



**UNIVERSIDAD AUTÓNOMA DE CHIAPAS**

---



**FACULTAD: CONTADURÍA Y ADMINISTRACIÓN, CAMPUS I**

**LICENCIATURA: INGENIERÍA EN DESARROLLO Y TECNOLOGÍAS DE  
SOFTWARE**

**UNIDAD DE APRENDIZAJE: TALLER DE DESARROLLO 4**

**DOCENTE: D.S.C LUIS GUTIÉRREZ ALFARO**

**ALUMNO: EDSSON DANIEL LÓPEZ MENDOZA**

**NUMERO DE CONTROL: A210089**

**GRADO Y GRUPO: 6° "M"**

**ACTIVIDAD 2.- EJERCICIOS**

**TUXTLA GUTIÉRREZ, CHIAPAS**

**20 DE FEBRERO DE 2024**

Crear carpeta

```
PS C:\> mkdir ExamenTaller4

Directorio: C:\

Mode                LastWriteTime         Length Name
----                -
d-----          19/02/2024  10:54 p. m.             ExamenTaller4
```

Creamos proyecto

```
PS C:\> cd .\ExamenTaller4\
PS C:\ExamenTaller4> nest new api-examen
⚡ We will scaffold your app in a few seconds..

? Which package manager would you ❤️ to use? npm
CREATE api-examen/.eslintrc.js (688 bytes)
CREATE api-examen/.prettierrc (54 bytes)
CREATE api-examen/nest-cli.json (179 bytes)
CREATE api-examen/package.json (2019 bytes)
CREATE api-examen/README.md (3413 bytes)
CREATE api-examen/tsconfig.build.json (101 bytes)
CREATE api-examen/tsconfig.json (567 bytes)
CREATE api-examen/src/app.controller.ts (286 bytes)
CREATE api-examen/src/app.module.ts (259 bytes)
CREATE api-examen/src/app.service.ts (150 bytes)
CREATE api-examen/src/main.ts (216 bytes)
CREATE api-examen/src/app.controller.spec.ts (639 bytes)
CREATE api-examen/test/jest-e2e.json (192 bytes)
CREATE api-examen/test/app.e2e-spec.ts (654 bytes)

/ Installation in progress... 🔄
```

Creamos controladores

```
PS C:\ExamenTaller4\api-examen> nest g co examen
CREATE src/examen/examen.controller.ts (105 bytes)
CREATE src/examen/examen.controller.spec.ts (510 bytes)
UPDATE src/app.module.ts (340 bytes)
```

Creamos servicios

```
PS C:\ExamenTaller4\api-examen> nest g s examen
CREATE src/examen/examen.service.ts (94 bytes)
CREATE src/examen/examen.service.spec.ts (478 bytes)
UPDATE src/app.module.ts (412 bytes)
```

Creamos modulo

```
PS C:\ExamenTaller4\api-examen> nest g mo examen
CREATE src/examen/examen.module.ts (87 bytes)
UPDATE src/app.module.ts (479 bytes)
```

Creamos Carpeta dto dentro de la carpeta de donde se crearon los modulos, controladores y servicios, creamos archivos dto.ts e interfaces.ts

Dto.ts

```
src > examen > dto > TS examen.dto.ts > ExamenDto
1  export class ExamenDto {
2      readonly description: string;
3      readonly isDone: boolean;
4  }
```

Interfaces.ts

```
src > examen > dto > TS examen.interface.ts > ExamenDto
1  export interface ExamenDto {
2      description: string;
3      isDone: boolean;
4  }
```

Configuración

```
src > examen > TS examen.controller.ts > ExamenController
1  import { Controller, Get, Render } from '@nestjs/common';
2  import { METHODS } from 'http';
3  import { Request } from '@nestjs/common';
4
5
6  @Controller()
7  export class ExamenController {
8      @Get()
9      @Render('index')
10     root() {
11         return { message: 'html cargo exitosamente UwU!' };
12     }
13 }
14
```

```
src > examen > examen.module.ts > ...
1  import { Module } from '@nestjs/common';
2  import { ExamenController } from '../examen.controller';
3  import { ExamenService } from '../examen.service';
4
5  @Module({
6    imports: [],
7    controllers: [ExamenController],
8    providers: [ExamenService],
9  })
10 export class ExamenModule {}
11
```

```
src > app.controller.ts > AppController > Hello
1  import { Controller, Get, Render } from '@nestjs/common';
2  import { AppService } from '../app.service';
3
4  @Controller('/')
5  export class AppController {
6    constructor(private readonly appService: AppService) {}
7
8    @Get('/')
9    @Render('index')
10   getHello() {
11     return { message: 'Hola Edsson Nestjs' };
12   }
13   @Get('/examen')
14   @Render('examen')
15   Hello() {
16     return { message: 'Hola Edsson Nestjs' };
17   }
18 }
19
```

```
src > app.module.ts > ...
1  import { Module } from '@nestjs/common';
2  import { AppController } from '../app.controller';
3  import { AppService } from '../app.service';
4  import { ExamenController } from '../examen/examen.controller';
5  import { ExamenService } from '../examen/examen.service';
6  import { ExamenModule } from '../examen/examen.module';
7  import { ServeStaticModule } from '@nestjs/serve-static';
8  import { join } from 'path';
9
10 @Module({
11   imports: [
12
13   ],
14   controllers: [AppController],
15   providers: [AppService],
16 })
17 export class AppModule {}
18
```

```

src > main.ts > bootstrap
1  import { NestFactory } from '@nestjs/core';
2  import { AppModule } from './app.module';
3
4  import { NestExpressApplication } from '@nestjs/platform-express';
5  import { join } from 'path';
6
7  async function bootstrap() {
8      const app = await NestFactory.create<NestExpressApplication>(AppModule);
9
10     app.useStaticAssets(join(__dirname, '..', 'public'));
11     app.setBaseViewsDir(join(__dirname, '..', 'views'));
12     app.setViewEngine('hbs');
13
14     await app.listen(3000);
15 }
16 bootstrap();
17

```

Creamos carpeta views para los index con hbs

Descargamos hbs

```
PS C:\ExamenTaller4\api-examen> npm install --save hbs
```

Creamos los hbs para las rutas que se hacen desde app.controles.ts y le damos estructura al index

