

# Flask로 restful API 만들기 (1)

## 기본적인 flask 구조를 만들어보자!

### 1. VSCode Python 환경설정



#### Looking for a specific release?

Python releases by version number:

Release version	Release date	Click for more	
<a href="#">Python 3.7.7</a>	March 10, 2020	<a href="#">Download</a>	<a href="#">Release Notes</a>
<a href="#">Python 3.8.2</a>	Feb. 24, 2020	<a href="#">Download</a>	<a href="#">Release Notes</a>
<a href="#">Python 3.8.1</a>	Dec. 18, 2019	<a href="#">Download</a>	<a href="#">Release Notes</a>
<a href="#">Python 3.7.6</a>	Dec. 18, 2019	<a href="#">Download</a>	<a href="#">Release Notes</a>
<a href="#">Python 3.6.10</a>	Dec. 18, 2019	<a href="#">Download</a>	<a href="#">Release Notes</a>
<a href="#">Python 3.5.9</a>	Nov. 2, 2019	<a href="#">Download</a>	<a href="#">Release Notes</a>
<a href="#">Python 3.5.8</a>	Oct. 29, 2019	<a href="#">Download</a>	<a href="#">Release Notes</a>
<a href="#">Python 2.7.17</a>	Oct. 19, 2019	<a href="#">Download</a>	<a href="#">Release Notes</a>

[Windows x86-64 embeddable zip file](#)

Windows

for  
AMD64/EM64T/x64

[Windows x86-64 executable installer](#)

Windows

for  
AMD64/EM64T/x64

[Windows x86-64 web-based installer](#)

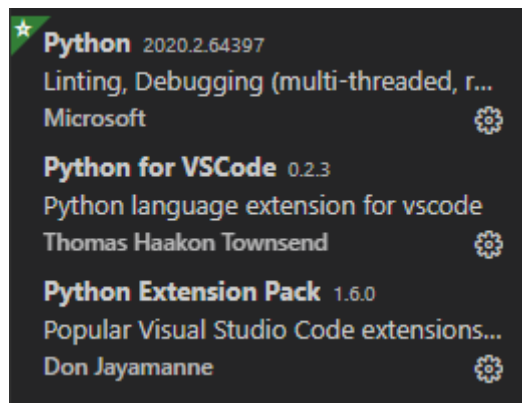
Windows

for  
AMD64/EM64T/x64

Python을 먼저 설치해준다.

필자는 가장 최신버전을 받았다. (작성일 기준 3.7.7)

3.7.7 버전을 클릭 후 자신의 운영체제에 맞는 버전을 설치한다.  
tensorflow는 64bit에서만 지원한다고한다.



VSCode에서도 Python 사용을 위해 설치가 필요하다.  
이와 같이 VSCode상에서 Python디버그 등을 위해 Extension을 install하였다.

## 2. VSCode상에서 기본적인 flask 코딩하기

```
OUTPUT  TERMINAL  DEBUG CONSOLE  PROBLEMS  1: powershell v + [ ] [X] ^ X

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

새로운 크로스 플랫폼 PowerShell 사용 https://aka.ms/pscore6

PS C:\workspace\TIL\PythonApiTest> pip install flask
```

처음으로 할 것은 flask 사용을 위해 flask를 다운로드 하는 것이다.  
cmd 또는 VSCode의 terminal을 통해 다운로드 하도록 한다.

```
from flask import Flask, jsonify, escape, request, render_template

app = Flask(__name__)

# app.route 를 사용해 매핑해준다.
# render_template -> 사용해 templates의 html 파일을 불러오겠다는 뜻

@app.route('/main')
def hello():
    name = request.args.get("name", "World")
    # return f'Hello, {escape(name)}!'
    return render_template('main.html')

@app.route('/testapi')
def test():
    return "안녕하세요?";

@app.route('/testjson')
def json():
    # flask 에서 기본적으로 제공하는 jsonify함수를 통해 값을 json형태로 전환할 수 있다.
    return jsonify(name='JKS')

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

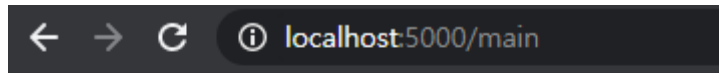
필자가 작성한 기본적인 flask 코드이다.

@app.route('값')을 통해 url에 대한 html파일 등 반환값을 매핑해준다.

```
* Serving Flask app "test.py"
* Environment: development
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

실행하게 되면 이와 같은 내용이 TERMINAL에 보여진다.

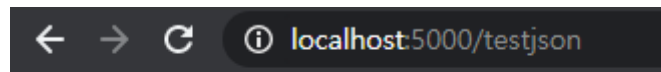
보여진 URL을 복사해보라는 얘기이다.



## 메인화면!!

앞서 작성한 flask 코드에 맞게 /main에 해당하는 url을 호출했다.

값이 정상적으로 나옵니다.



```
{"name": "JKS"}
```

이번에는 /testjson 을 호출해보도록한다.

jsonify를 통해 json형태로 변환된 값이 호출된다!!

이를 통해 spring boot환경에서 분석 결과값을 변환한 json형태의 값을 호출할 예정이다.

## Made By 장경석

---