

LAB02 – Set up a basic Express server and Work with Git

Objective: Students will learn how to

- Set up an Express project using TypeScript
- Configure the TypeScript compiler (tsconfig.json)
- Run an Express server using ts-node
- Initialize Git and make their first commit
- Serve a simple route: GET /hello

Lab instruction

- The LAB01 instruction is posted on Mango CMU of the subject 953261.
- The assignment is scheduled and assigned to your account on Mango CMU.
 - **Check the calendar for the 'due date' and 'close date.'**
 - The submission later than the 'due date' will get 50% off your score.
 - At the 'close date', you cannot submit your assignment to the system.
- When you finish each question, capture the screen of your execution output for later upload to the MS Teams assignment.
- When you finish all the questions,
 1. Make sure you upload all capture files to Mango CMU assignment.
 2. Post your name and student ID as a comment under the Lab02 assignment to queue for TA to verify your work.
- This lab is worth for 10 points in total.

Prerequisites

Students should have:

- Node.js and npm installed
- VS Code installed
- Git installed and configured
- LAB01 completed

PART 1 — Create Express.js and TypeScript Project

1 Create a project folder

- In the terminal:

```
mkdir LAB02  
cd LAB02
```

Your terminal path must now be inside the LAB02 directory. (E.g, .../Desktop/LAB02 \$)

2 Initialize npm

- Run the following command in the terminal:

```
npm init -y
```

This creates a package.json file.

3 Install TypeScript + ts-node + nodemon

- Run the following command in the terminal:

```
npm install -D typescript ts-node nodemon @types/node
```

- After installation, you should see listed in the devDependencies section of your package.json file, like this:

```
"devDependencies": {  
  "@types/node": "^24.10.1",  
  "nodemon": "^3.1.11",  
  "ts-node": "^10.9.2",  
  "typescript": "^5.9.3"  
}
```

4 Install Express (with TypeScript types)

- Run the following command in the terminal:

```
npm install express  
npm install -D @types/express
```

- After installation, you should see listed in the devDependencies section of your package.json file, like this:

```
"dependencies": {  
  "express": "^5.1.0"  
},  
"devDependencies": {  
  "@types/express": "^5.0.5",  
  "@types/node": "^24.10.1",  
  "nodemon": "^3.1.11",  
  "ts-node": "^10.9.2",  
  "typescript": "^5.9.3"  
}
```

5 Initialize TypeScript configuration

- Run the following command in the terminal:

```
npx tsc --init
```

This creates tsconfig.json file.

- Recommended changes for Express
- Open tsconfig.json and ensure

```
{
  "compilerOptions": {
    "target": "es2020",
    "module": "commonjs",
    "rootDir": "./src",
    "outDir": "./dist",
    "esModuleInterop": true,
    "strict": true
  }
}
```

6 Create project structure

- Run the following command in the terminal:

```
mkdir src
```

Inside src, create index.ts:

```
LAB02/
├── package.json
├── tsconfig.json
├── node_modules/
├── src/
│   └── index.ts
```

PART 2 — Write a Express Server in TypeScript

- Open src/index.ts and add:

```
Import express, { Request, Response } from "express";
const app = express();
const port = 3000;

// Simple route
app.get("/hello", (req:Request, res:Response) => {
  res.send("Hello from Express + TypeScript! ");
});
app.listen(port, () => {
  console.log(`Server is running at http://localhost:${port}`);
});
```

PART 3 — Run the Server

- Add a script to package.json

Inside "scripts":

```
"scripts": {  
  "dev": "nodemon --exec ts-node src/index.ts"  
}
```

- Now run:

```
npm run dev
```

- In your browser, visit:

```
http://localhost:3000/hello
```

Here is an expected output:

Hello from Express + TypeScript!

PART 4 — Initialize Git Repository

1. Initialize Git

- Run the following command in the terminal:

```
git init
```

The .git is created in the folder to initialize Git.

2. Stage all files

- In VS Code Source control, click "+".

3. Commit

- Type the commit message:

```
Initial setup: Express + TypeScript project
```

PART 5 — Optional: Push to GitHub

- If logged in to GitHub in VS Code:
- Click Publish Branch → choose private repository.

PART 6 — Deliverables

Students must submit:

1. Screenshot of project folder structure
2. Screenshot of Express server running (npm run dev)
3. Screenshot of browser showing /hello
4. Screenshot of commit in VS Code
5. GitHub repo URL (if published)