

1. To setup the project first run **npm install** to install the dependencies.
2. To run the project run the command: **node server.js**

As per the requirement I have created 6 API's which are as follows :

1. Create User(<http://localhost:7000/api/controller/createUser>) :

This api will create a user in Users table which can act as an assigner or an assignee

This can be accessed by using the curl command or postman by giving appropriate data

Postman : Example Json for the postman

```
{
  "email": "tushar.kudal@gmail.com",
  "name": "tushar",
  "surname": "kudal"
}
```

Curl : run the command from terminal

```
curl --data "email=tk@example.com&name=tushar&surname=kudal"
http://localhost:7000/api/controller/createUser
```

2. Create Project(<http://localhost:7000/api/controller/createProject>):

This api will create a project in Projects table it has userId column which acts as assigner for the project

Can be accessed using postman or curl. Sample Json :

```
{
  "name": "tushar",
  "body": "req.body.description",
  "status": "active",
  "UserId": "1" // Assigner of the project which need's to be assigned at the time of project
  creation
}
```

Will throw an error if UserId is not present in Users table.

3. Create Task(<http://localhost:7000/api/controller/createTask>):

This api will create a task in Tasks table

Sample Json :

```
{
  "name": "tushar",
  "description": "req.body.description",
  "score": 1,
  "status": "active",
  "userId": "1", // Assigner of the task which need's to be assigned at the time of project creation

  "projectId": "1" // Assign the task to the project
}
```

4. Find Users(<http://localhost:7000/api/controller/findUsers?name=tushar&surname=kudal>):

This api will find the users this can be accessed by entering the above url in browser of postman also can be filtered using name and surname as follow :

Filter using name :

<http://localhost:7000/api/controller/findUsers?name=tushar>

Filter using surname :

<http://localhost:7000/api/controller/findUsers?surname=tushar>

5. FindTasks([http://localhost:7000/api/controller/findTasks?status=\['active','inactive'\]&description=req.body.description&score=1&name=tushar&assigner_id=1](http://localhost:7000/api/controller/findTasks?status=['active','inactive']&description=req.body.description&score=1&name=tushar&assigner_id=1)):

This api return the list of tasks based on different filters given in requirement, all the fields are mandatory in this api.

6. FindProjects([http://localhost:7000/api/controller/findProjects?status=\['active','inactive'\]&description=req.body.description&name=tushar&assigner_id=1](http://localhost:7000/api/controller/findProjects?status=['active','inactive']&description=req.body.description&name=tushar&assigner_id=1)):

This api return the list of projects based on different filters given in requirement, all the fields are mandatory in this api.

Any violation will given an error eg : if a task is created without a project, or a user is assigned which is not in users table