

Proyecto Final Trimestre 1: Base de datos 'Empleados' mediante TYPESCRIPT - MONGODB - MONGOOSE - NODEJS.

Autor: Juan Rivas Ibáñez

El objetivo del proyecto es crear una clase denominada Empleados, hemos empleado lenguaje Typescript para la creación de la clase, los métodos, schema, etc... Hemos conectado nuestro cluster en MongoDB con el proyecto, y mediante distintas rutas podemos realizar consultas, y con el programa Postman además podemos insertar y borrar datos directamente. Además tenemos un CRUD en el que podemos trabajar en local a través de la consola de nuestro programa Visual Studio.

Nuestra clase empleados pertenece a la base de datos de recursos humanos de una empresa, en ella podemos comprobar el nombre de los empleados, el DNI, que será el identificador principal, la fecha de nacimiento, el teléfono personal, si posee un contrato fijo o temporal, el sueldo anual, el campo profesional donde realiza su jornada laboral en la empresa y por último la fecha de finalización del contrato.

Iniciar proyecto node.js

```
npm init -y
```

Iniciar proyecto typescript

```
tsc --init // Si global.
```

```
npx tsc --init // Si local.
```

Para crear el repositorio local

```
git init
```

Cambiamos la configuración del tsconfig.json

```
"target": "es6",
```

```
"outDir": "./build",
```

Instalamos los tipos de datos y módulos de desarrollo

```
npm install @types/node @types/mongoose @types/express @types/morgan nodemon  
typescript -D
```

Cambiamos el package.json para acceder más fácilmente a las distintas opciones como son compilar, etc:

```
"scripts": {
```

```
"ts": "tsc -w",
```

```
"dev": "nodemon ./build/server.js",
```

```
"start": "node ./build/server.js"  
},
```

Para compilar

```
npm run ts
```

Para ejecutar en desarrollo:

```
npm run dev
```

Para ejecutar en producción:

```
npm start
```

Para identificarse:

```
localhost:3000/id/user&password
```

Para listar los empleados de nuestra BD:

```
localhost:3000/empleados
```

En identificacionRoutes:

Hay que decidir si usamos BD Local o Atlas:

```
setBD(false, user, password) // true BD Local; false BD Atlas
```

Si queremos usar POSTMAN usaremos FALSE.

Ahora lo subimos a un repositorio en GITHUB:

```
git init
```

Creamos el repositorio en Github con el mismo nombre del proyecto.

Ahora subimos nuestro proyecto a Github:

```
git add .
```

```
git commit -m "first commit"
```

```
git remote add origin https://github.com/...
```

```
git push -u origin master
```