In VS Code, your folder should look like this:

```
project_python_dockerfile/
├── hello.py
├── Dockerfile
├── docker-compose.yml
```

Open **VS Code → Open Folder → project_python_dockerfile**

Step 1: Create `first.py`

```
from flask import Flask

app = Flask(__name__)

@app.route("/")

def test_program():
    return """
    <p> Hello There </p>
    <p>Welcome to CMPT 353 Tutorials with
Pyhton!</p>
    """
```

Step 2: Create the Dockerfile

In VS Code:

- Right-click → **New File**
- Name it exactly: **Dockerfile**
  (⚠️ no `.txt`, no extension)

```
# Base image
FROM python:latest
# Set working directory
WORKDIR /app
# Copy files
COPY first.py /app
# Install dependencies
RUN pip install Flask
EXPOSE 5000
CMD ["/bin/bash"]
```

Step 3: Create docker-compose.yml

In VS Code:

- Right-click → **New File**
- Name it exactly: **docker-compose.yml**

```
version: "3.9"
services:
 python1:
  build: .
  container_name: p1
  command: flask run --host=0.0.0.0
  ports:
   - "5000:5000"
  volumes:
   - ./:/app
  environment:
   - FLASK_APP=first
   - FLASK_ENV=development
   - FLASK_DEBUG=1
   #- FLASK_APP=${FLASK_APP}
   #- FLASK_ENV=production
```

- `FLASK_APP=first`
  My Flask application is in first.py and contains app = Flask(...)
- `FLASK_ENV=development`
  I am a developer working on this app.
- `FLASK_ENV=production`
  This app is live and used by real users.
- `FLASK_DEBUG=1`
  explicitly turns on debug mode (by default if `FLASK_ENV=development`)
- `FLASK_APP=${FLASK_APP}`

IF you want to decide the Python file name outside the code / compose file,

```
export FLASK_APP=python_file_name
docker compose up
```

used when you deploy the same container to different environments

Step 4: Run (Compose way)

From VS Code terminal:

```
docker compose down
docker compose up
```

Step 5: Test in Browser

http://localhost:5000