



NODE JS + EXPRESS + MYSQL

CRUD

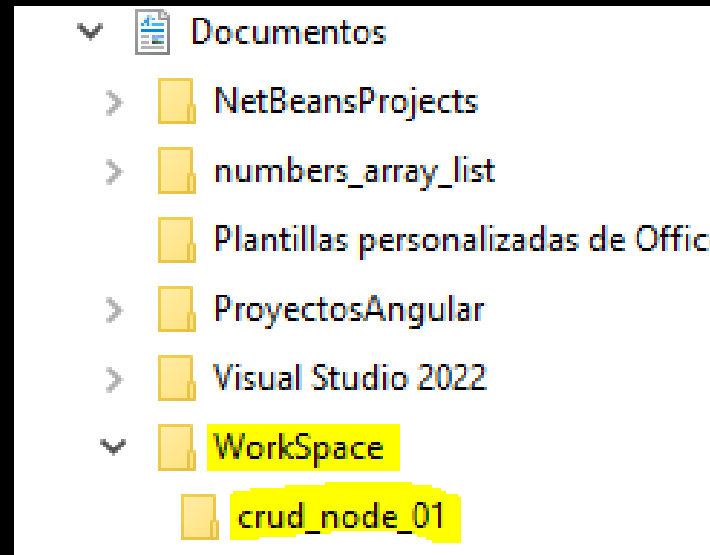
Nixon Duarte Acosta

PRIMERA PARTE

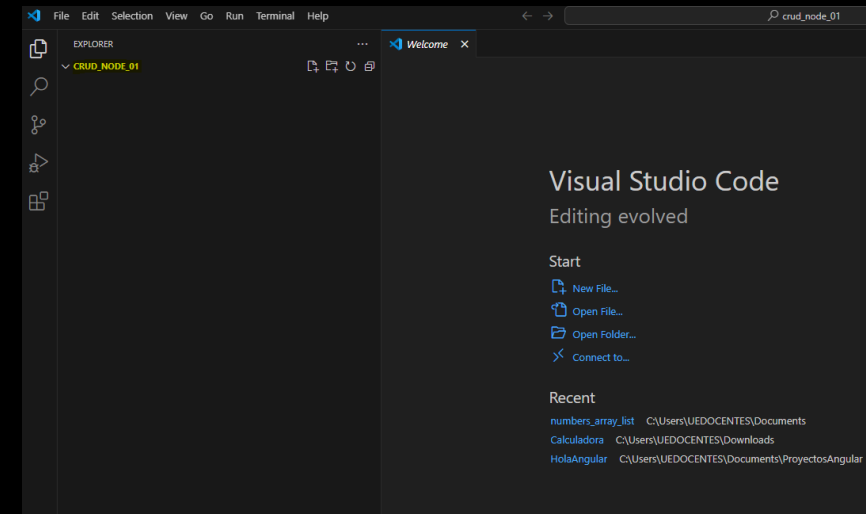
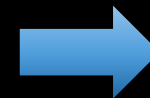


ESTRUCTURA DEL PROYECTO

1. Crear proyecto

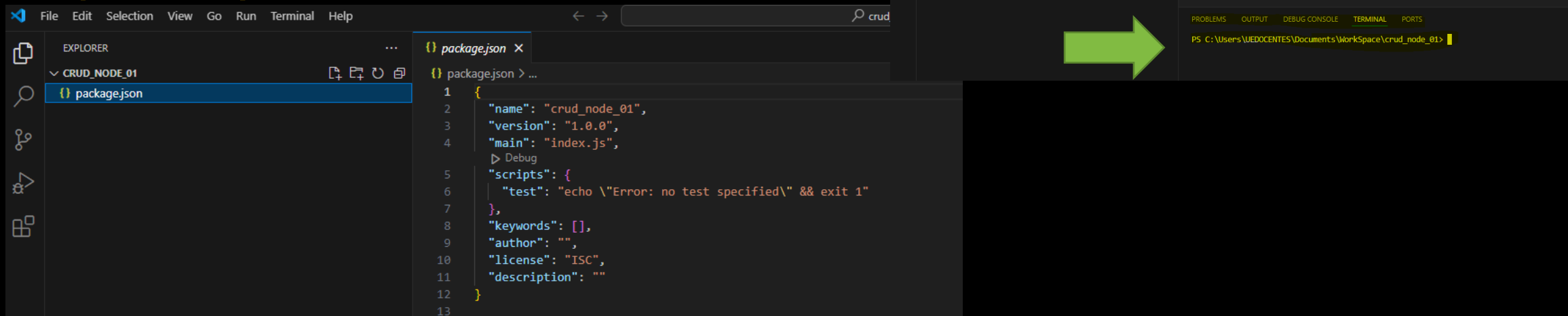


2. Abrir el folder en Visual Studio Code



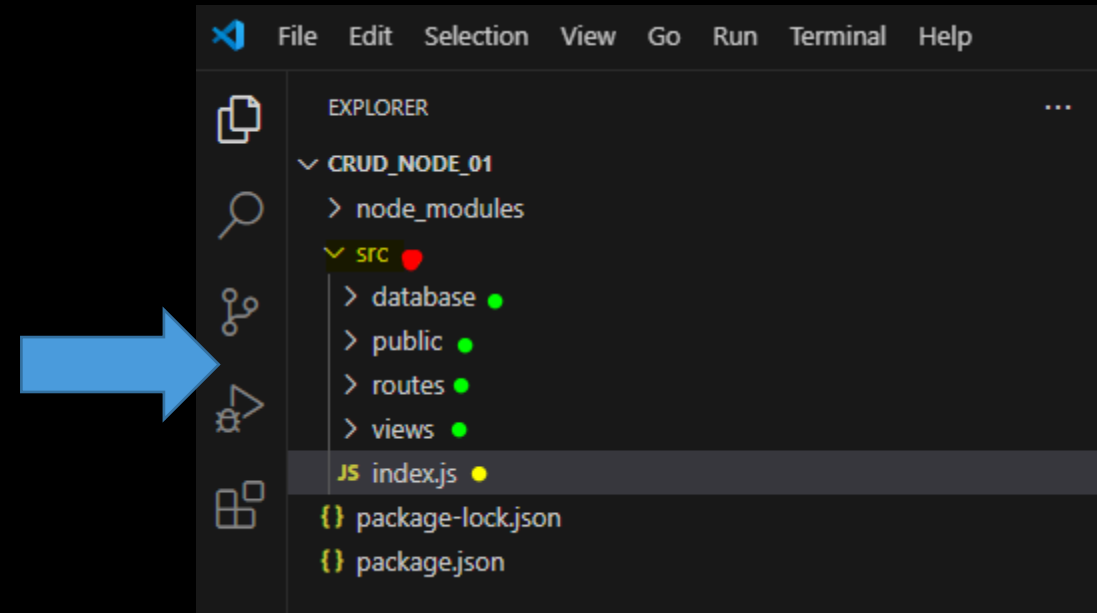
CONFIGURACIÓN DEL SERVIDOR

3. Abrir terminal en el IDE – VSC
 - Ctrl + shift + p
 - Buscar: *Create new terminal*
 - ...ó *abrir CMD* para ejecutar los comandos
4. Ejecutar comando para crear el `package.json`
 - **`npm init --y`**



CONFIGURACIÓN DEL SERVIDOR

5. Instalar los paquetes necesarios (librerías) que necesitamos para el funcionamiento de nuestro proyecto.
 - ***npm i express express-handlebars morgan mysql2***
 - Se genera la carpeta node_modules
6. Estructura del proyecto:



CONFIGURACIÓN DEL SERVIDOR

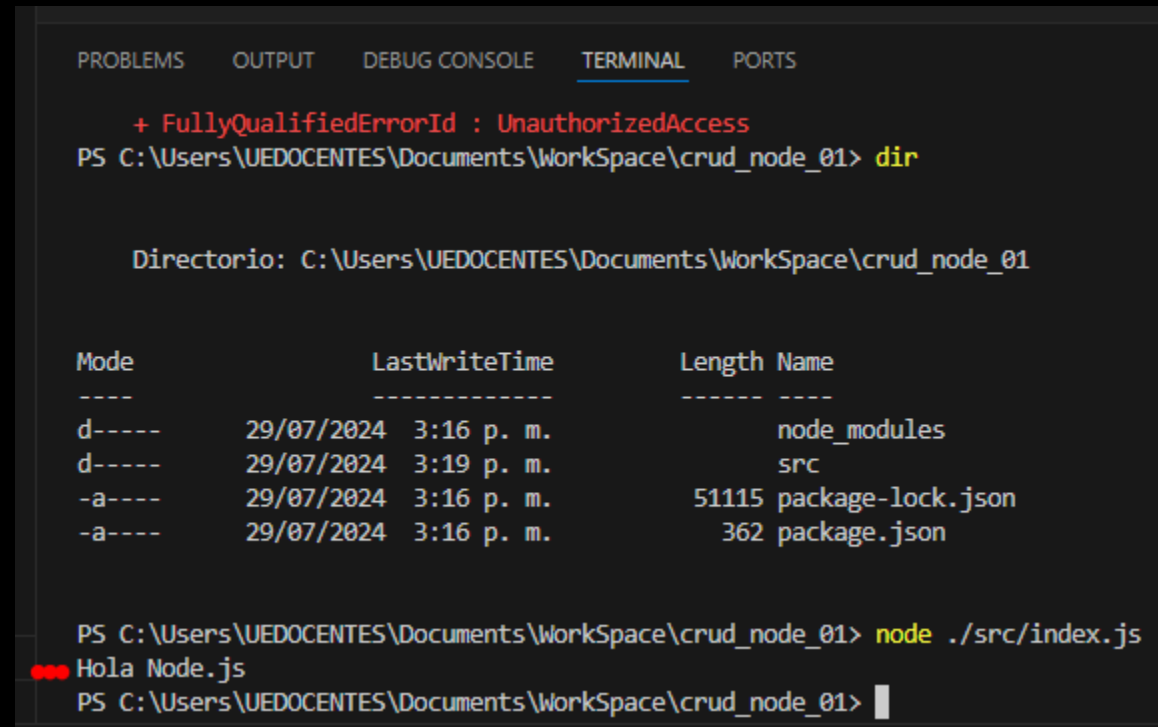
7. Crear *Hola Node.js*

- En *index.js*
 - Ingresar: **`console.log('Hola Node.js');`**
- En Terminal/CMD en el proyecto, ejecutar:

- **`node ./src/index.js`**

8. Instalar *nodemon* para facilitar la ejecución y no estar ejecutando el comando anterior:

- **`npm i nodemon -D`**



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

+ FullyQualifiedErrorId : UnauthorizedAccess
PS C:\Users\UEDOCENTES\Documents\Workspace\crud_node_01> dir

Directorio: C:\Users\UEDOCENTES\Documents\Workspace\crud_node_01

Mode                LastWriteTime         Length Name
----                -
d-----          29/07/2024   3:16 p. m.      node_modules
d-----          29/07/2024   3:19 p. m.          src
-a----          29/07/2024   3:16 p. m.    51115 package-lock.json
-a----          29/07/2024   3:16 p. m.     362 package.json

PS C:\Users\UEDOCENTES\Documents\Workspace\crud_node_01> node ./src/index.js
●●● Hola Node.js
PS C:\Users\UEDOCENTES\Documents\Workspace\crud_node_01>
```

CONFIGURACIÓN DEL SERVIDOR

9. Modificar package.json

10. Ejecutamos:

- ***npm run dev***



```
▶ Debug  
"scripts": {  
  "dev": "nodemon ./src/index.js"  
},
```



```
C:\windows\system32\cmd.exe  
at Function.executeUserEntryPoint [as runMain] (node:internal/modules/run_main:174:12)  
at node:internal/main/run_main_module:28:49 {  
  code: 'MODULE_NOT_FOUND',  
  requireStack: []  
}  
Node.js v20.13.0  
C:\Users\UEDOCENTES\Documents\Workspace\crud_node_01>npm i nodemon -D  
added 28 packages, and audited 159 packages in 11s  
30 packages are looking for funding  
  run `npm fund` for details  
found 0 vulnerabilities  
C:\Users\UEDOCENTES\Documents\Workspace\crud_node_01>npm run dev  
> crud_node_01@1.0.0 dev  
> nodemon ./src/index.js  
[nodemon] 3.1.4  
[nodemon] to restart at any time, enter `rs`  
[nodemon] watching path(s): *.*  
[nodemon] watching extensions: js,mjs,cjs,json  
[nodemon] starting `node ./src/index.js`  
• Hola Node.js •  
[nodemon] clean exit - waiting for changes before restart
```


CONFIGURACIÓN DEL SERVIDOR

11. Modificamos el package.json

- Agregamos
 - **"type": "module",**

```
{ } package.json > ...
1  {
2    "name": "crud_node_01",
3    "version": "1.0.0",
4    "main": "index.js",
5    "type": "module",
6    "scripts": {
7      "dev": "nodemon ./src/index.js"
8    },
9  }
```

12. Estructuramos el index.js

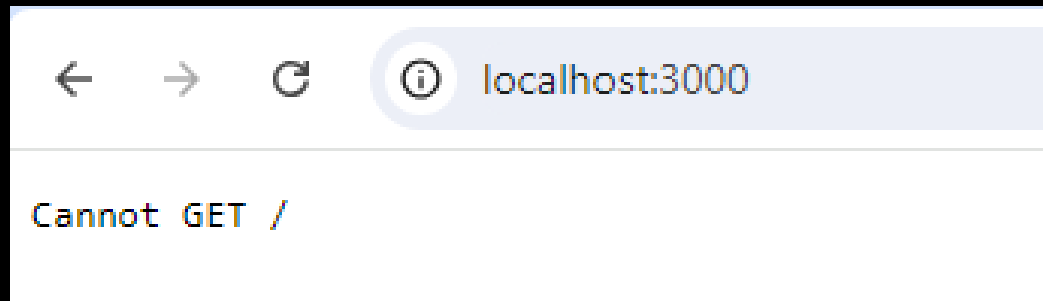
```
JS index.js x
src > JS index.js > app.listen() callback
1  import express from 'express'
2
3  //Initialization
4  const app = express();
5
6  //Setting
7  app.set('port', process.env.PORT || 3000);
8
9  //Middlewares
10
11 //Routes
12
13 //Public files
14
15 //Run server
16 app.listen(app.get('port'), ()=>
17   console.log('El server esta escuchando en el puerto', app.get('port')));
```


CONFIGURACIÓN DEL SERVIDOR

13. Ejecutamos o si esta en ejecución se puede ver por consola el mensaje:

- *El server esta escuchando en el puerto 3000*

14. Ahora podemos ir al navegador web e ingresar: *localhost:3000*

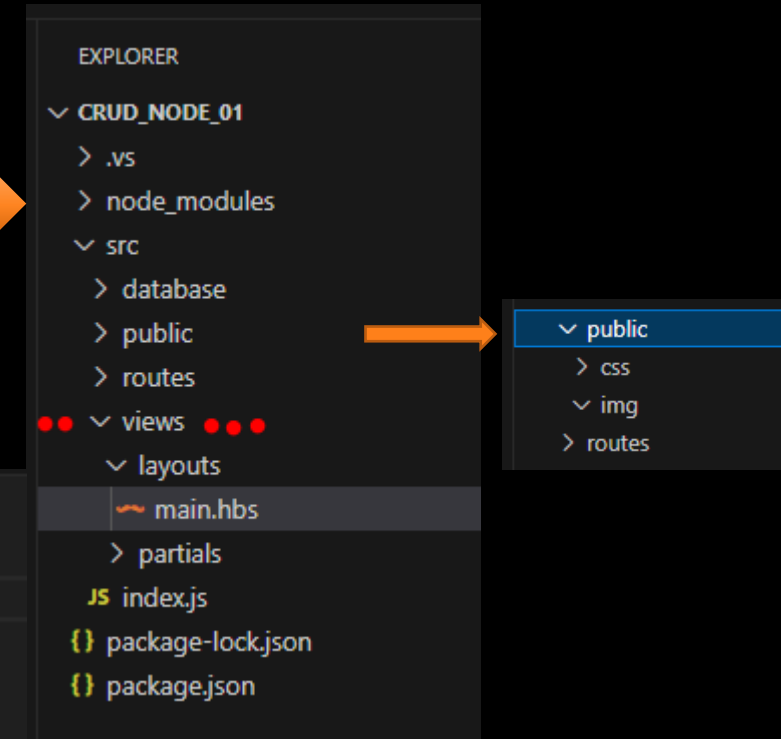


- *Esto significa que node.js ya se esta ejecutando, solo que no tenemos ningún routes.*

CONFIGURACIÓN DEL SERVIDOR

15. Agregamos en *views* las siguientes carpetas

16. Continuamos con la configuración del *main.hbs*




src > views > layouts > main.hbs > html


```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Document</title>
7 </head>
8 <body>
9   {{{body}}}
10 </body>
11 </html>
```

CONFIGURACIÓN DEL SERVIDOR

17. Continuamos con la configuración del *index.js*



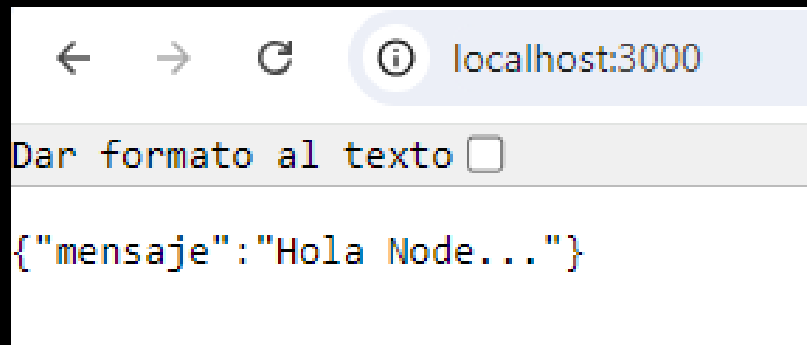
```
JS index.js X main.hbs
src > JS index.js > ...
1 import express from 'express'
2 import morgan from 'morgan';
3 import { engine } from 'express-handlebars';
4 import {join, dirname} from 'path'
5 import { fileURLToPath } from 'url';
6
7 //Initialization
8 const app = express();
9 const __dirname = dirname(fileURLToPath(import.meta.url));
10
11 //Setting
12 app.set('port', process.env.PORT || 3000);
13 app.set('views', join(__dirname, 'views'));
14 app.engine('.hbs', engine({
15   defaultLayout: 'main',
16   layoutsDir: join(app.get('views'), 'layouts'),
17   partialsDir: join(app.get('views'), 'partials'),
18   extname: '.hbs'
19 }));
20 app.set('view engine', '.hbs');
21
```



```
22
23 //Middlewares
24 app.use(morgan('dev'));
25 app.use(express.urlencoded({extended: false}));
26 app.use(express.json());
27
28 //Routes
29 app.get('/', (req, res) => {
30   res.json({"mensaje": "Hola Node..."})
31 });
32
33 //Public files
34 app.use(express.static(join(__dirname, 'public')));
35
36 //Run server
37 app.listen(app.get('port'), ()=>
38   console.log('El server esta escuchando en el puerto', app.get('port')));
```

CONFIGURACIÓN DEL SERVIDOR

- Al actualizar el navegador web, debe mostrar:



FIN PRIMERA PARTE



SEGUNDA PARTE

1. Continuamos con la configuración del *main.hbs*
 - Agregamos Bootstrap

```
JS index.js  main.hbs  index.hbs
src > views > layouts > main.hbs > html
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      {{!--Bootstrap--}}
7      <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet"
      Dwwykc2MPK8M2HN" crossorigin="anonymous">
8      <title>:: CRUD NODE.js ::</title>
9  </head>
10 <body>
11     {{{body}}}
12
13     {{!--Script Bootstrap--}}
14
15     <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"
        crossorigin="anonymous"></script>
16 </body>
17 </html>
```

CONFIGURACIÓN VIEWS

2. Creamos *index.hbs*

```
src > views > index.hbs > main.mt-5
1  <main class="mt-5">
2    <div class="container">
3      <div class="card mx-auto" style="width:500px;">
4        <div class="card-header">
5          
6          <h3 class="text-uppercase text-center p-4">CRUD con node js y mysql</h3>
7        </div>
8        <div class="card-body text-center">
9          <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Repellendus maiores pariatur at placeat vel quam corporis quod odit itaque
          eveniet cupiditate.</p>
10         <a href="#" class="btn btn-primary">Crear una persona</a>
11       </div>
12     </div>
13   </div>
14 </main>
```

3. Modificar en *index.js*

```
28 //Routes
29 app.get('/', (req, res) => {
30   res.render('index')
31 });
```