

# Crear la base de Datos de MongoDB y conexión desde Node

**ANÁHUAC**

A graphic featuring a blue square with a white border, containing a quote in white text. The square is set against a background of light blue wavy lines. The entire graphic is centered on a solid orange background.

**"El poder se ha  
trasladado al  
cliente y no va a  
volver."**

- Geoffrey Moore

**ANÁHUAC**

**En esta semana realizaremos lo siguiente:**

- 1. Crear base de Datos dentro de MongoDB con Studio 3T**
  - a. Crear una colección**
  - b. Crear un documento**
- 2. Conectar MongoDB con NodeJS y con nuestro proyecto NodeJS**
- 3. Crear un servidor con NodeJS y express (sistema de ruta y recibir peticiones HTTP)**



# IMPORTANTE



Deberás tomar capturas de pantalla para documentar las evidencias del proceso que seguiste y los resultados obtenidos.

En donde veas la imagen de la cámara,  significa que debes tomar evidencia como la que se muestra en esa parte del ejercicio.

Revisa en la descripción de la actividad los detalles de cómo entregar tu evidencia.

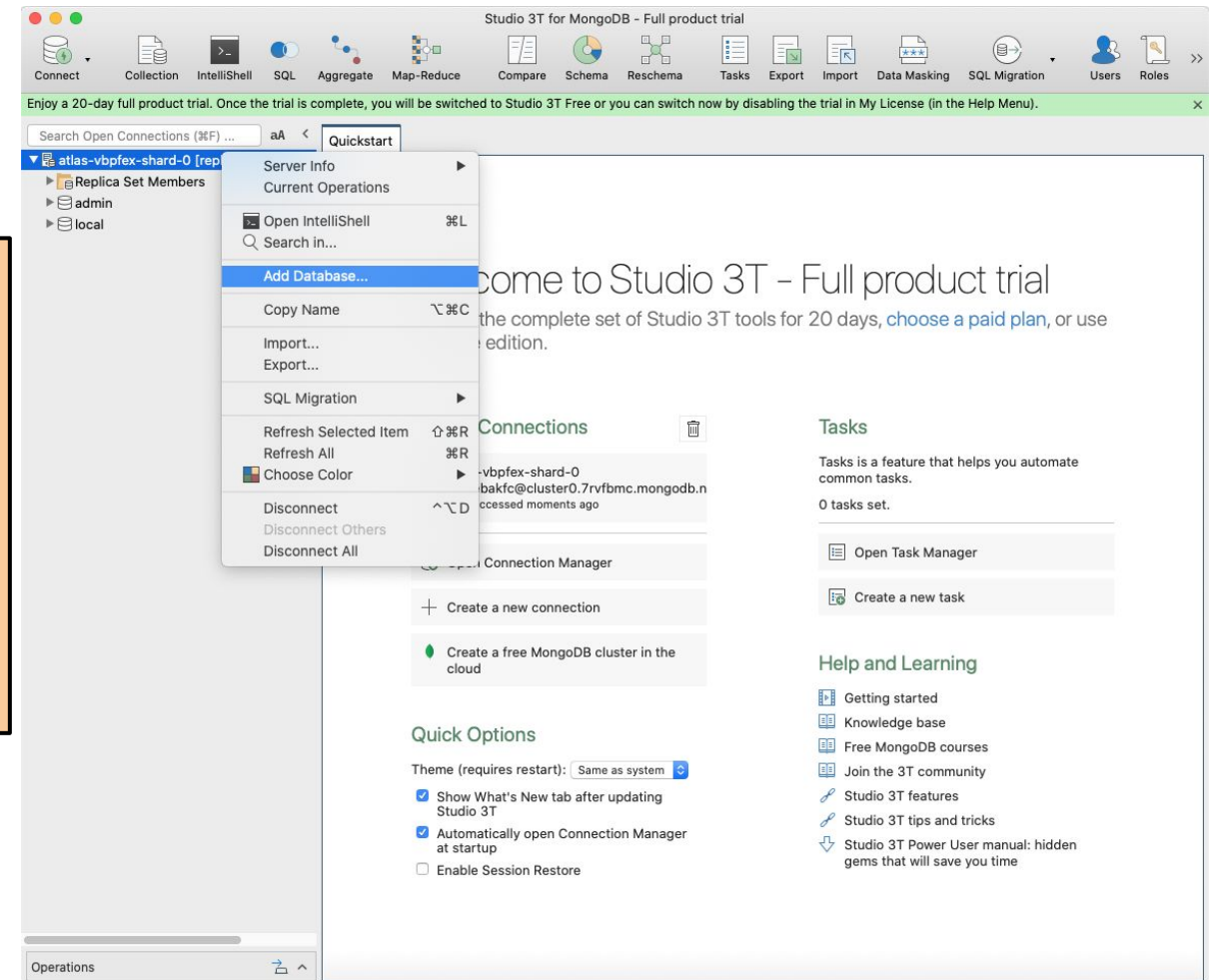


# 1. Crear base de Datos dentro de MongoDB con Studio 3T

Es muy importante que hayas realizado la actividad de la semana 10.

- Instalar MongoDB y Studio 3T.
- En MongoDB, conectar al cluster, generar y copiar la URL
- En Studio 3T pegar la URL y conectar.

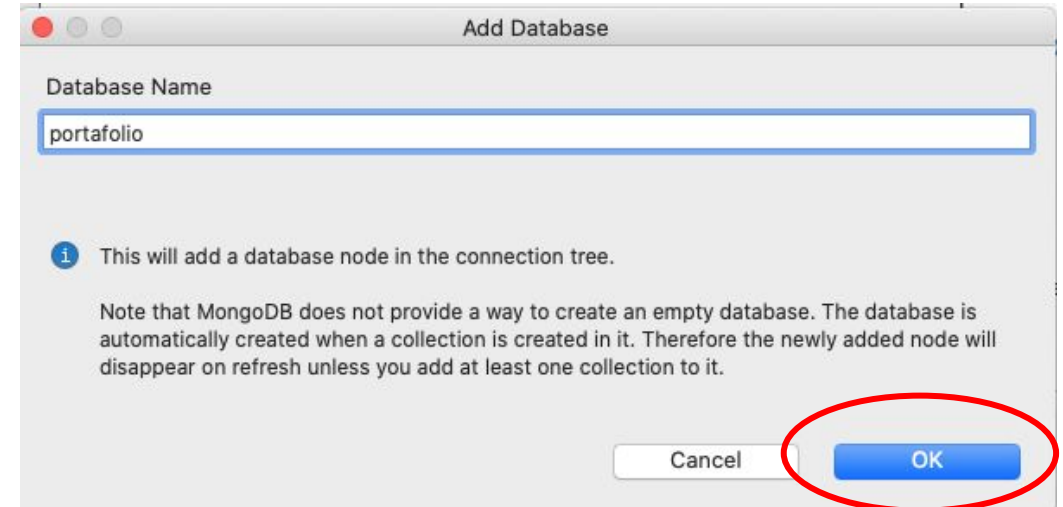
1. Ahora, abre Studio 3T
2. Haz click derecho sobre la conexión
3. Haz click en <Add Database>



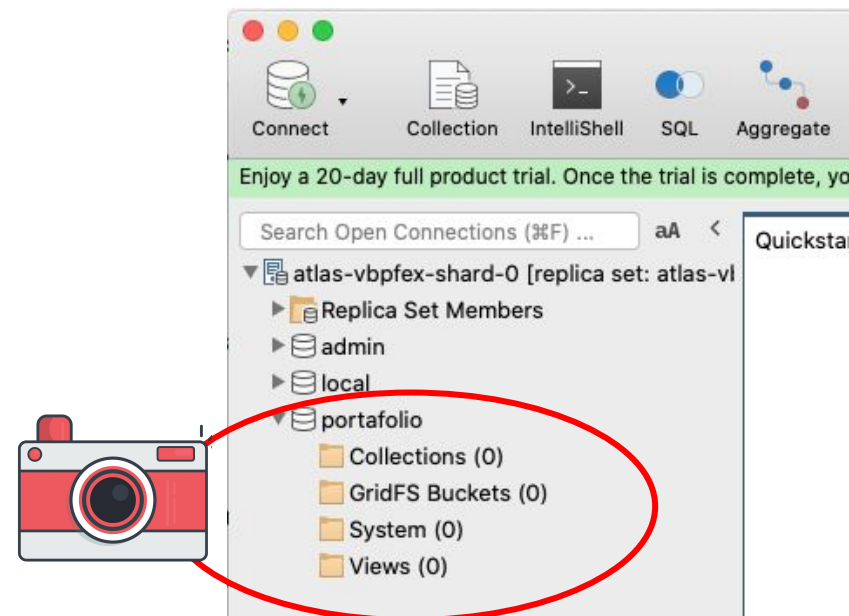


# 1. Crear base de Datos dentro de MongoDB con Studio 3T

4. Escribe un nombre para la base de datos.
5. Haz click en <OK>



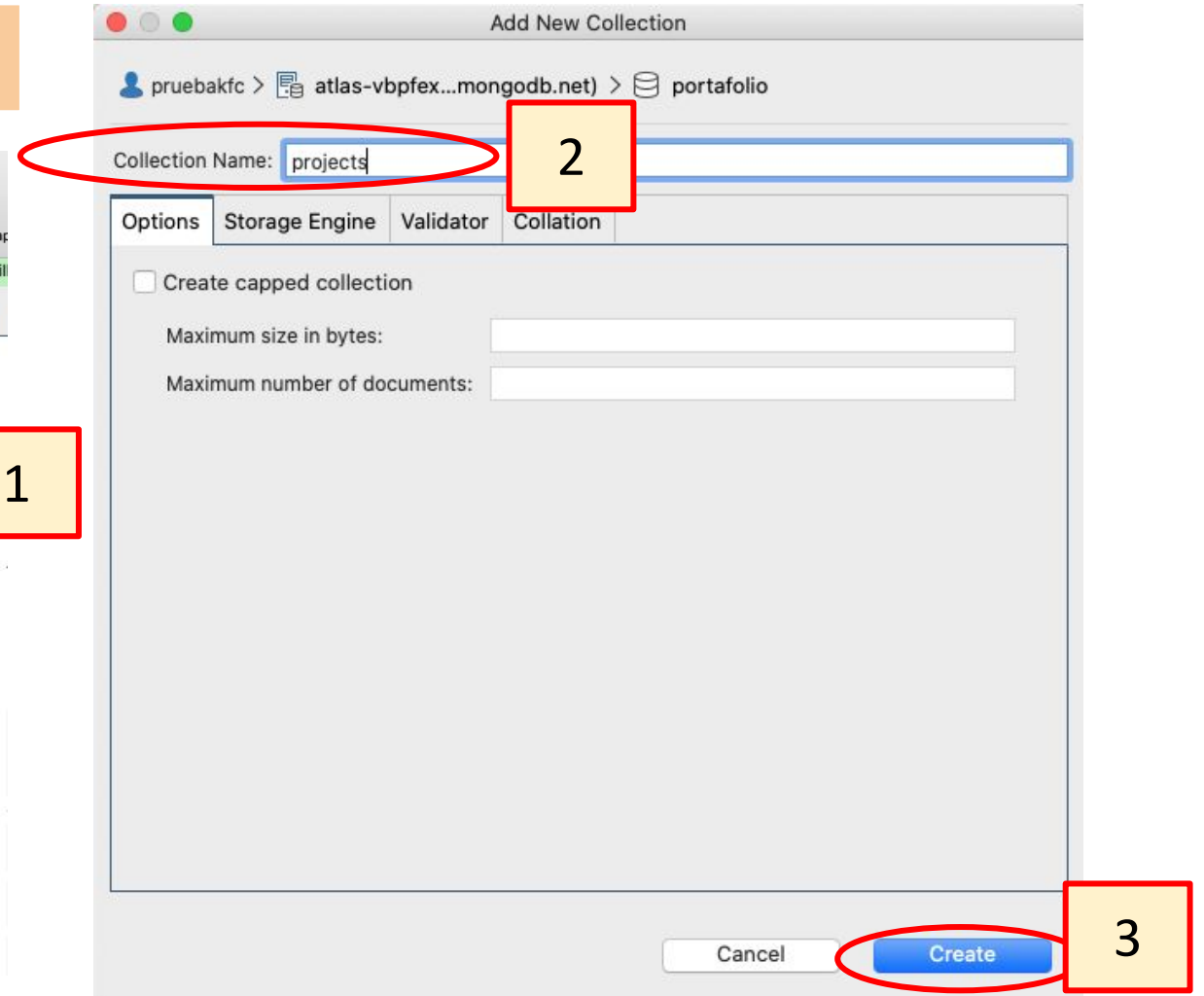
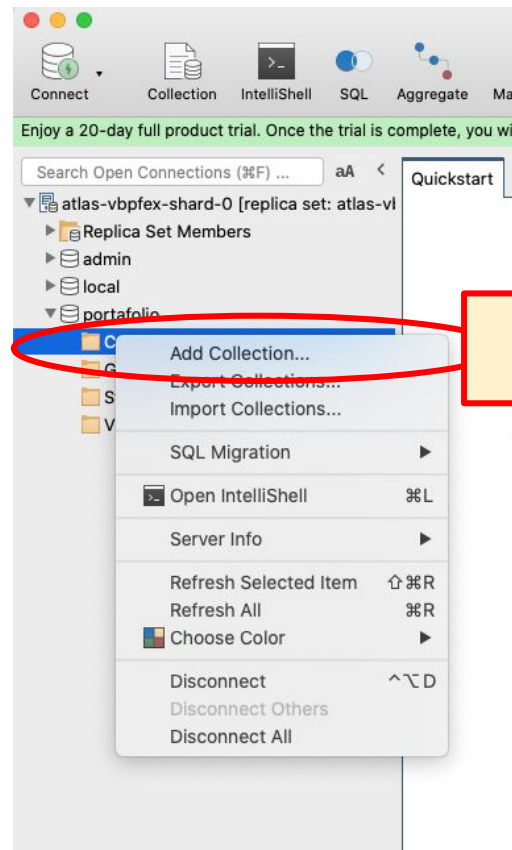
6. Podrás ver que la base de datos se ha creado con éxito y contiene varias carpetas.





# 1. Crear base de Datos dentro de MongoDB con Studio 3T

## a. Crear una colección

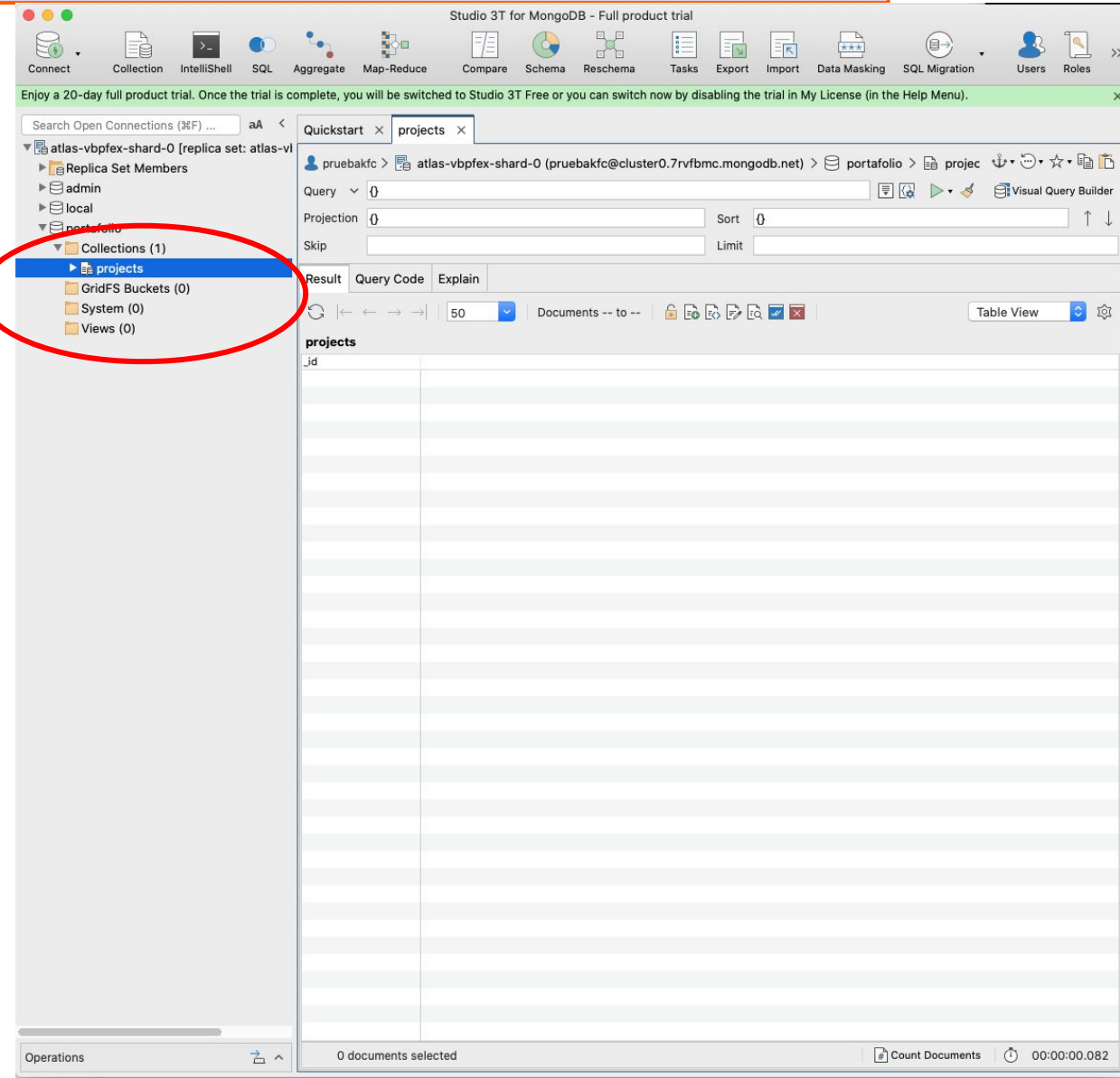




# 1. Crear base de Datos dentro de MongoDB con Studio 3T

## a. Crear una colección

Colección creada

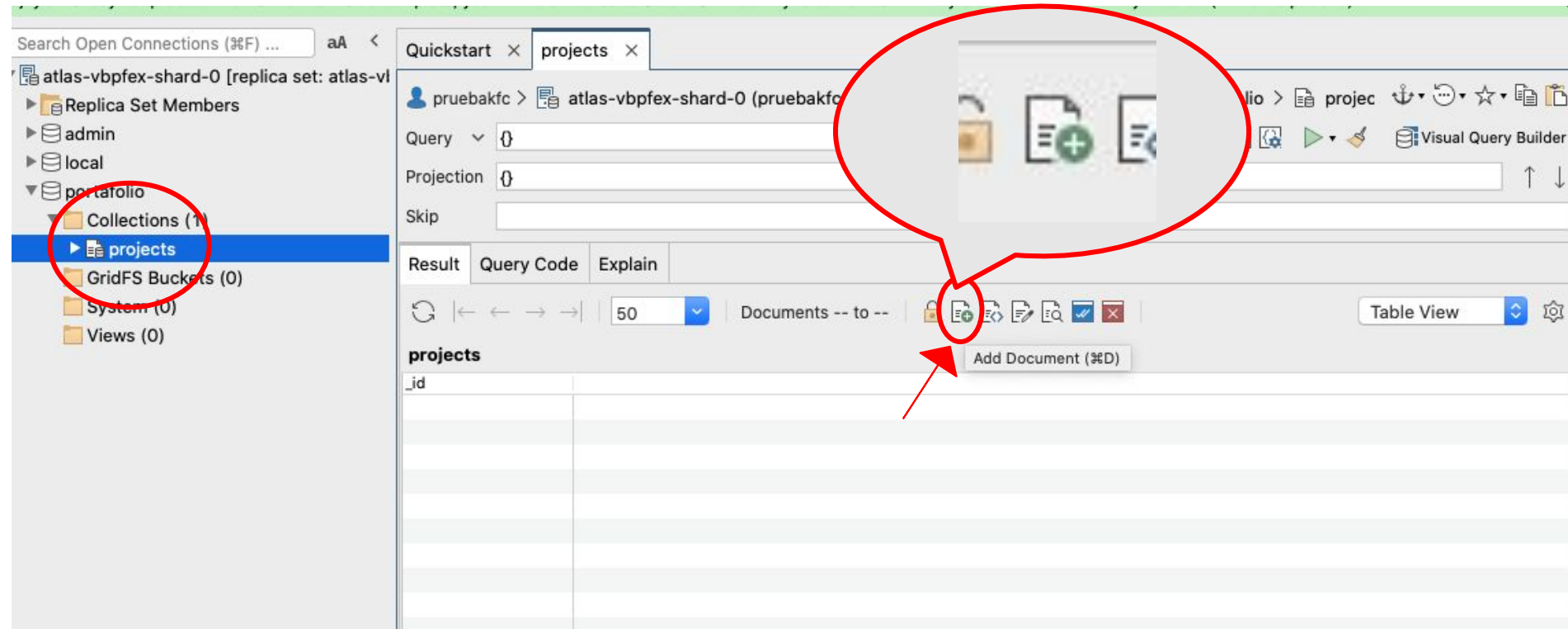






## 1. Crear base de Datos dentro de MongoDB con Studio 3T

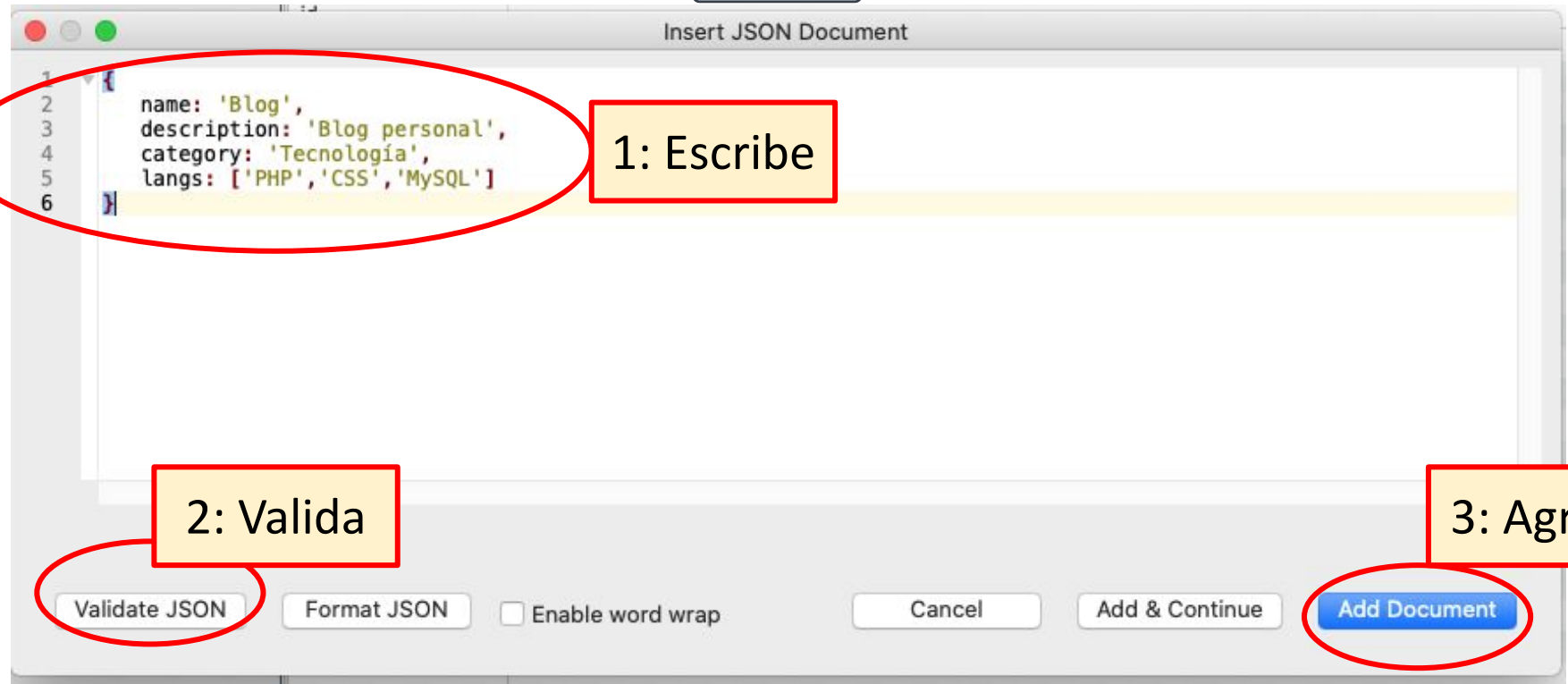
### a. Crear un documento





## 1. Crear base de Datos dentro de MongoDB con Studio 3T

### a. Crear un documento

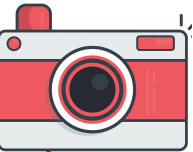




## 2. Conectar MongoDB con NodeJS y con nuestro proyecto NodeJS

Escribe el siguiente código en el archivo index.js

Observa en la línea 5, debes agregar la URL de tu base de datos (recuerda cambiar la palabra **<password>** por tu password).



```
1
2 const mongodb = require("mongodb");
3
4
5 const connectionURL = "mongodb+srv://pruebakfc:<password>@cluster0.7rvfbmc.mongodb.net/?retryWrites=true&w=majority"
6 const dbName = "portafolio"
7
8 //get MongoClient
9 const MongoClient = mongodb.MongoClient;
10
11 let db = null;
12
13 MongoClient.connect(connectionURL,{
14   useNewUrlParser: true,
15   useUnifiedTopology: true
16 },(err,connectedClient) => {
17   if(err){
18     throw err;
19   }
20   //connectedClient will be the connected instance of MongoClient
21   db = connectedClient.db(dbName);
22   //now you can write queries
23
24   db.collection("projects").find({}).toArray()
25   .then(r => {
26     console.log(r);
27   }).catch(e => {
28     console.error(`ERROR:`,e);
29   })
30
31 })
```



## 2. Conectar MongoDB con NodeJS y con nuestro proyecto NodeJS

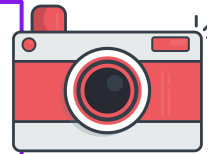
Para ejecutar el script y ver si mi NodeJS se conecta con MongoDB voy a utilizar el comando npm start.

```
misskaren@iMac-de-Karenina-6 Backend % npm start

> api-rest@1.0.0 start
> nodemon index.js

[nodemon] 2.0.20
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node index.js`

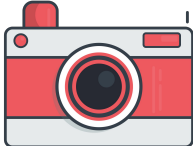
[
  {
    _id: new ObjectId("6362b292a1fc640b4ac603d7"),
    name: 'Blog',
    description: 'Blog personal',
    category: 'Tecnología',
    langs: [ 'PHP', 'CSS', 'MySQL' ]
  }
]
^C
misskaren@iMac-de-Karenina-6 Backend %
```





### 3. Crear un servidor con NodeJS y express (sistema de ruta y recibir peticiones HTTP)

Crea el archivo app.js

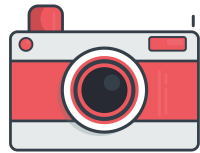


```
index.js x app.js x package.json x
Scroll Tabs
2 var express = require('express');
3 var bodyParser = require('body-parser');
4
5 var app = express();
6
7 // middlewares
8 app.use(bodyParser.urlencoded({extended: false}));
9 app.use(bodyParser.json());
10
11 //rutas
12
13 app.get('/', (req, res) => {
14   res.status(200).send(
15     "<h1>Página de inicio</h1>"
16   );
17 })
18
19
20 app.get('/test', (req, res) => {
21   res.status(200).send({
22     message: "Hola mundo desde mi API de NodeJS"
23   });
24 });
25
26
27 //exportar
28 module.exports = app;
29
```



### 3. Crear un servidor con NodeJS y express (sistema de ruta y recibir peticiones HTTP)

Realiza los siguientes cambios en tu index.js



```
index.js x app.js x package.json x
1 var app = require('./app');
2 var port = 3700;
3 const mongodb = require("mongodb");
4
5
6 const connectionURL = "mongodb+srv://pruebakfc:asdf123@cluster0.7rvfbmc.mongodb.net/?retryWrites=true&w=majority"
7 const dbName = "portafolio"
8
9 //get MongoClient
10 const MongoClient = mongodb.MongoClient;
11
12 let db = null;
13
14 MongoClient.connect(connectionURL,{
15   useNewUrlParser: true,
16   useUnifiedTopology: true
17 },(err,connectedClient) => {
18   if(err){
19     throw err;
20   }
21   //connectedClient will be the connected instance of MongoClient
22   db = connectedClient.db(dbName);
23   //now you can write queries
24
25   db.collection("projects").find({}).toArray()
26   .then(r => {
27     console.log(r);
28     app.listen(port, () => {
29       console.log("servidor corriendo correctamente en la url: localhost:3700");
30     });
31   }).catch(e => {
32     console.error(`ERROR:`,e);
33   })
34
35 })
```





### 3. Crear un servidor con NodeJS y express (sistema de ruta y recibir peticiones HTTP)

Resultado en la consola o terminal



```
misskaren@iMac-de-Karenina-6 Backend % npm start

> api-rest@1.0.0 start
> nodemon index.js

[nodemon] 2.0.20
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node index.js`
[
  {
    _id: new ObjectId("6362b292a1fc640b4ac603d7"),
    name: 'Blog',
    description: 'Blog personal',
    category: 'Tecnología',
    langs: [ 'PHP', 'CSS', 'MySQL' ]
  }
]
servidor corriendo correctamente en la url: localhost:3700
```

Resultado en el navegador ejecutando localhost:3700



Resultado en el navegador ejecutando localhost:3700/test

