

# 파이썬 플라스크 활용

- Flask는 경량화된 웹 애플리케이션 프레임워크
- 웹 개발 언어 - PHP, ASP, .NET, JSP/JAVA(Spring, Struts...)
- 파이썬 웹 개발 - Django, Flask, Fast API.....
- 파이썬으로 작성되어 있으며, 복잡하지 않으면서도 강력한 웹 애플리케이션을 개발할 수 있는 도구를 제공
- Flask는 Werkzeug WSGI 툴킷과 Jinja2 템플릿 엔진을 기반으로 제작

## 플라스크 공식 문서

- 새로운 지식을 습득할 때는 공식 문서 참고하는 것을 추천
- 현재 가장 최신 버전은 3.1.x

Quickstart — Flask Documentation (3.1.x)

Warning: This is the development version. The latest stable version is Version 3.0.x.

<https://flask.palletsprojects.com/en/latest/quickstart/>

## 플라스크 서버 실행

- 플라스크 설치

```
pip install Flask
```

- hello.py

```
from flask import Flask app = Flask(__name__) @app.route("/") def hello_world(): return "<p><font color='red'>Hello, World! 테스트입니다.</font></p>" if __name__ == '__main__': app.run(debug=True)
```

- 플라스크 서버 실행
- Flask에 애플리케이션 위치를 `--app` 옵션으로 지정 필요

```
flask --app hello run # WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.2 # 경고: 이것은 개발 서버입니다. 프로덕션 배포에서는 사용하지 마세요. 대신 프로덕션 WSGI(위스키) 서버를 사용하십시오.
```

- 로컬 서버 접속

```
http://127.0.0.1:5000/
```

- 서버 중지

```
Ctrl + C
```

## 라우팅 기본

- 현대 웹 애플리케이션은 사용자에게 의미 있는 URL을 제공하여 사용자 경험을 향상시킴
- 사용자는 기억하고 직접 방문할 수 있는 의미 있는 URL이 있는 페이지를 선호하며, 이로 인해 해당 페이지를 다시 방문할 가능성이 높아짐
- 플라스크에서는 `route()` 데코레이터를 사용하여 특정 함수를 URL에 바인딩 가능

## Quickstart — Flask Documentation (3.1.x)

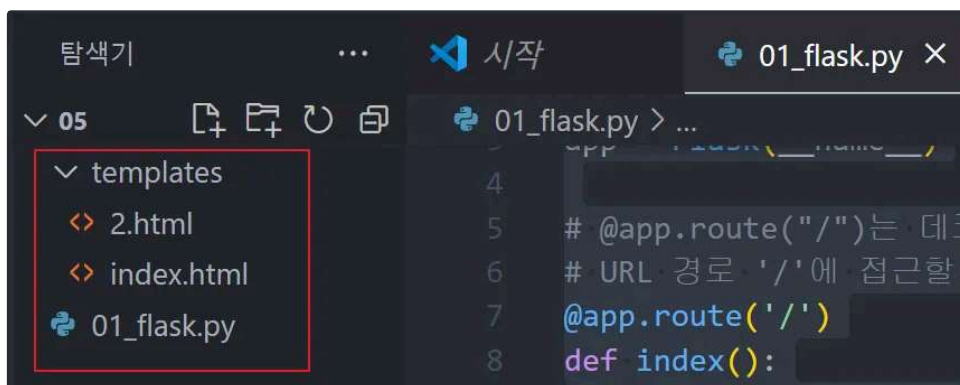
Warning: This is the development version. The latest stable version is Version 3.0.x.

<https://flask.palletsprojects.com/en/latest/quickstart/#variable-rules>

## route() 데코레이터의 역할

- 플라스크(Flask)에서 `route()` 데코레이터는 가장 핵심적인 기능 중 하나
- 이 데코레이터는 특정 함수를 하나 이상의 URL 경로에 연결하는 역할
- 사용자가 웹 애플리케이션의 특정 경로에 접근할 때, `route()` 에 의해 지정된 함수가 호출되어 사용자에게 콘텐츠를 반환

```
from flask import Flask app = Flask(__name__) # @app.route("/")는 데코레이터로, # URL 경로 '/'에 접근할 때 이어지는 함수를 호출하도록 지정
@app.route("/") def index(): return '<h1>Index Page</h1>'
@app.route("/hello") def hello(): return '<h1>Hello Page</h1>' if __name__ == '__main__': app.run(debug=True)
```



01\_flask.py

```

from flask import Flask, render_template app = Flask(__name__) #
@app.route("/")는 데코레이터로, # URL 경로 '/'에 접근할 때 이어지는 함수를 호
출하도록 지정 @app.route('/') def index(): return
render_template('index.html') @app.route('/2') def hello(): return
render_template('2.html') # 이 조건문은 스크립트가 직접 실행될 때만 아래 코드
를 실행도록 함 if __name__ == '__main__': app.run(debug=True)

```

## index.html 코드

```

<!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <meta
name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title> </head> <body> <h1>홈페이지 테스트</h1> </body>
</html>

```

## 2.html 코드

```

<!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <meta
name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title> </head> <body> <h1>두번째 페이지</h1> </body>
</html>

```