

# Quickstart

Eager to get started? This page gives a good introduction to Flask. Follow [Installation](#) to set up a project and install Flask first.

## A Minimal Application

A minimal Flask application looks something like this:

```
from flask import Flask

app = Flask(__name__)

@app.route("/")
def _hello_world():
    return "<p>Hello, World!</p>"
```

So what did that code do?

1. First we imported the **Flask** class. An instance of this class will be our WSGI application.
2. Next we create an instance of this class. The first argument is the name of the application's module or package. `__name__` is a convenient shortcut for this that is appropriate for most cases. This is needed so that Flask knows where to look for resources such as templates and static files.
3. We then use the **route()** decorator to tell Flask what URL should trigger our function.
4. The function returns the message we want to display in the user's browser. The default content type is HTML, so HTML in the string will be rendered by the browser.

Save it as **hello.py** or something similar. Make sure to not call your application **flask.py** because this would conflict with Flask itself.

To run the application, use the **flask** command or **python -m flask**. You need to tell the Flask where your application is with the **--app** option.

```
$ flask --app hello run
* Serving Flask app 'hello'
* Running on http://127.0.0.1:5000 (Press CTRL+C to quit)
```

---

## Application Discovery Behavior:

As a shortcut, if the file is named **app.py** or **wsgi.py**, you don't need to specify the **--app** option. See the [Command Line Interface](#) for more details.