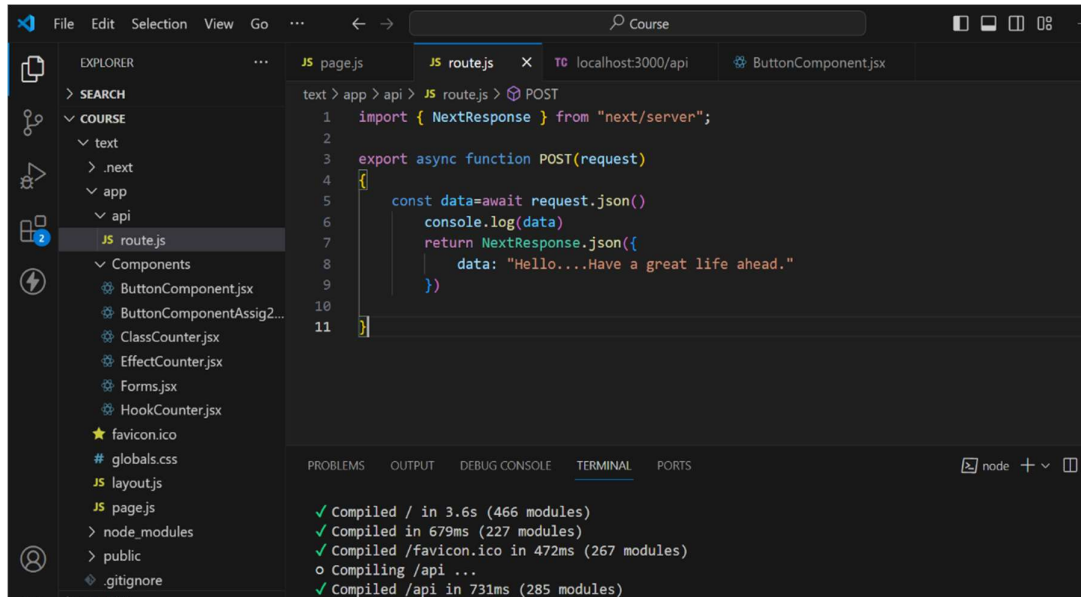


## ASSIGNMENT 3- A simple RESTFUL API using React.js, DataBase Creation, and Creating Index

RestFul API code:



The screenshot shows a VS Code editor with a file explorer on the left and a terminal at the bottom. The file explorer shows a project structure with a 'course' directory containing 'text', 'api', and 'components'. The 'api' directory contains 'route.js'. The 'route.js' file is open in the editor, showing a POST endpoint that returns a JSON response. The terminal shows the output of the command 'node route.js', which is 'Hello....Have a great life ahead.'.

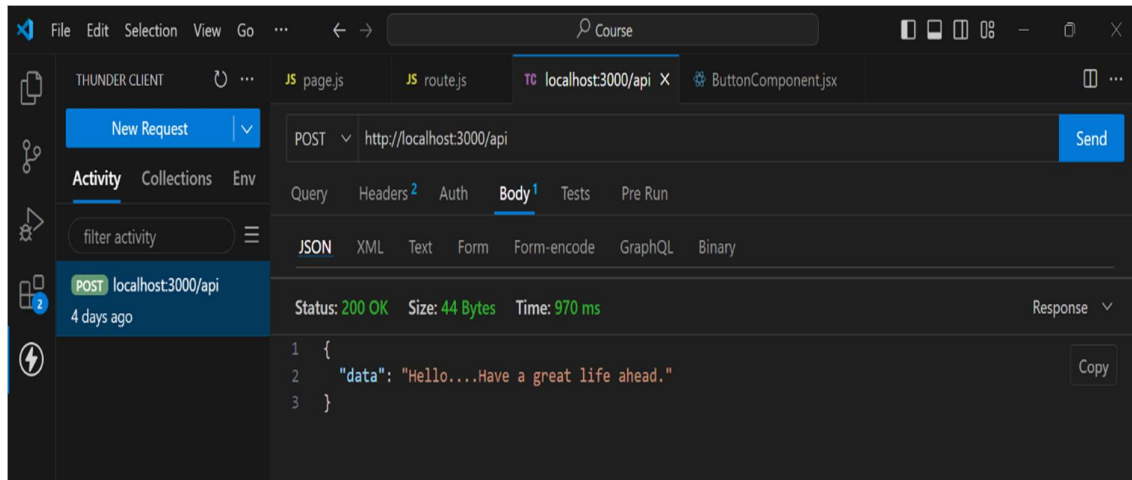
```
1 import { NextResponse } from "next/server";
2
3 export async function POST(request)
4 {
5   const data=await request.json()
6   console.log(data)
7   return NextResponse.json({
8     data: "Hello....Have a great life ahead."
9   })
10 }
11 }
```

Terminal Output:

```
✓ Compiled / in 3.6s (466 modules)
✓ Compiled in 679ms (227 modules)
✓ Compiled /favicon.ico in 472ms (267 modules)
o Compiling /api ...
✓ Compiled /api in 731ms (285 modules)
```

## Rest Output on ThunderClient

This is the output when we create an API using REST.



The screenshot shows the ThunderClient interface. A POST request to 'http://localhost:3000/api' has been sent. The response is a JSON object with a 'data' property containing the string 'Hello....Have a great life ahead.'.

Request Details:

- Method: POST
- URL: http://localhost:3000/api

Response Details:

- Status: 200 OK
- Size: 44 Bytes
- Time: 970 ms

Response Body:

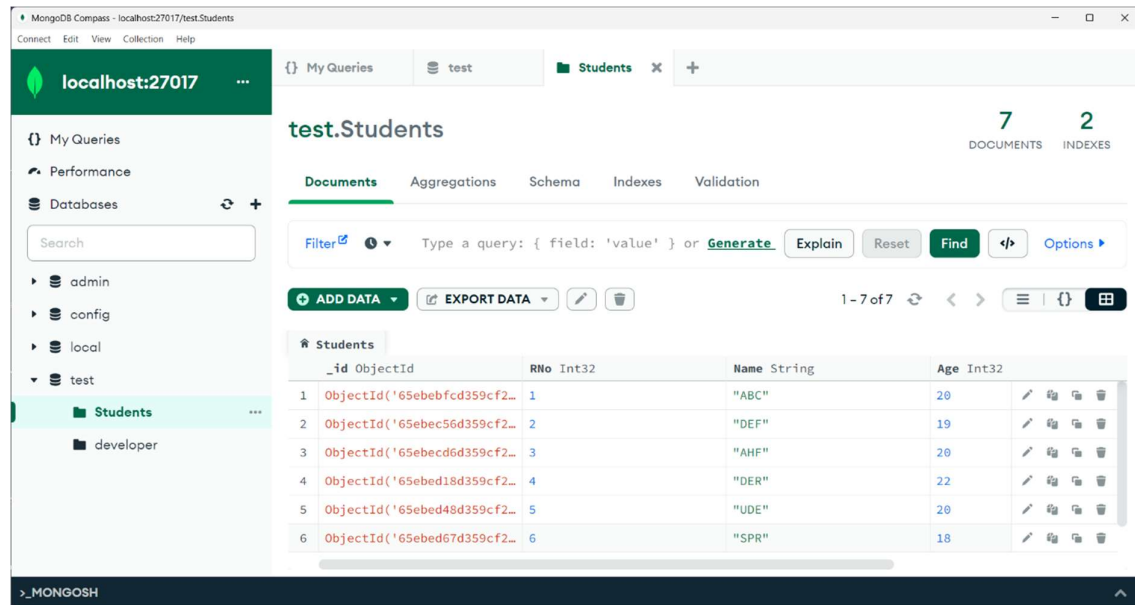
```
1 {
2   "data": "Hello....Have a great life ahead."
3 }
```

# DataBase Creation

1<sup>st</sup> we created a database named test

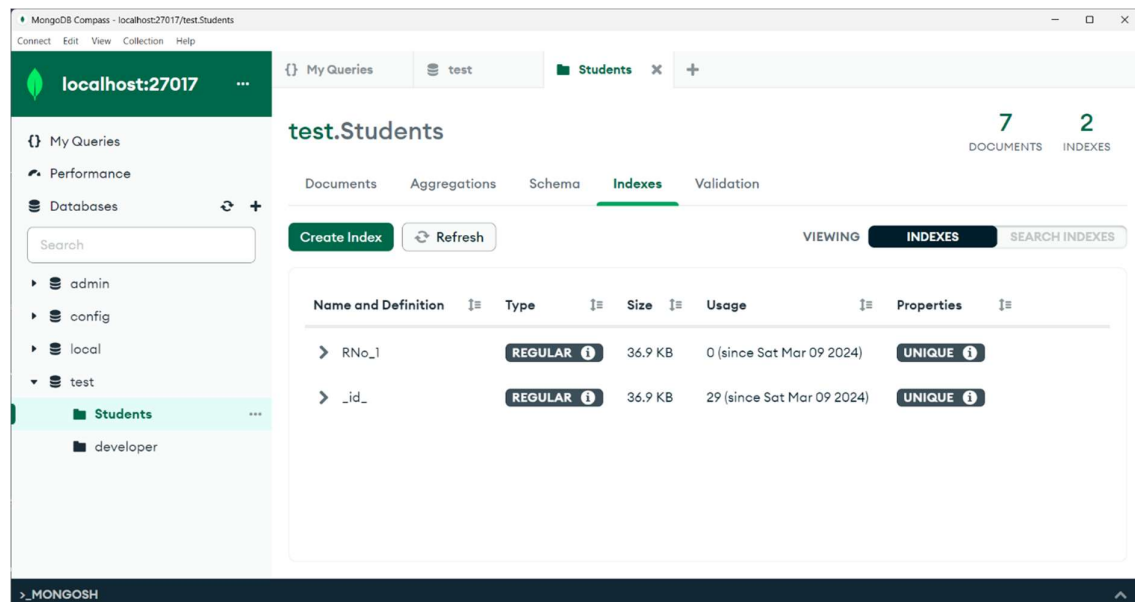
In that test we created a collection named Students

And finally by using ADD DATA , we added data of the students like rno, name, age.



# Indexing Creation

As rno is the unique attribute through which we identify a student uniquely. So, here I created an index based on Rno. And assigned unique key to it.



# Indexing usage

As we gave Rno as unique, If we try to insert a duplicate document that is existed it shows an error like..duplicate collection.

localhost:27017

My Queries

Performance

Databases

admin

config

local

test

Students

developer

My Queries

test

Students

test.Students

7 DOCUMENTS2 INDEXES

Documents

Aggregations

Schema

Indexes

Validation

Filter

Type a query: { field: 'value' } or

Generate

Explain

Reset

Find

Options

ADD DATA

EXPORT DATA

1 - 7 of 7

Students

	_id	ObjectId	RNo	Int32	Name	String	Age	Int32
3	ObjectId('65e8e6d6d359cf2...	3			"AHF"		20	
4	ObjectId('65e8ed18d359cf2...	4			"DER"		22	
5	ObjectId('65e8ed48d359cf2...	5			"UDE"		20	
6	ObjectId('65e8ed67d359cf2...	6			"SPR"		18	
7	ObjectId('65e8ee2ad359cf2...	6			No field		No field	

Plan executor error during findAndModify :: caused by :: E11000 duplicate key error coll... CANCEL UPDATE

>\_MONGOSH