

รหัสนักศึกษา : 61070143

Pre-Lab

1. Fork project จาก <https://github.com/saranonuan/sdte-lab5>
2. Clone project ที่ได้ fork ไป ลงมายังเครื่อง
3. รัน "npm install" เพื่อลง Dependencies ทั้งหมด

LAB A – Prettier บันทึกผลในหน้าที่ 2

4. ปรับ .prettierrc.json ให้มีข้อกำหนดดังนี้ และบันทึกไฟล์
 1. การ Tab ให้ใช้ 2 space
 2. ให้ลงท้ายด้วย Semi colon เพื่อจบ Statement
 3. ให้ใช้ Double Quote
5. รัน "npm run prettier --write ." เพื่อให้ prettier ปรับปรุง Source code ให้ตามมาตรฐานที่ตั้งค่า

LAB B – Jest บันทึกผลในหน้าที่ 3

6. รัน "npm run test" เพื่อรัน Test ดูว่าผ่าน Test หรือไม่
7. ปรับปรุง Code ให้ผ่าน Test โดยแก้ไขในไฟล์ "src/server.js" ให้ return "NO ONE" เป็น "รหัสนักศึกษา" เช่น "64000001"
8. รัน "npm run test" เพื่อทดสอบให้ผ่าน Test โดยต้องขึ้น Success เขียว Test ผ่านทั้งหมด

LAB C - Docker Build & Push to Registry บันทึกผลในหน้าที่ 4

9. เพิ่มไฟล์ที่ไม่ต้องการใน .dockerignore เพื่อให้เหลือแค่ไฟล์และโฟลเดอร์ต่อไปนี้

```
> docker run --rm --entrypoint "/bin/ls" sdte_lab_5:v1 -a
.
..
node_modules
package-lock.json
package.json
src
```

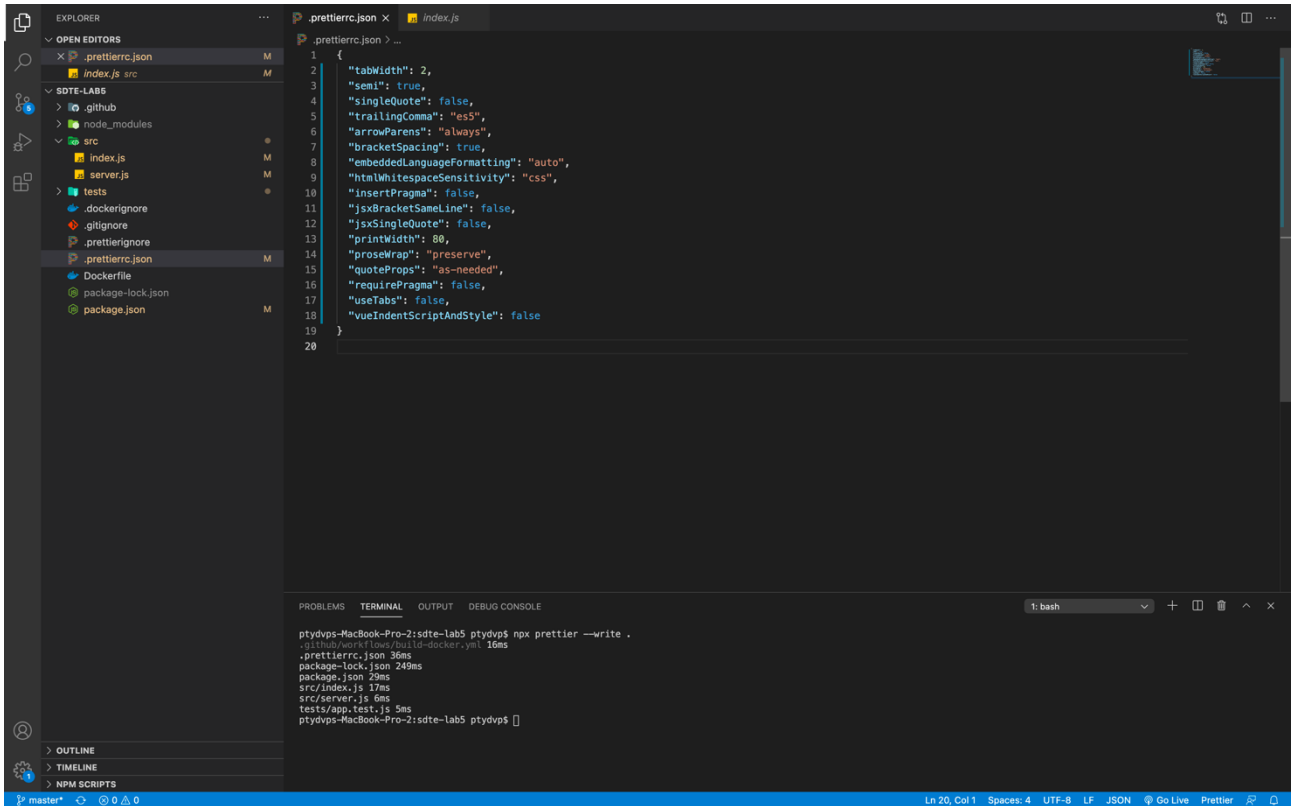
10. Build Docker Image โดยให้ตั้งชื่อ Image ว่า SDTE_LAB_5 และ tag v1
11. ตรวจสอบว่าเหลือแค่ไฟล์ที่จำเป็นตามข้อ 9 หรือไม่ (ทดสอบ Image ว่าเหลือไฟล์อะไรใน Image บ้าง ได้ด้วยคำสั่ง "docker run --rm --entrypoint "/bin/ls" sdte_lab_5:v1 -a" ดังตัวอย่างในภาพข้อ 9)
12. สร้าง Image Repository บน Docker Hub โดยให้ตั้งชื่อ Repository ชื่อว่า "SDTE_LAB_5" Tag Image SDTE_LAB_5:v1 ที่สร้างในข้อ 9 ใหม่ ให้ตามรูปแบบสำหรับเก็บขึ้น Docker Hub (ex. USERNAME/SDTE_LAB_5:v1) และทำการ Push Image ขึ้นไปเก็บใน Repository ที่สร้างขึ้นในข้อ 10

Post Lab บันทึกผลในหน้าที่ 5

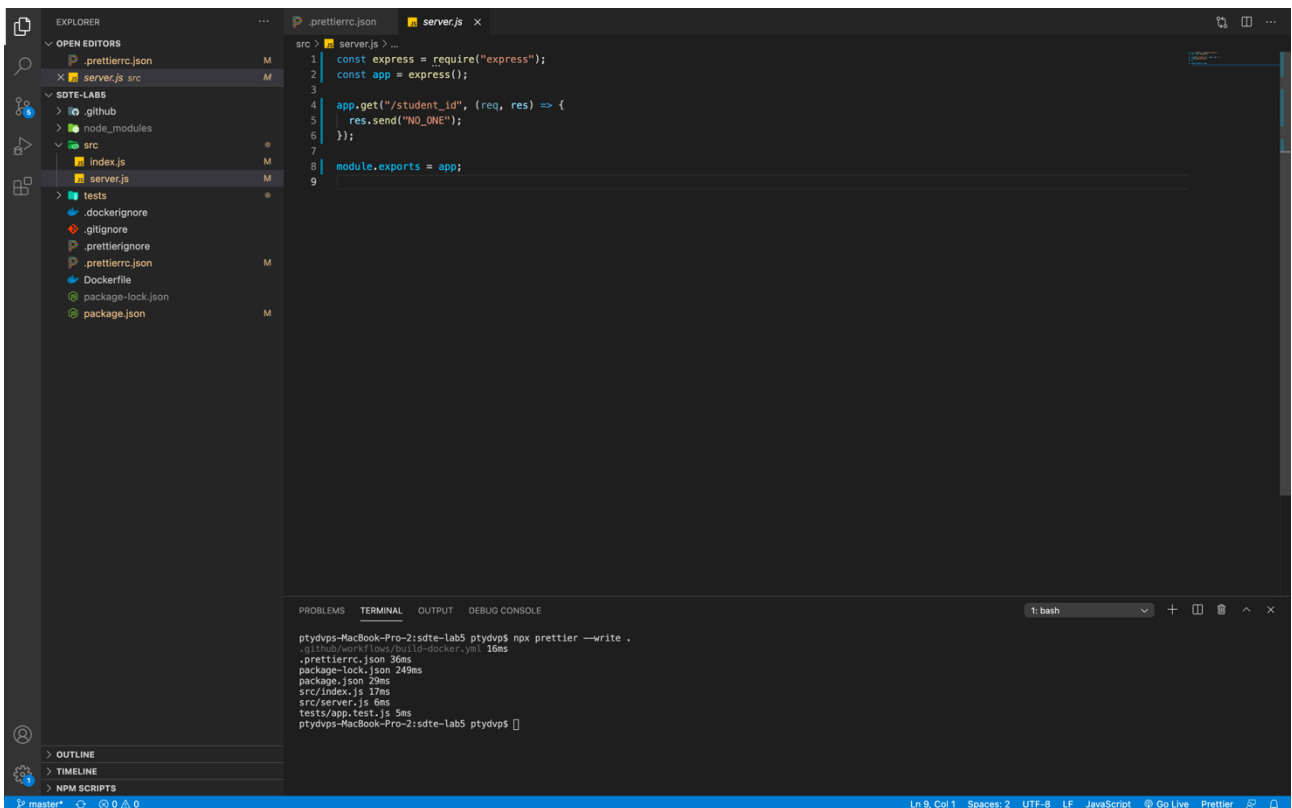
13. Commit ไฟล์ที่แก้ไขทั้งหมด และ Push เพื่อเก็บ Code ที่เสร็จแล้ว

LAB A - Prettier Result

1. Capture screen ไฟล์ `.prettierrc.json`



2. Capture screen ไฟล์ **src/server.js** หลังจากรัน lab ข้อที่ 5



LAB B - Jest Result

1. Capture screen output ของการรันคำสั่งในข้อที่ 6

```
ptydvps-MacBook-Pro-2:sdte-lab5 ptydvp$ npx jest
FAIL tests/app.test.js
  StudentID Endpoints
    ✕ Should get student id (39 ms)

● StudentID Endpoints > Should get student id

expect(received).not.toEqual(expected) // deep equality

Expected: not "NO_ONE"

   6 |     const res = await request(server).get("/student_id");
   7 |     expect(res.statusCode).toEqual(200);
>  8 |     expect(res.text).not.toEqual("NO_ONE");
      |                           ^
   9 |   });
  10 | });
  11 |

at Object.<anonymous> (tests/app.test.js:8:26)

Test Suites: 1 failed, 1 total
Tests:       1 failed, 1 total
Snapshots:   0 total
Time:        1.356 s
Ran all test suites.
ptydvps-MacBook-Pro-2:sdte-lab5 ptydvp$
```

2. Capture screen ไฟล์ **src/server.js** เมื่อแก้ไขตามข้อที่ 7

```
src > JS server.js > ...
1 | const express = require("express");
2 | const app = express();
3 |
4 | app.get("/student_id", (req, res) => {
5 |   res.send("61070143");
6 | });
7 |
8 | module.exports = app;
9 |
```

3. Capture screen output ของการรันคำสั่งในข้อที่ 8

```
ptydvps-MacBook-Pro-2:sdte-lab5 ptydvp$ npx jest
PASS tests/app.test.js
  StudentID Endpoints
    ✓ Should get student id (19 ms)

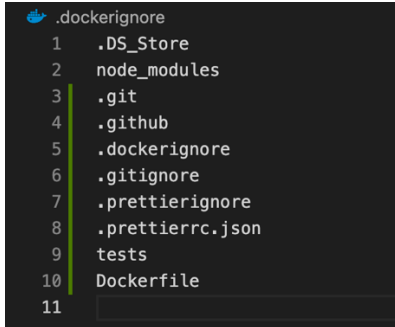
Test Suites: 1 passed, 1 total
Tests:       1 passed, 1 total
Snapshots:   0 total
Time:        0.683 s, estimated 2 s
Ran all test suites.
ptydvps-MacBook-Pro-2:sdte-lab5 ptydvp$
```

LAB C - Docker Build & Push to Registry

1. กรอก URL ของ Docker hub repository ที่สร้างในข้อที่ 10

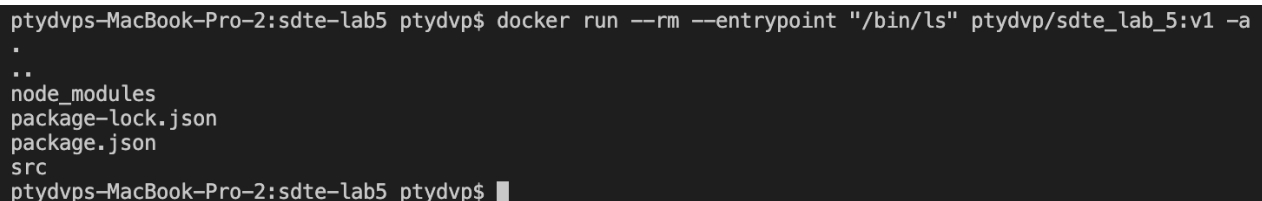
https://hub.docker.com/repository/docker/ptydvp/sdte_lab_5 << ตัวอย่าง ให้ replace ทับได้เลย

2. Capture screen ไฟล์ **.dockerignore**



```
.dockerignore
1  .DS_Store
2  node_modules
3  .git
4  .github
5  .dockerignore
6  .gitignore
7  .prettierrignore
8  .prettierrc.json
9  tests
10 Dockerfile
11
```

3. รัน “docker run --rm --entrypoint “/bin/ls” **otvisa01/sdte_lab_5:v1** -a” และ Capture screen output ว่าใน Image เหลือไฟล์อะไรบ้าง (เปลี่ยน IMAGE **otvisa01/sdte_lab_5:v1** เป็น image ของนักศึกษา ที่เก็บบน repository ในข้อที่ 13 ด้วย)



```
ptydvps-MacBook-Pro-2:sdte-lab5 ptydvp$ docker run --rm --entrypoint "/bin/ls" ptydvp/sdte_lab_5:v1 -a
.
..
node_modules
package-lock.json
package.json
src
ptydvps-MacBook-Pro-2:sdte-lab5 ptydvp$
```

Post Lab

1. URL ของ Git hub repository ที่เก็บ Source code ของ Lab นี้

<https://github.com/ptydvp/sdte-lab5>