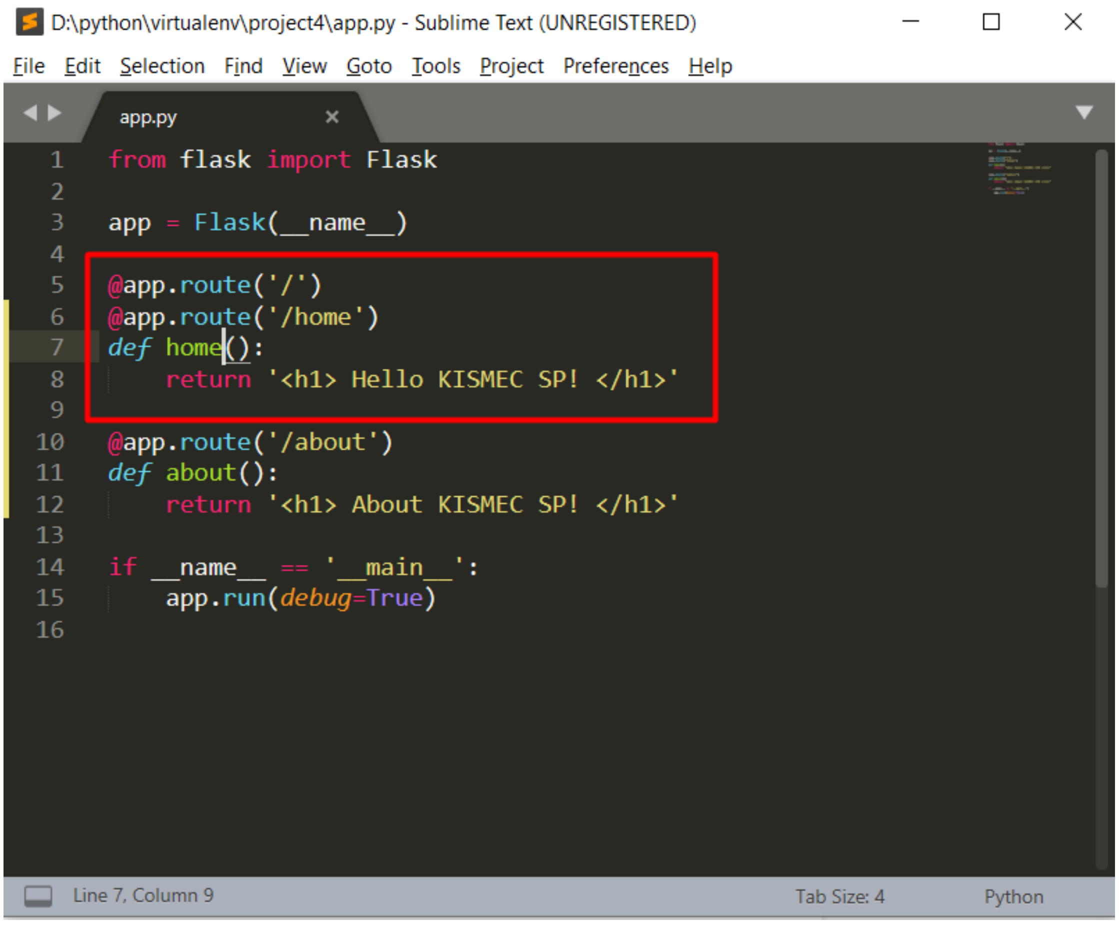
File Edit Selection Find View Goto Tools Project Preferences Help
app.py

```python
from flask import Flask

app = Flask(__name__)

@app.route('/')
def hello():
    return '<h1> Hello KISMEC SP! </h1>'

@app.route('/about')
def about():
    return '<h1> About KISMEC SP! </h1>'

if __name__ == '__main__':
    app.run(debug=True)
```

9. Route home + no route


127.0.0.1:5000/home

Hello KISMEC SP!


D:\python\virtualenv\project4\app.py - Sublime Text (UNREGISTERED)

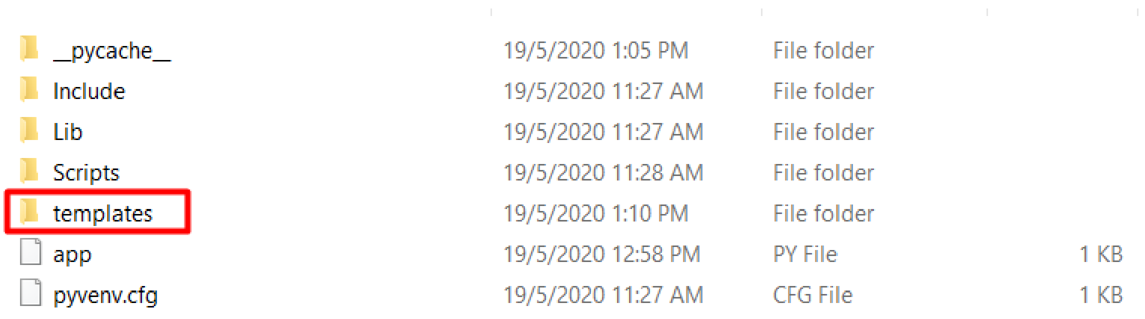
```python
from flask import Flask

app = Flask(__name__)

@app.route('/')
@app.route('/home')
def home():
    return '<h1> Hello KISMEC SP! </h1>'

@app.route('/about')
def about():
    return '<h1> About KISMEC SP! </h1>'

if __name__ == '__main__':
    app.run(debug=True)

```

Line 7, Column 9 Tab Size: 4 Python

File Edit Selection Find View Goto Tools Project Preferences Help
app.py

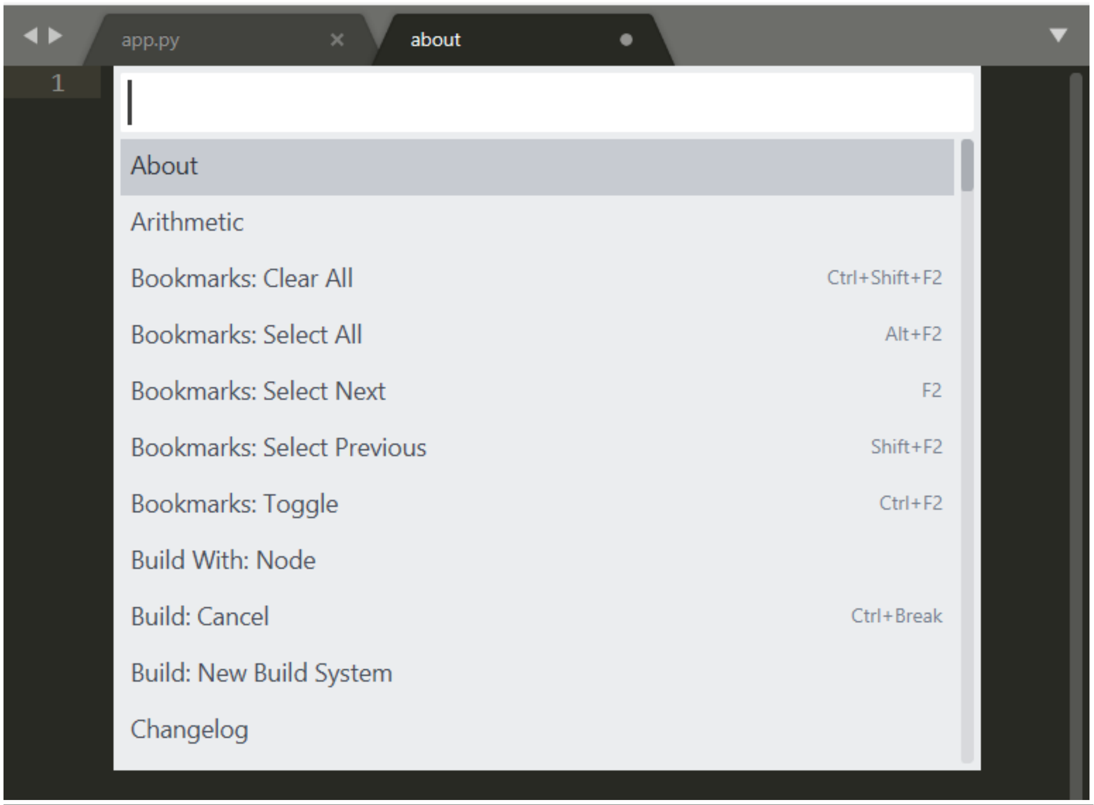
Template

1. Create new folder and name it “templates”

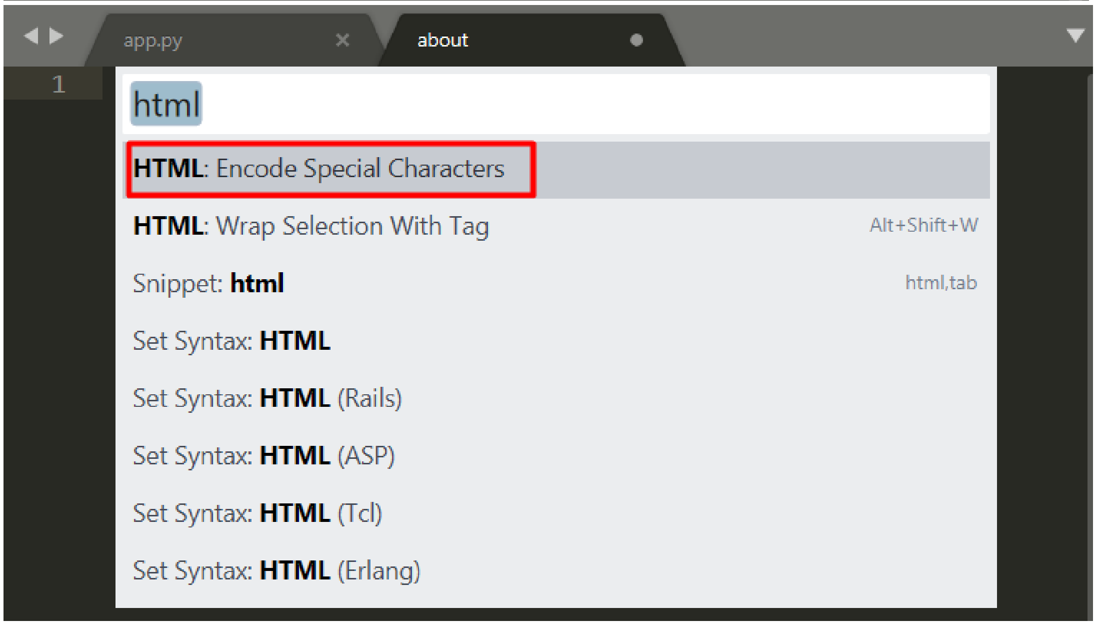


__pycache__	19/5/2020 1:05 PM	File folder	
Include	19/5/2020 11:27 AM	File folder	
Lib	19/5/2020 11:27 AM	File folder	
Scripts	19/5/2020 11:28 AM	File folder	
templates	19/5/2020 1:10 PM	File folder	
app	19/5/2020 12:58 PM	PY File	1 KB
pyvenv.cfg	19/5/2020 11:27 AM	CFG File	1 KB

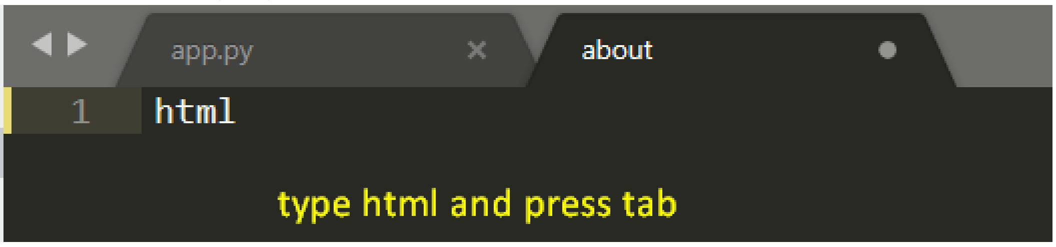
2. Create new html file and save in templates folder
   Ctrl-shift-p



app.py
about
About
Arithmetic
Bookmarks: Clear All Ctrl+Shift+F2
Bookmarks: Select All Alt+F2
Bookmarks: Select Next F2
Bookmarks: Select Previous Shift+F2
Bookmarks: Toggle Ctrl+F2
Build With: Node
Build: Cancel Ctrl+Break
Build: New Build System
Changelog

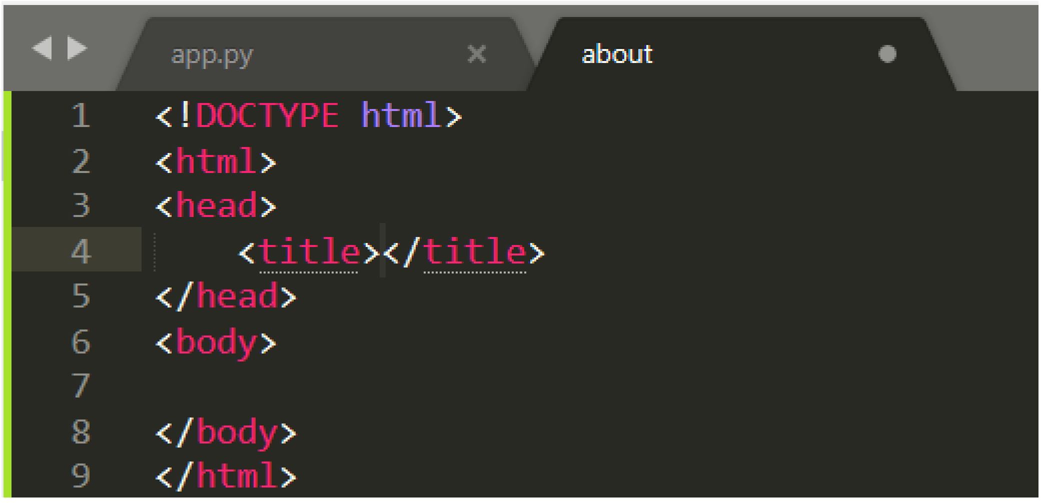


3. Code in html language



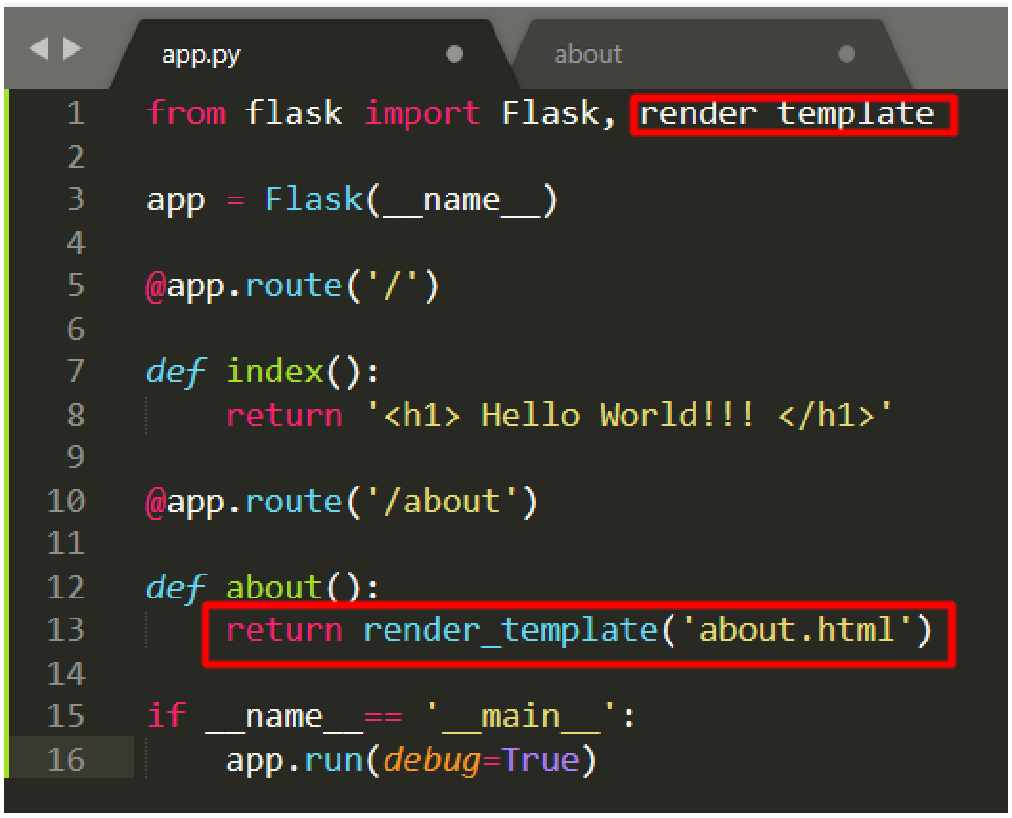
```
html
```

type html and press tab



```
<!DOCTYPE html>
<html>
<head>
	<title></title>
</head>
<body>

</body>
</html>
```

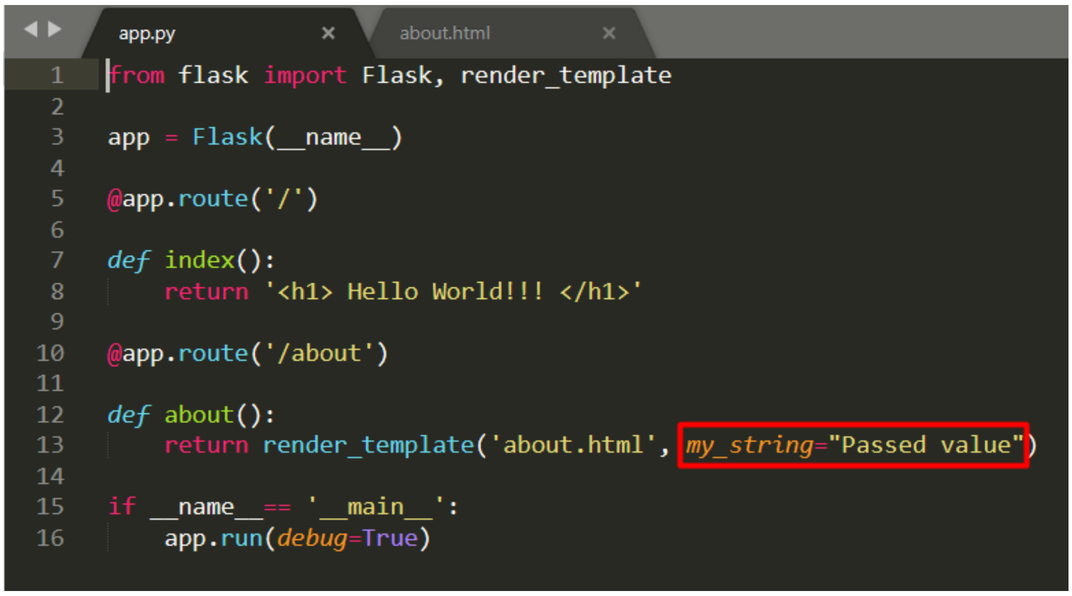
4. Import render_template



```python
from flask import Flask, render template

app = Flask(__name__)

@app.route('/')

def index():
    return '<h1> Hello World!!! </h1>'

@app.route('/about')

def about():
    return render_template('about.html')

if __name__== '__main__':
    app.run(debug=True)
```

Pass in keyword arguments to the template, like in the example with my_string

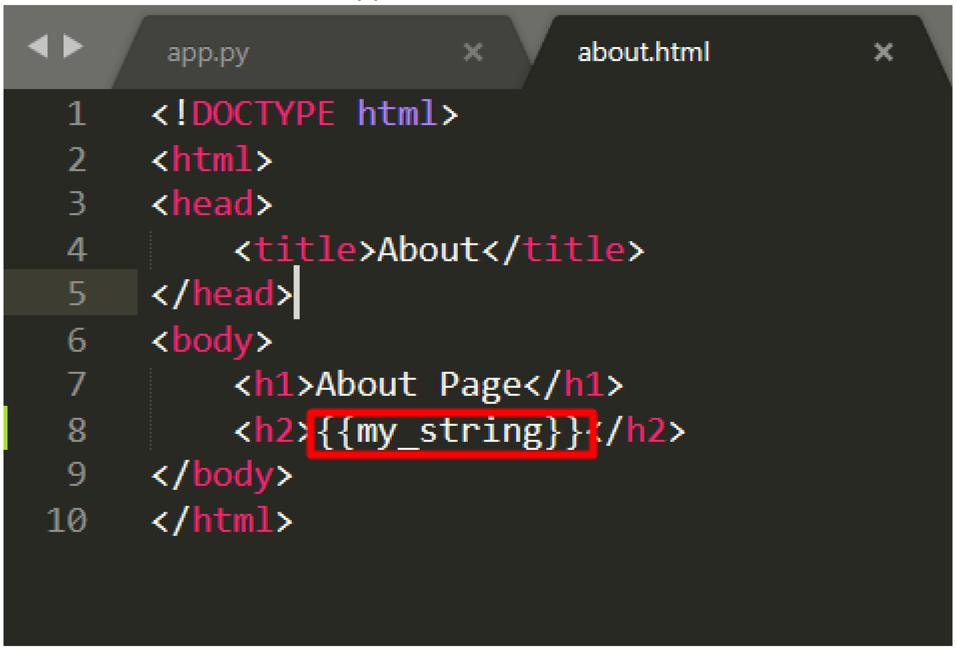
1. Render value to the template



app.py about.html

```
from flask import Flask, render_template

app = Flask(__name__)

@app.route('/')

def index():
    return '<h1> Hello World!!! </h1>'

@app.route('/about')

def about():
    return render_template('about.html', my_string="Passed value")

if __name__ == '__main__':
    app.run(debug=True)
```

2. Take the value from flask app



app.py about.html

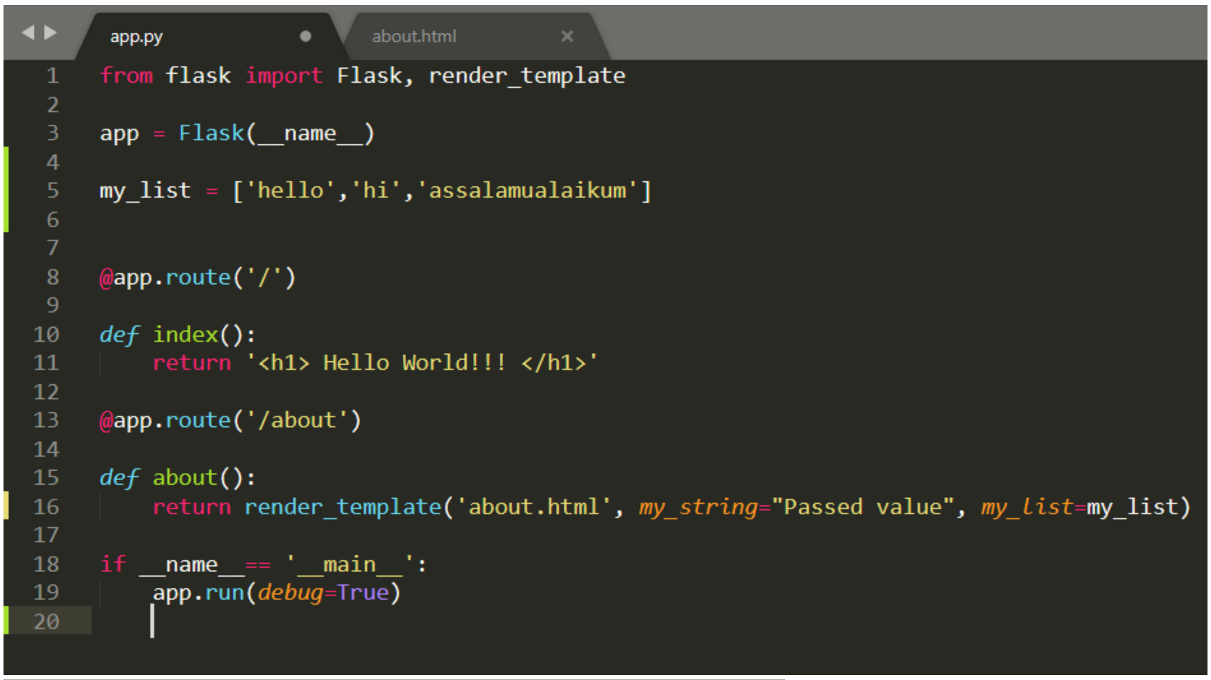
```
<!DOCTYPE html>
<html>
<head>
    <title>About</title>
</head>
<body>
    <h1>About Page</h1>
    <h2>{{my_string}}</h2>
</body>
</html>
```

Reference: https://realpython.com/primer-on-jinja-templating/, https://jinja.palletsprojects.com/en/2.11.x/templates/#template-inheritance

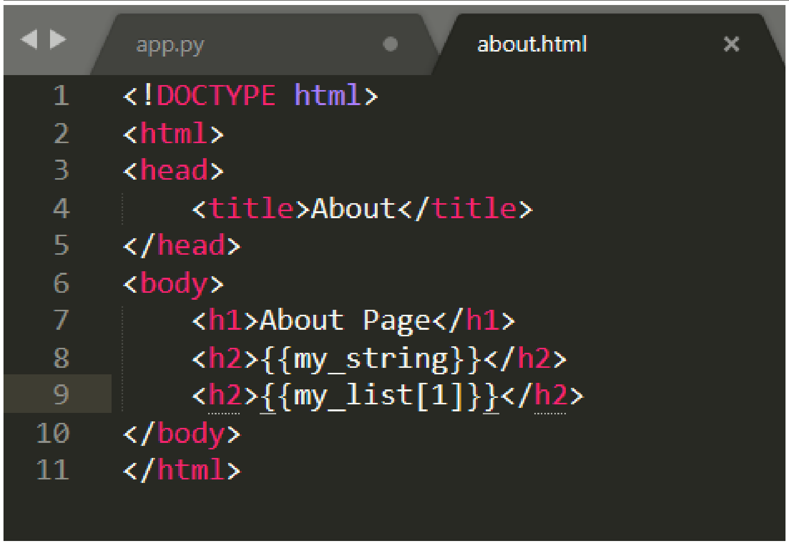
There are a few kinds of delimiters. The default Jinja delimiters are configured as follows:

- `{% ... %}` for Statements
- `{{ ... }}` for Expressions to print to the template output
- `{# ... #}` for Comments not included in the template output
- `#  ... ##` for Line Statements

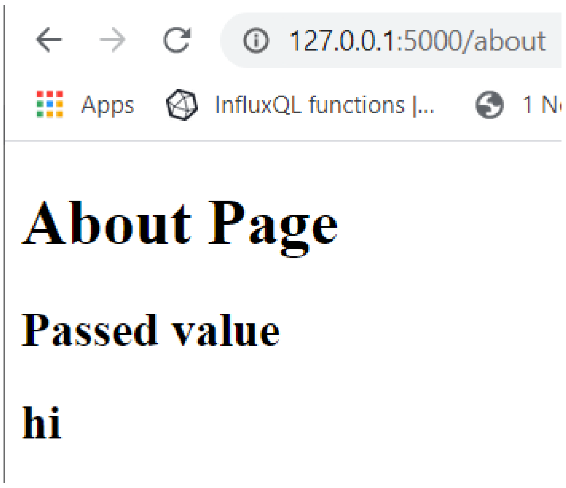
With list



```python
from flask import Flask, render_template

app = Flask(__name__)

my_list = ['hello','hi','assalamualaikum']


@app.route('/')

def index():
    return '<h1> Hello World!!! </h1>'

@app.route('/about')

def about():
    return render_template('about.html', my_string="Passed value", my_list=my_list)

if __name__ == '__main__':
    app.run(debug=True)

```



```html
<!DOCTYPE html>
<html>
<head>
    <title>About</title>
</head>
<body>
    <h1>About Page</h1>
    <h2>{{my_string}}</h2>
    <h2>{{my_list[1]}}</h2>
</body>
</html>
```

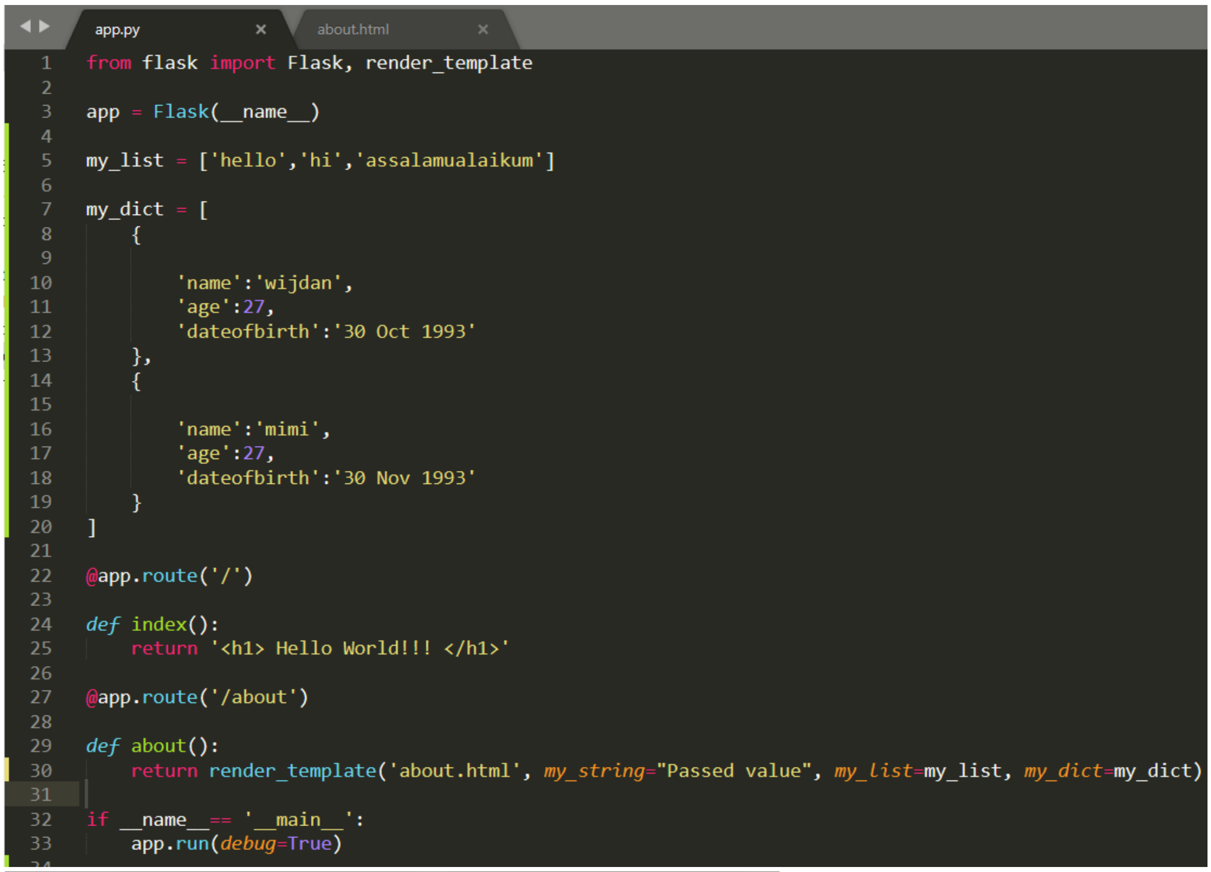


About Page

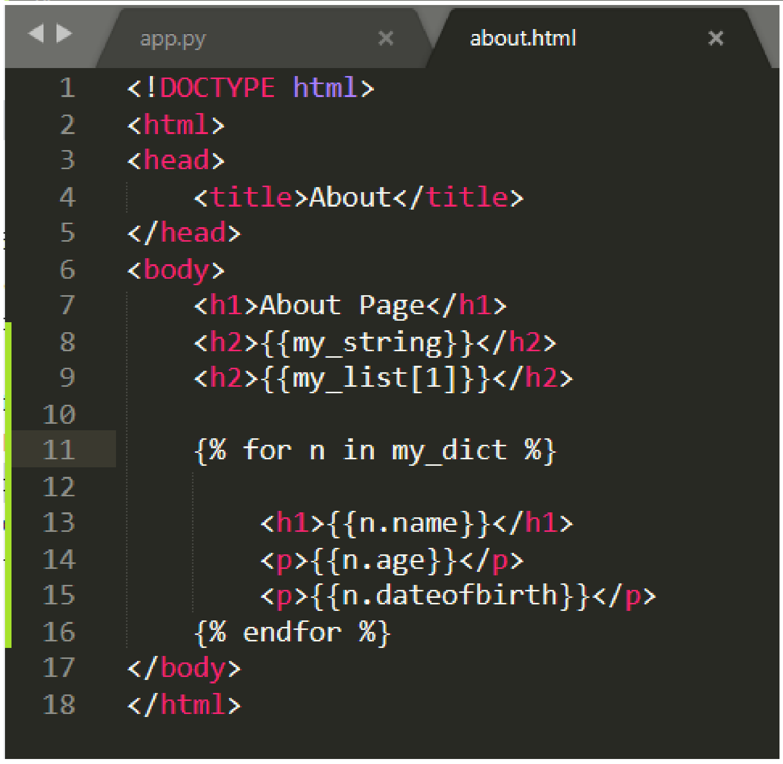
Passed value

hi

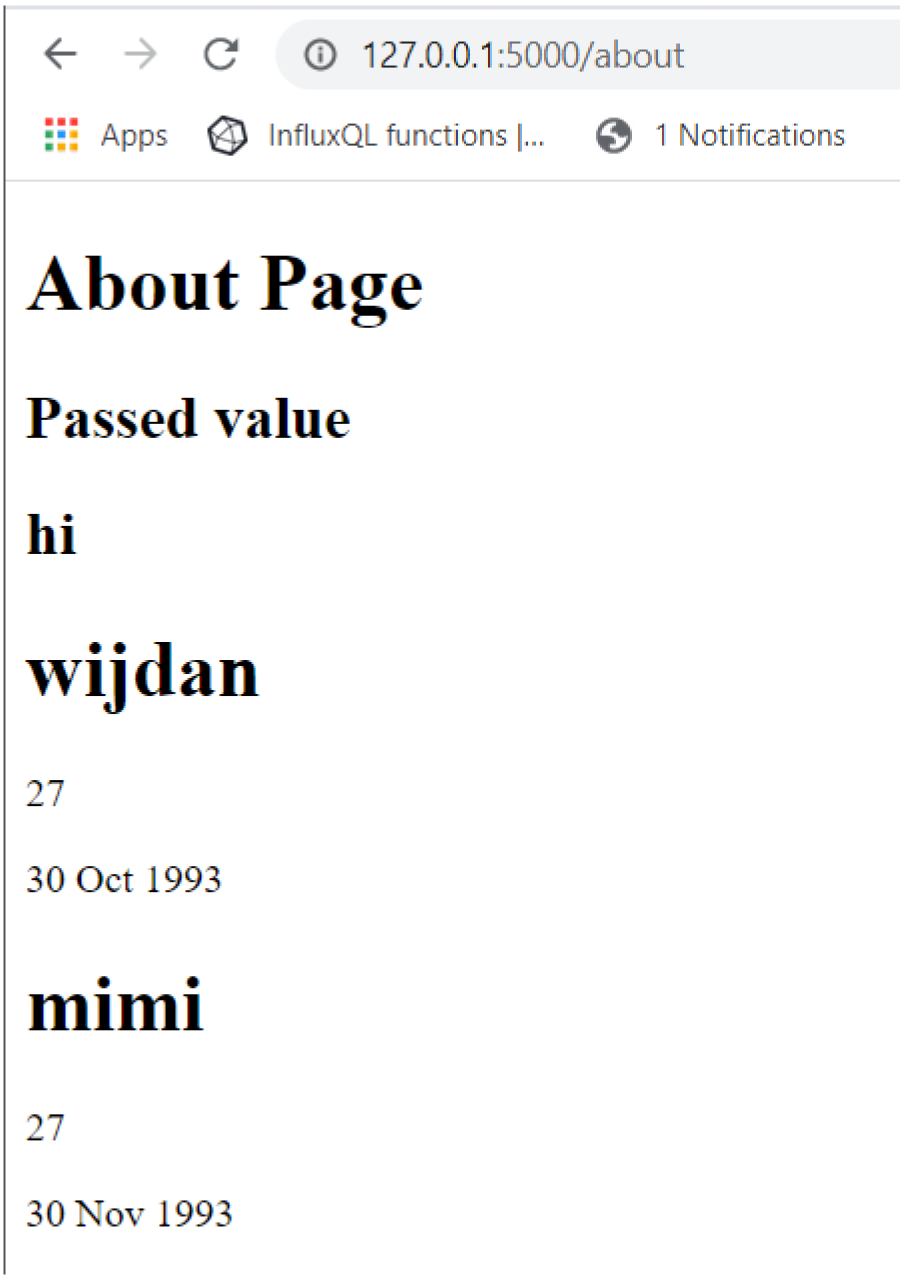
For loop



```python
from flask import Flask, render_template

app = Flask(__name__)

my_list = ['hello','hi','assalamualaikum']

my_dict = [
    {

        'name':'wijdan',
        'age':27,
        'dateofbirth':'30 Oct 1993'
    },
    {

        'name':'mimi',
        'age':27,
        'dateofbirth':'30 Nov 1993'
    }
]

@app.route('/')

def index():
    return '<h1> Hello World!!! </h1>'

@app.route('/about')

def about():
    return render_template('about.html', my_string="Passed value", my_list=my_list, my_dict=my_dict)

if __name__== '__main__':
    app.run(debug=True)
```



```html
<!DOCTYPE html>
<html>
<head>
    <title>About</title>
</head>
<body>
    <h1>About Page</h1>
    <h2>{{my_string}}</h2>
    <h2>{{my_list[1]}}</h2>

    {% for n in my_dict %}

        <h1>{{n.name}}</h1>
        <p>{{n.age}}</p>
        <p>{{n.dateofbirth}}</p>
    {% endfor %}
</body>
</html>
```



```
About Page

Passed value

hi

wijdan

27

30 Oct 1993

mimi

27

30 Nov 1993
```

References:

LAB 1: GETTING STARTED WITH WEB DEVELOPMENT

1. https://blog.miguelgrinberg.com/post/designing-a-restful-api-with-python-and-flask