

Starting a Flask Project from Scratch

Follow these steps to create a basic Flask project:

1. Install Python

Ensure Python is installed on your system. You can download it from python.org.

2. Set Up a Virtual Environment

```
# Create a virtual environment
python -m venv venv

# Activate the virtual environment
# On Windows:
venv\Scripts\activate
# On macOS/Linux:
source venv/bin/activate
```

3. Install Flask

```
pip install flask
```

4. Create the Project Structure

```
habit_tracker/
├── app/
│   ├── routes.py    ← all your Flask pages (views)
│   ├── models.py    ← database tables
│   ├── forms.py     ← forms for user input
│   └── templates/   ← HTML files go here
├── static/          ← CSS, images, JavaScript files
├── tests/           ← where you'll put test files (later)
├── README.md        ← info about your project
├── requirements.txt ← auto-generated list of packages
├── run.py           ← how to start the app
└── deliverables/    ← intermediate presentations
```

5. Write Your First Flask App

Create a file named `run.py` and add the following code:

```
from app import app

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

In the `app/` directory, create a file named `routes.py` and add the following code:

```
from flask import Flask

app = Flask(__name__)

@app.route('/')
def home():
    return "Hello, Flask!"
```

6. Run the Application

```
python run.py
```

Visit <http://127.0.0.1:5000> in your browser to see your app running.

7. Add More Features

- Use the `templates/` folder for HTML files.
- Use the `static/` folder for CSS, JavaScript, and images.
- Add database models in `models.py`.
- Add forms for user input in `forms.py`.
- Write tests in the `tests/` folder.
- Use `requirements.txt` to track dependencies with `pip freeze > requirements.txt`.

8. Learn More

Refer to the [Flask Documentation](#) for advanced topics.