**API DOCUMENT**

**ติดตั้ง Node.js และ Postman**

Node.js - ตัวติดตั้งอยู่ใน Folder installer/node-v6.2.2-x64.msi

Postman - ลิงค์ในการติดตั้งอยู่ใน File postman-link-download.txt

**SQL**

1. ไฟล์ .sql อยู่ใน Folder sql/todo\_task.sql
2. Table ชื่อ todo\_task
3. Import ใน MySQL ได้เลยครับ

**การ Run server (Window)**

1. git clone จาก <https://github.com/siggysic/AssignTodo.git>
2. เข้า Command line แล้วไปยัง Path directory ที่ติดตั้ง เช่น d:\AssignTodo
3. ใช้คำสั่ง node server เพื่อ run ไฟล์ server.js

**การ Setup Postman**

1. เมื่อติดตั้งเสร็จเรียบร้อยแล้วใน Tabs ที่ชื่อ Headers ให้ตั้ง key เป็น Content-Type และ value เป็น application/json (ติ้กถูกเพื่อใช้งาน)
2. Tabs ที่ชื่อ Body ใส่ JSON โดยต้องกำหนด key ให้ตรงกับ Field ใน Database เช่น

{

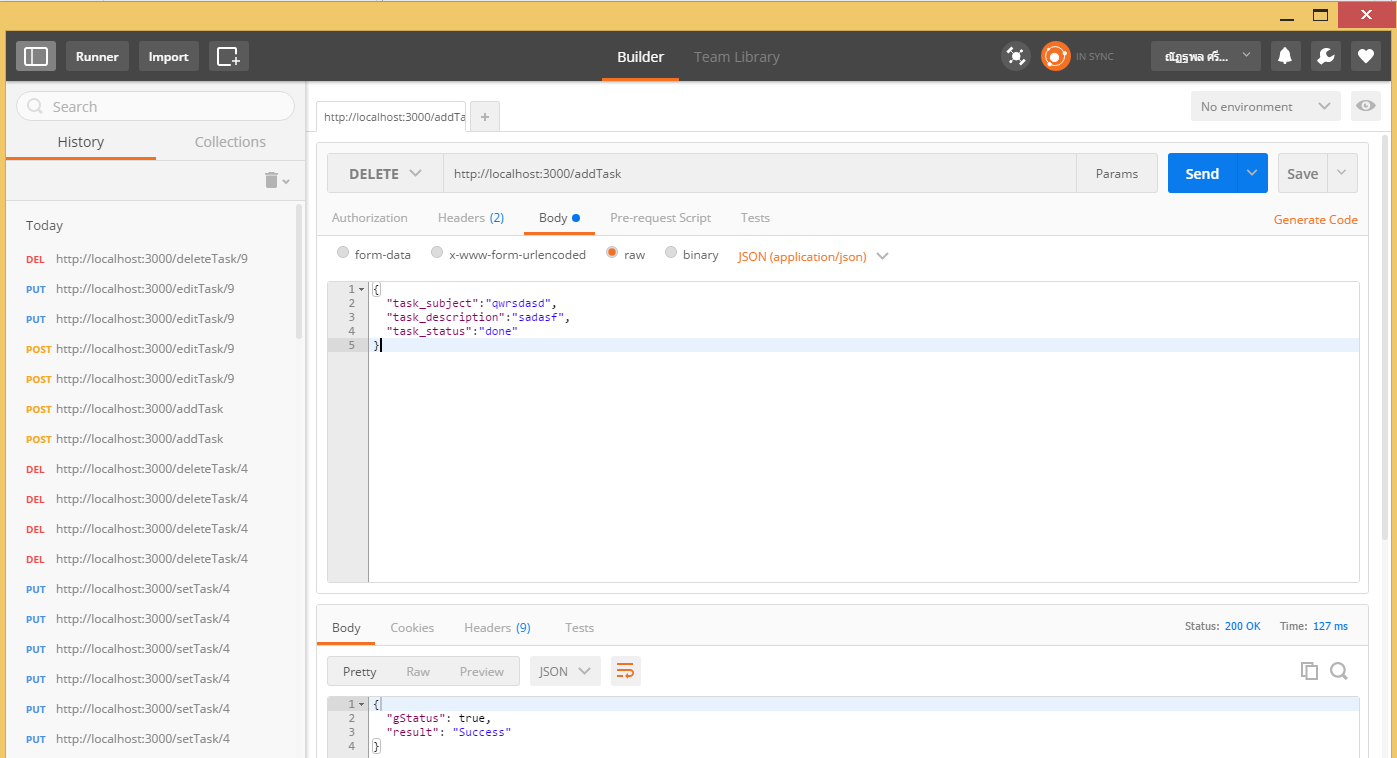
"task\_subject":"สวัสดีครับ",

"task\_description":"สบายดีไหมครับ",

"task\_status":"pending"

}

1. เลือก method และใส่ URL ให้ครบถ้วน แล้วกด Send



**Database**

1. task\_id เป็น primary key ชนิด int (11)
2. task\_subject คือชื่อของ task ชนิด varchar (200)
3. task\_description คือคำอธิบาย task ชนิด varchar(350)
4. task\_status คือสถานะของ task ชนิด varchar(100) (pending และ done)

**API DOCUMENT**

1. <http://localhost:3000/getTasks> ดึงข้อมูลใน Task ทั้งหมดออกมาแสดง ที่อยู่ในรูปของ JSON และmethod เป็น GET
2. <http://localhost:3000/getTask/:id> ดึงข้อมูลใน Task แค่อันเดียวโดย :id จะเป็นตัวเลข (task\_id ตอนนี้มีแค่ 1, 2, 3) อยู่ในรูปของ JSON และ method เป็น GET
3. <http://localhost:3000/addTask> เพิ่มข้อมูลใน Task โดยต้องอยู่ในรูปแบบของ JSON และ method เป็น POST (ตรงนี้ผมใช้ Postman ในการส่งแบบ POST, PUT, DELETE ครับ)
4. <http://localhost:3000/editTask/:id> แก้ไขข้อมูลใน Task แค่อันเดียวโดย :id จะเป็นตัวเลข (task\_id ตอนนี้มีแค่ 1, 2, 3) data ที่จะเพิ่มอยู่ในรูปแบบของ JSON สามารถแก้ไขได้ทั้ง task\_subject, task\_description, task\_status และ method เป็น PUT (ตรงนี้ผมใช้ Postman ในการส่งแบบ POST, PUT, DELETE ครับ)
5. [http://localhost:3000/setTask/:id](http://localhost:3000/editTask/:id) แก้ไขข้อมูลใน Task แค่อันเดียวโดย :id จะเป็นตัวเลข (task\_id ตอนนี้มีแค่ 1, 2, 3) data ที่จะเพิ่มอยู่ในรูปแบบของ JSON สามารถแก้ไขได้แค่ task\_status (ถ้าใส่ key อื่นที่ไม่ใช่ task\_status มาจะไม่ update และไม่ error) และ method เป็น PUT (ตรงนี้ผมใช้ Postman ในการส่งแบบ POST, PUT, DELETE ครับ)
6. <http://localhost:3000/deleteTask/:id> ลบข้ออมูลใน Task แค่อันเดียวโดย :id จะเป็นตัวเลข (task\_id ตอนนี้มีแค่ 1, 2, 3) และ method เป็น DELETE (ตรงนี้ผมใช้ Postman ในการส่งแบบ POST, PUT, DELETE ครับ)