21F-9519

# Task 2: -

## Server.mjs:

import config from "config";

import express from "express";

import cors from "cors";

import db from "./db.mjs";

const app = express();

const port = config.get("port");

app.use(cors(

    {

        origin:'\*'

    }

));

app.use(express.json());

app.get("/listingsAndReviews", async (req, res) => {

    let collectioncol = await db.collection("listingsAndReviews");

    try {

        const listings = await collectioncol.find({}).limit(10).toArray();

        if (!listings || listings.length === 0) {

            throw new Error("No data found.");

        }

        return res.status(200).json(listings);

    } catch (e) {

        console.error(`Error: ${e}`);

        return res.status(404).send('Error');

    }

});

app.post("/listingsAndReviews", async (req, res) => {

    let collectioncol = await db.collection("listingsAndReviews");

    try {

        const newListing = await collectioncol.findOne({ \_id: 12 });

        return res.status(201).json(newListing);

    } catch (e) {

        console.error(`Error: ${e}`);

        return res.status(500).send('Error');

    }

});

app.put("/listingsAndReviews/:id", async (req, res) => {

    let collectioncol = await db.collection("listingsAndReviews");

    try {

        const updatedListing = await collectioncol.findOneAndUpdate(

            { \_id: 100 },

            { $set: { name: "Rai" } }

        );

        return res.status(200).json(updatedListing.value);

    } catch (e) {

        console.error(`Error: ${e}`);

        return res.status(500).send('Error');

    }

});

app.delete("/listingsAndReviews/:id", async (req, res) => {

    let collectioncol = await db.collection("listingsAndReviews");

    try {

        const deletedListing = await collectioncol.findOneAndDelete({ \_id: 10 });

        return res.status(200).json({ message: "Listing deleted successfully" });

    } catch (e) {

        console.error(`Error: ${e}`);

        return res.status(500).send('Error');

    }

});

app.listen(port, () => {

    console.log("Server Started on port http://localhost:" + port + "/listingsAndReviews");

});

## DB.mjs: -

import config from "config";

import { MongoClient } from "mongodb";

const url = config.get("dbURI");

const client = new MongoClient(url);

let conn;

let db;

try {

    conn = await client.connect();

    console.log("Connected to MongoDB");

} catch (err) {

    console.error(`Failed to connect to the database: ${err}`);

}

db = conn.db("sample\_airbnb");

export default db;

## Default.mjs: -

{

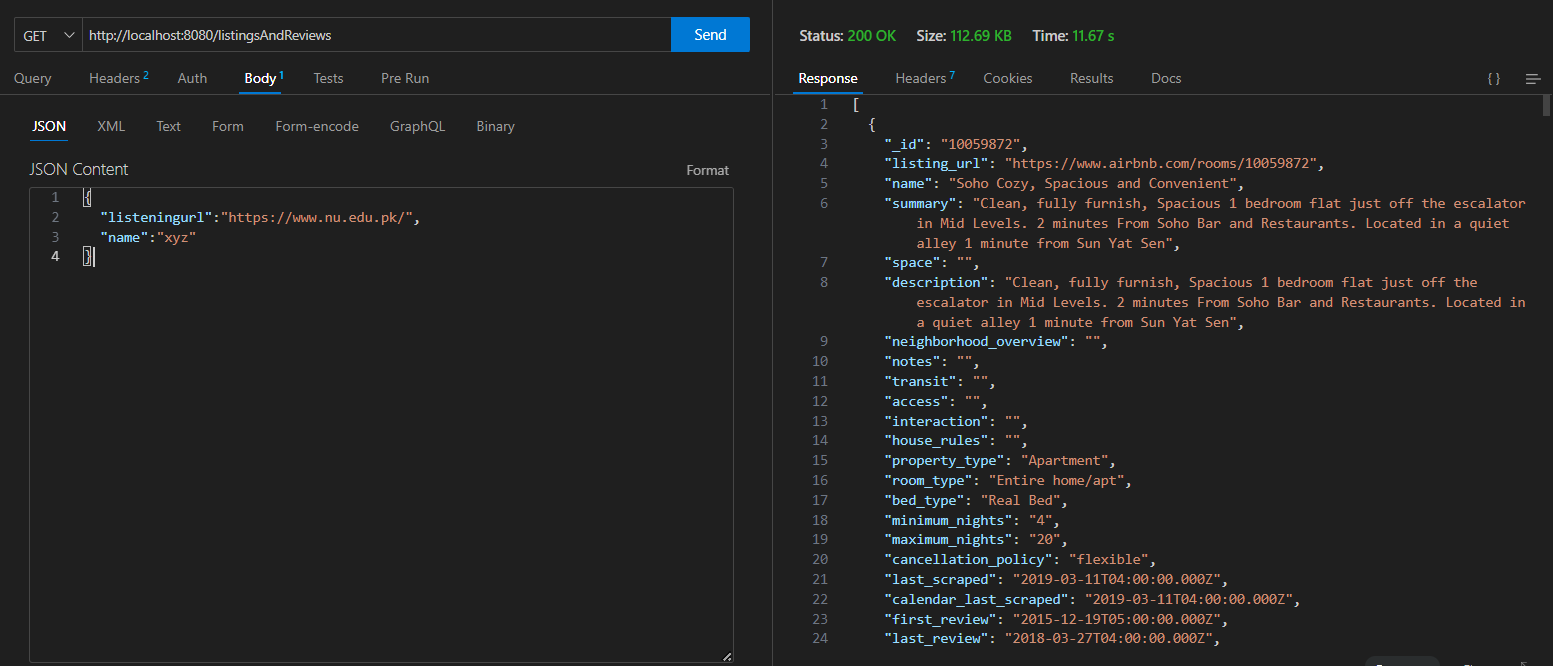
    "port":"8080",

    "dbURI":"mongodb+srv://f219519:just1me2@newcluster.rpidtcf.mongodb.net/?retryWrites=true&w=majority&appName=NewCluster"

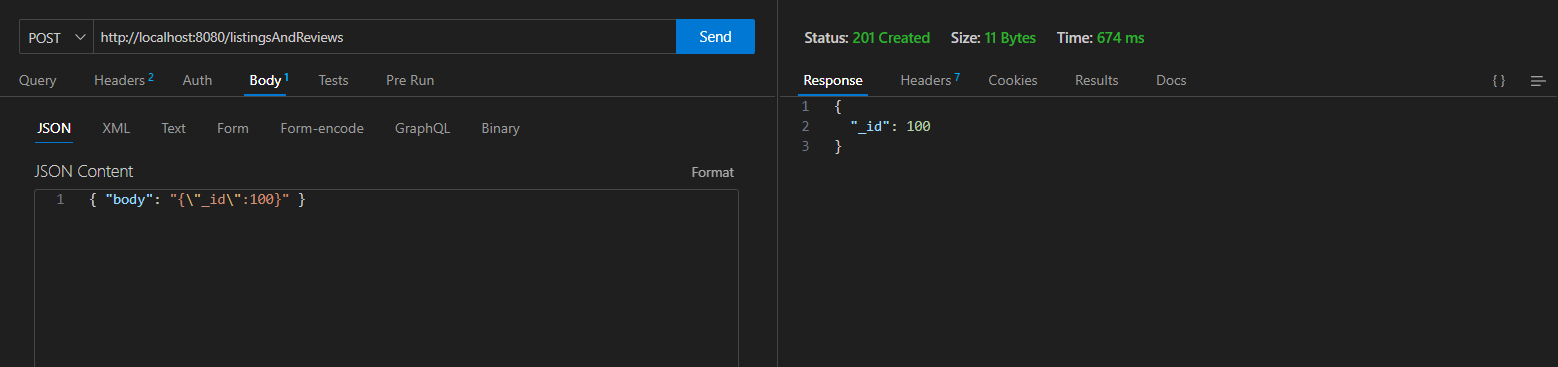
}

Output: -

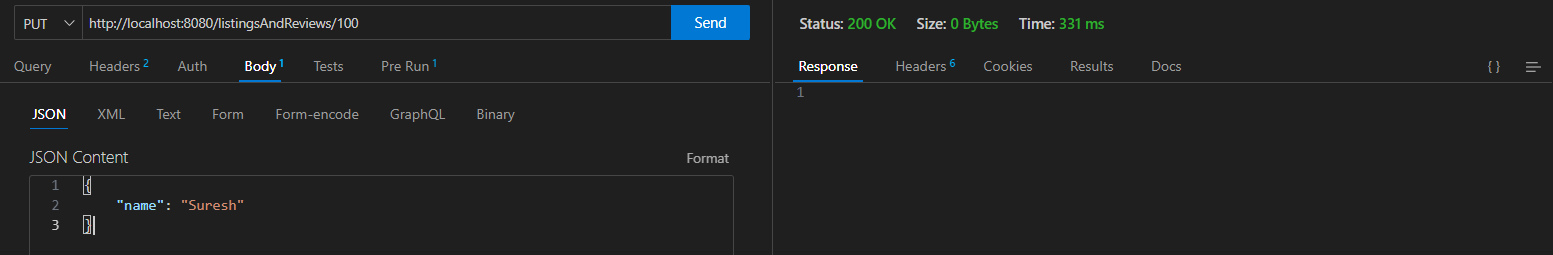
##### Get Request: -



##### POST Request: -



##### PUT Request: -



##### DELETE Request: -

