

# Lab 01. Get Started with FastAPI

---

## Objectives:

- Setup development environment using Anaconda and VS Code
- Create first FastAPI project
- Understand basic routing

## A. Setup Development Environment

---

Following setup is only done once.

### Setup Python

1. Install Python using Anaconda.
  - Refer to <https://tinyurl.com/Setup-Python-with-Anaconda>
2. Visual Studio Code and get familiar with it.
  - Refer to <https://tinyurl.com/get-started-with-vs-code>
3. Start command prompt; Install FastAPI in current virtual environment.

```
pip install fastapi uvicorn
```

### Setup Visual Studio Code

1. Install following extensions in VS Code.
  - [Python \(by Microsoft\)](#)
  - [Pylance \(by Microsoft\)](#)
  - [REST Client \(Huachao Mao\)](#)
2. Create a project folder `FastProjects`, which will contains all labs.

## B. Hello World

---

1. Create a folder `lab01`, which will be the project folder of Lab 01.
2. Create a python file `main.py` inside the `lab01` folder.
3. Add following code to the `main.py` file.

```
from fastapi import FastAPI
app = FastAPI()

@app.get("/")
def get_root():
    return {'message': 'Hello world'}
```

- Import FastAPI, which is a Python class that provides all the functionality for your API.
  - Create a FastAPI "instance" and assign it to `app`.

- Create a route for path `/` with `GET` operation, where `@app.get("/")` is **path operation decorator**, and `def get_root()` is **path operation function**.

9. In the terminal, goes into the `Tab01` folder, and run following command.

```
uvicorn main:app --reload
```

- `main`: the file `main.py` (the Python "module").
  - `app`: the object created inside of `main.py` with the line `app = FastAPI()`.
  - `--reload`: make the server restart after code changes. Only use for development.
10. Above command serves your app in local machine at <http://127.0.0.1:8000>. Visit that URL in the browser.

## C. Exercises

---

### Part 1

1. Add another route for path `/about` with `GET` operation. It returns a JSON object `{message: "My First FastAPI Project"}` to the client.

### Part 2

1. Add a `GET /time` endpoint which returns current date time value in a JSON object `{datetime: "2020-09-17 22:17:30"}`

## Lab 01 - Suggested Answers

---

### Part 1

```
@app.get("/about")
def get_about():
    return {'message': 'My First FastAPI Project'}
```

### Part 2

```
@app.get('/now')
def get_time():
    from datetime import datetime
    now = datetime.now()
    return {'datetime': now.strftime('%Y-%m-%d %H:%M:%S')}
```