

Assignment-3

Create an express API to store registration details from the url

Procedure:

Create a folder ,install the express using cmd “`npm install express --save`”

also install the “CORS” “`npm install CORS --save`”

install mongodb “`npm i mongodb --save`”

now create a file api.js

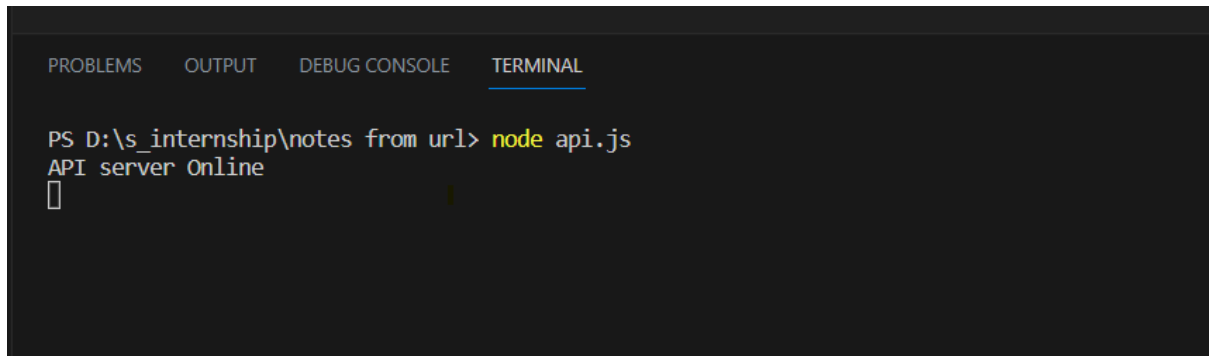
api.js

```
const express=require('express');
const url=require('url');
const cors=require('cors');

const {MongoClient}=require('mongodb');
const mongourl='mongodb://127.0.0.1:27017';
const client=new MongoClient(mongourl);
const dbName='Ammu';
var api=express()
api.use(cors())
api.get('/',function(request,response){
    response.send("API server is online")
})
api.get('/insertnotes',async function(request,response){
    urldata=url.parse(request.url,true);
    var notes=urldata.query.notes;
    var owner=urldata.query.owner;
    console.log(notes,owner);
    await client.connect();
    console.log('connected with mongodb server');
    const db=client.db(dbName);
    const collection=db.collection('expressnotes');
    const result=await collection.insertOne({'notes':notes,'owner':owner});
    console.log(result);
    response.json({status:'success'});
})
api.get('/newregistration',function(request,response){
    urldata=url.parse(request.url,true);
    var name=urldata.query.name;
    var mobile=urldata.query.mobile;
    var password=urldata.query.password;
    var email=urldata.query.email;
    var org=urldata.query.org;
    console.log(name,mobile,password,email,org);
    response.json({status:'success'});
})
```

```
})  
api.listen(2000,function(){  
  console.log('API server Online');  
})
```

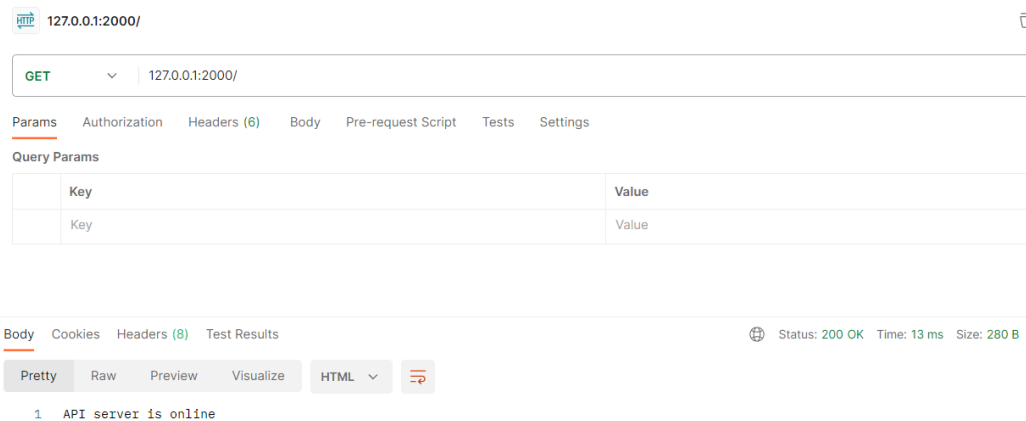
Run the code “ node api.js”



The screenshot shows a terminal window with a dark background. At the top, there are tabs for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', and 'TERMINAL', with 'TERMINAL' being the active tab. The terminal prompt is 'PS D:\s_internship\notes from url>'. The user has entered the command 'node api.js', and the output is 'API server Online'. A cursor is visible on the line following the output.

Now ,at the server side,the api server is in online.

Open the postman tool app and select the get method,and call the default handler



The server side,also is on online,now call the “/newregistration” handler and enter the values

127.0.0.1:2000/newregistration?name=Ammu&email=ammu123@gmail.com&mobile=9347244193&org=makeskilled&password=abc123 Add to collection

GET 127.0.0.1:2000/newregistration?name=Ammu&email=ammu123@gmail.com&mobile=9347244193&org=makeskilled&password=abc123 Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookie:

Key	Value	Bulk Edit
<input checked="" type="checkbox"/> name	Ammu	
<input checked="" type="checkbox"/> email	ammu123@gmail.com	
<input checked="" type="checkbox"/> mobile	9347244193	
<input checked="" type="checkbox"/> org	makeskilled	
<input checked="" type="checkbox"/> password	abc123	
Key	Value	

Body Cookies Headers (8) Test Results Status: 200 OK Time: 12 ms Size: 287 B Save Response

Pretty Raw Preview Visualize JSON Copy

```
1
2  "status": "success"
3
```

Now, the status is success. The inserted data is visible at the server side also

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

PS D:\s_internship\notes from url> node api.js
API server Online
Ammu 9347244193 undefined ammu123@gmail.com makeskilled
Ammu 9347244193 abc123 ammu123@gmail.com makeskilled
█
```