4-) Create a new DB called lab6.

Develop an Express Application for the following operations:

Find/FindOne/Add/Delete

Try to use appropriate HTTP verbs for an entity called: lectures, document structure is:

lectures = [{\_id, course, lecture}]

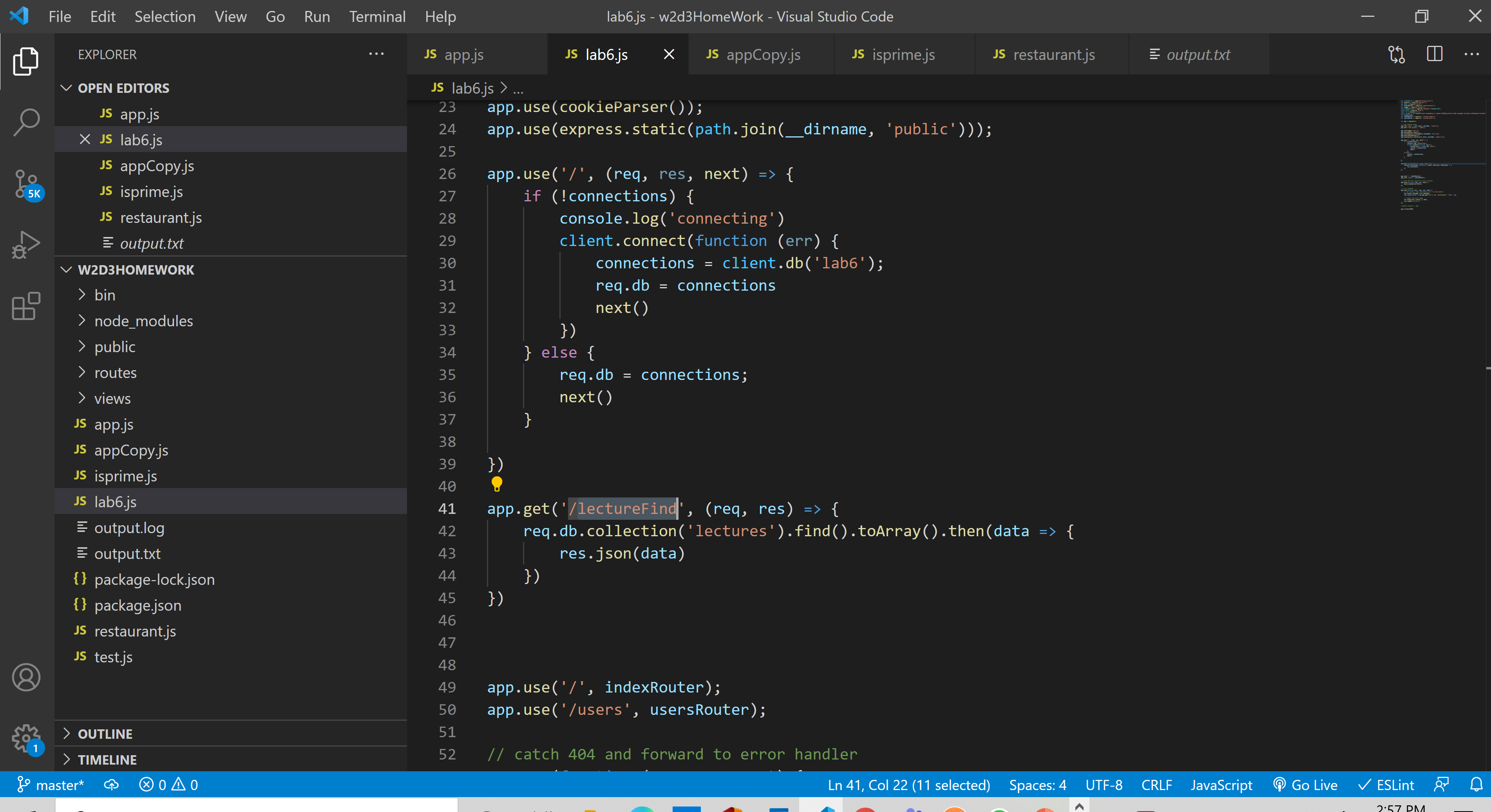
• Test your API using Postman

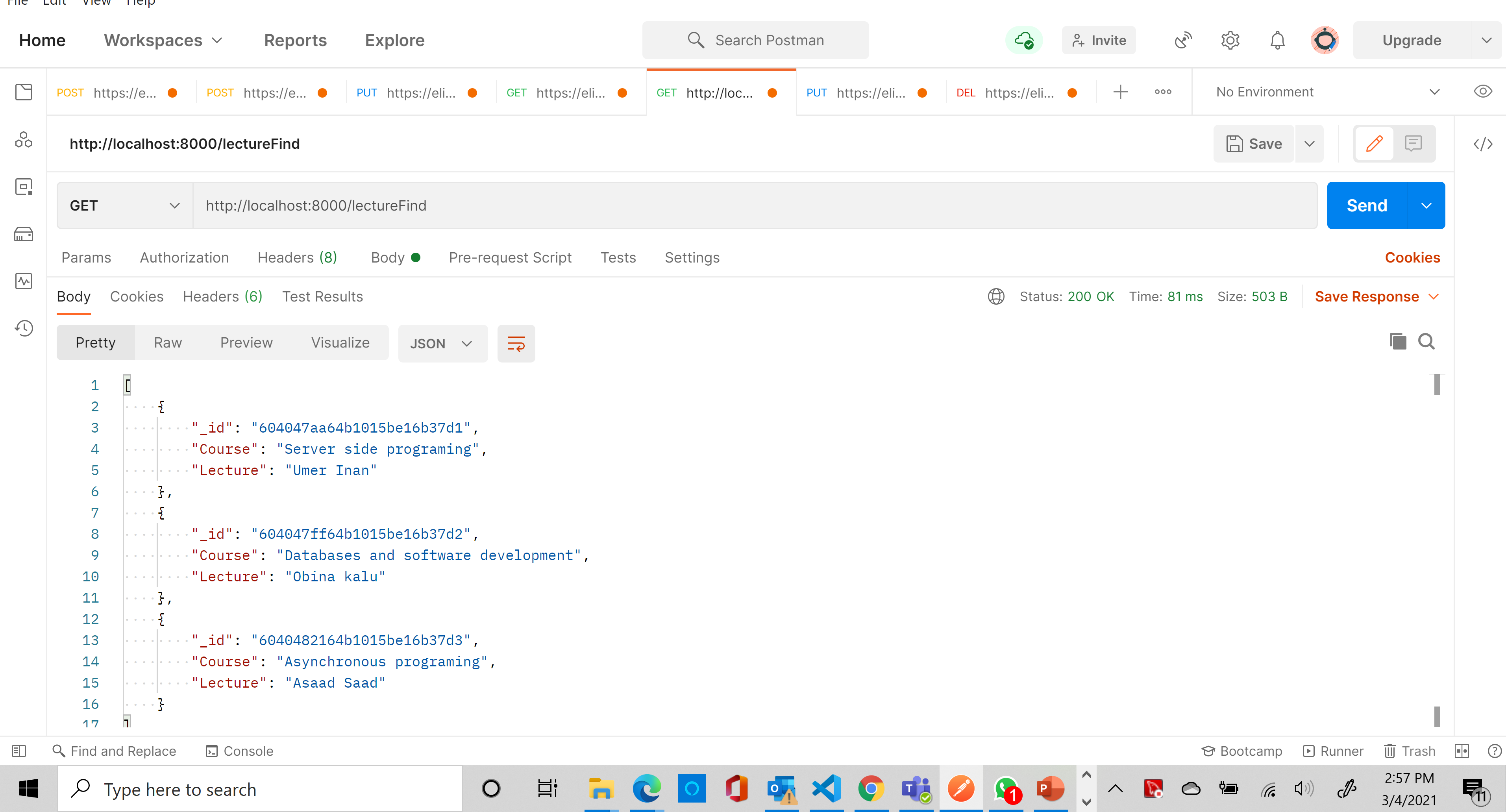
• Implement a route (GET /search/:q) to search if the lecture name contains the

passed :q parameter.

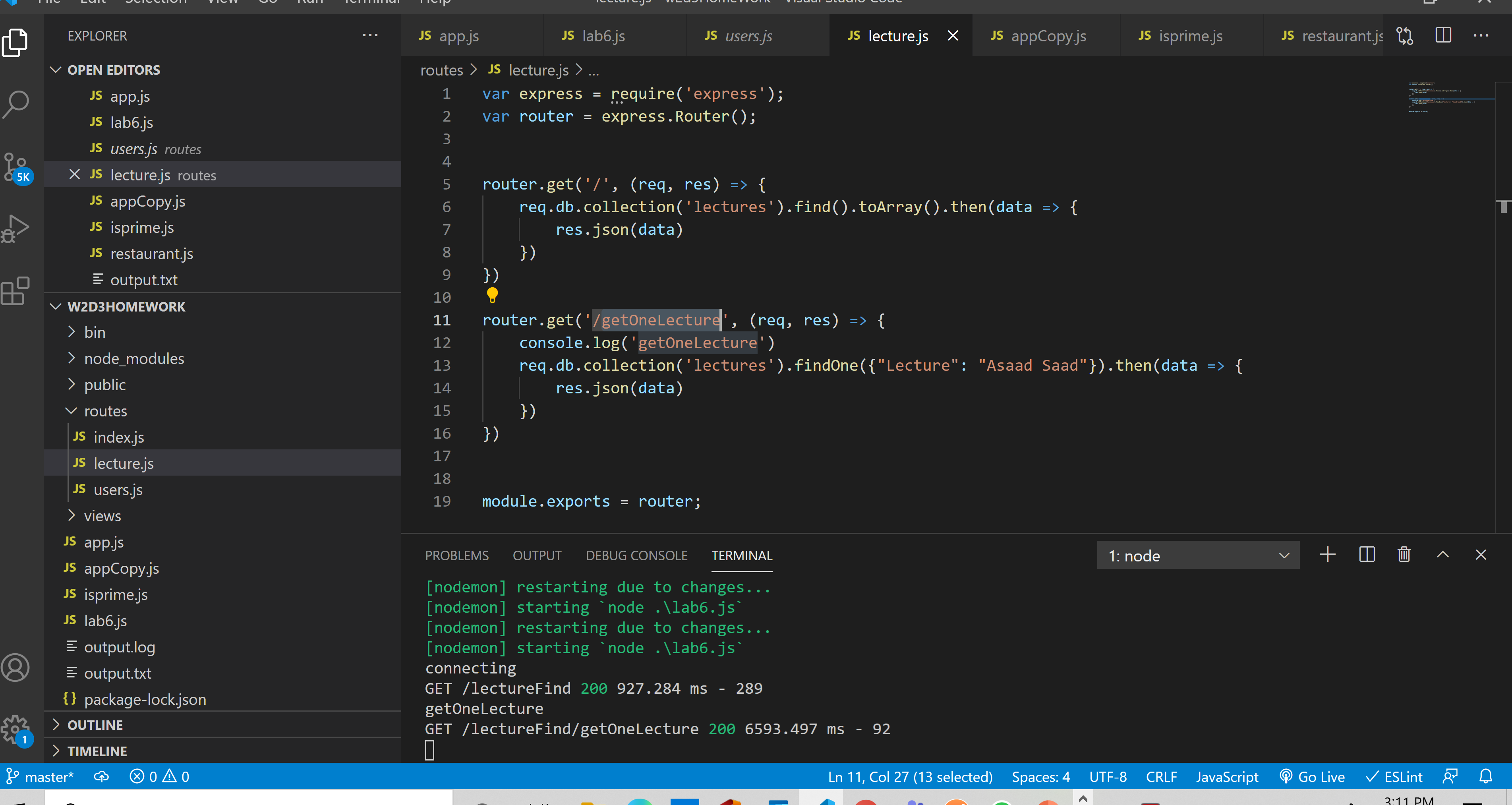
• Send the results as JSON.

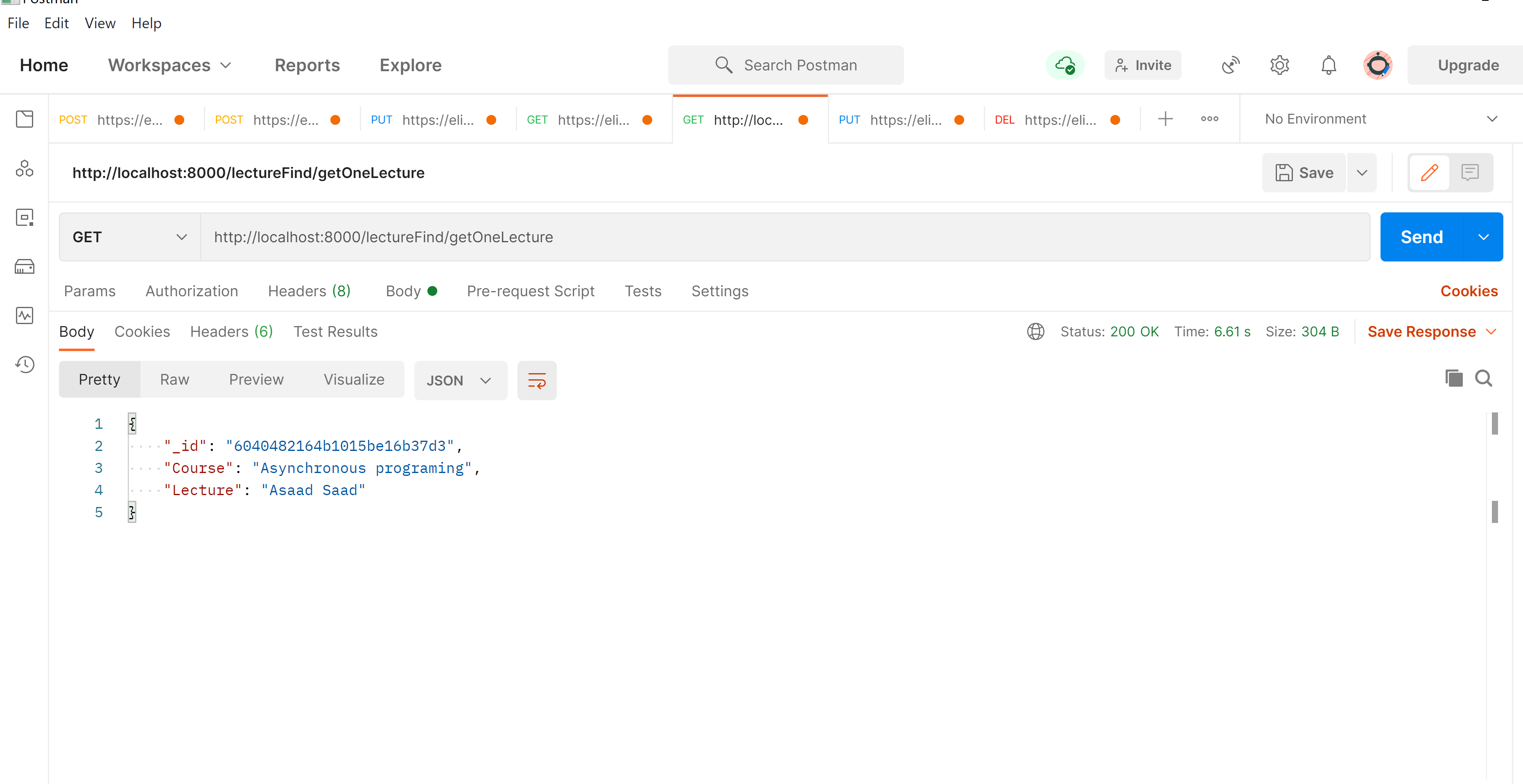
Find:



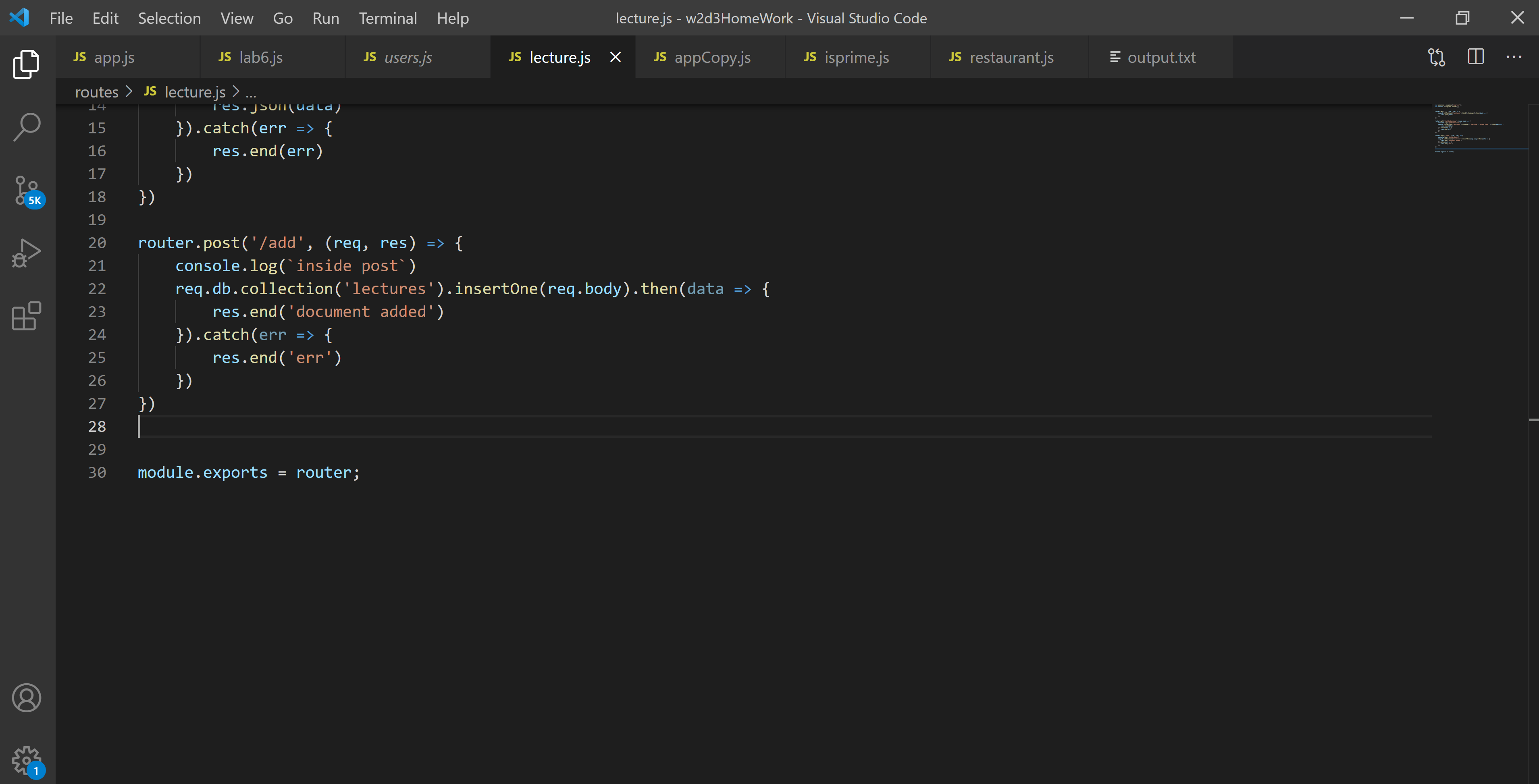


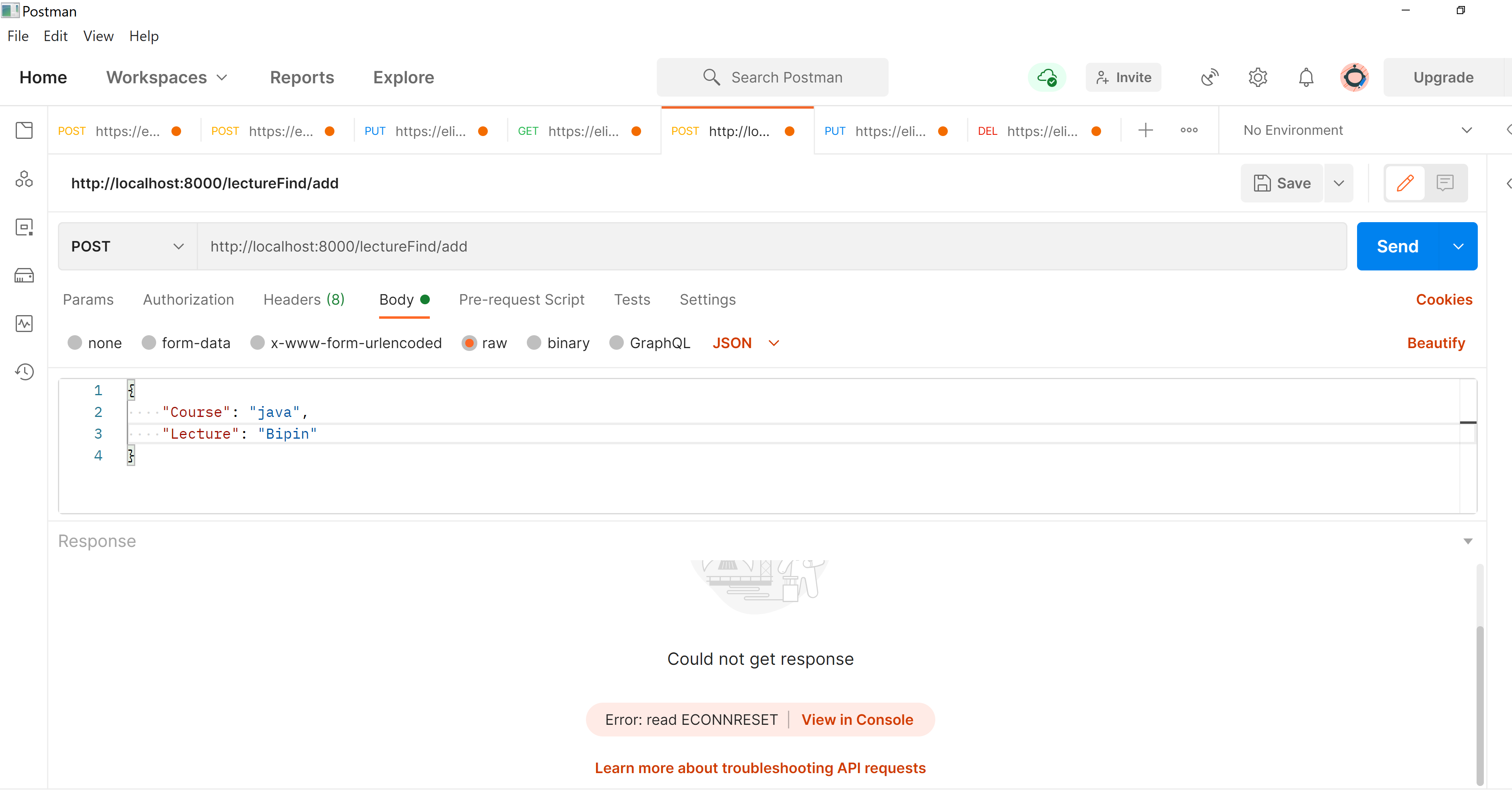
FindOne:



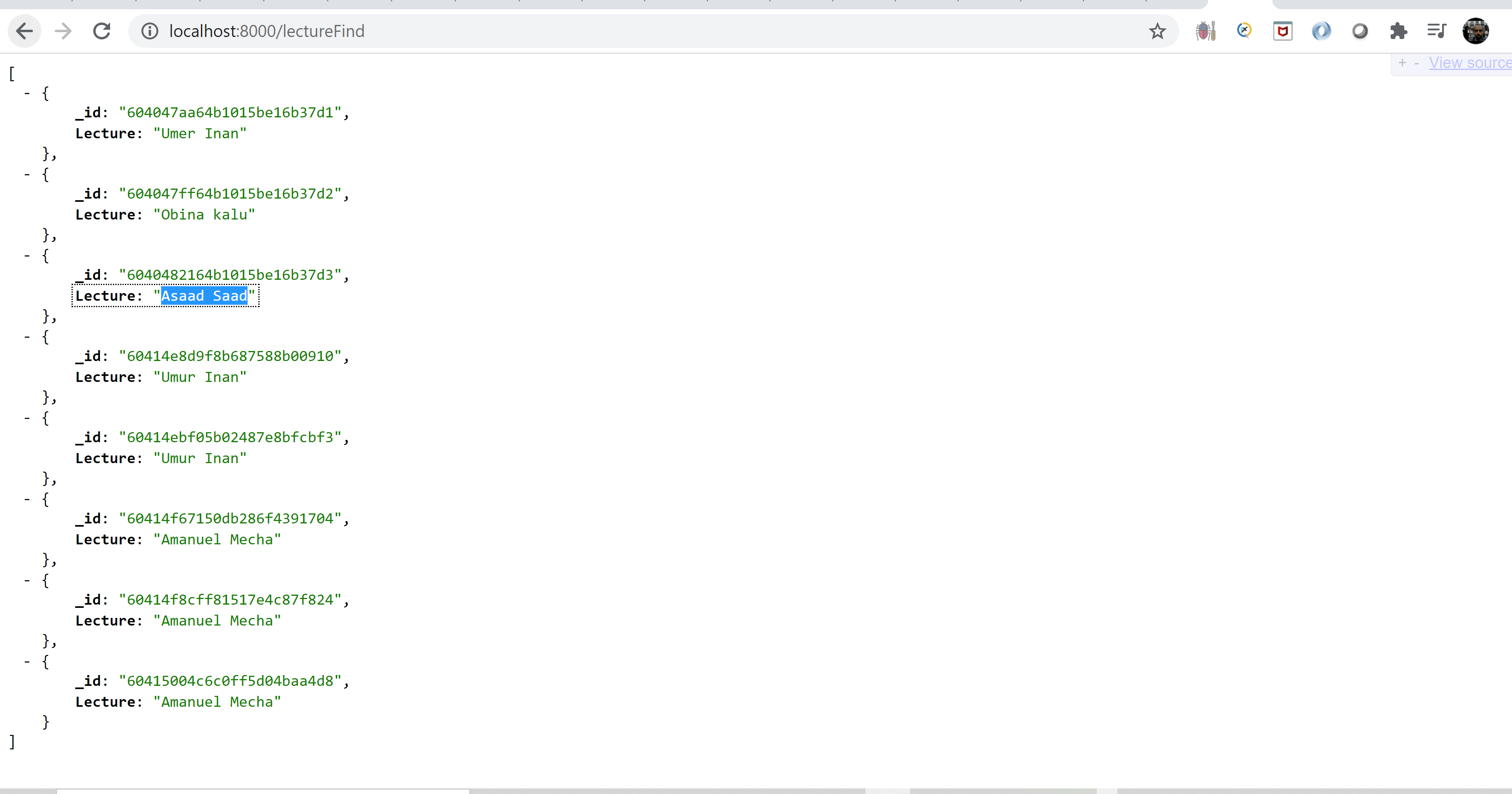


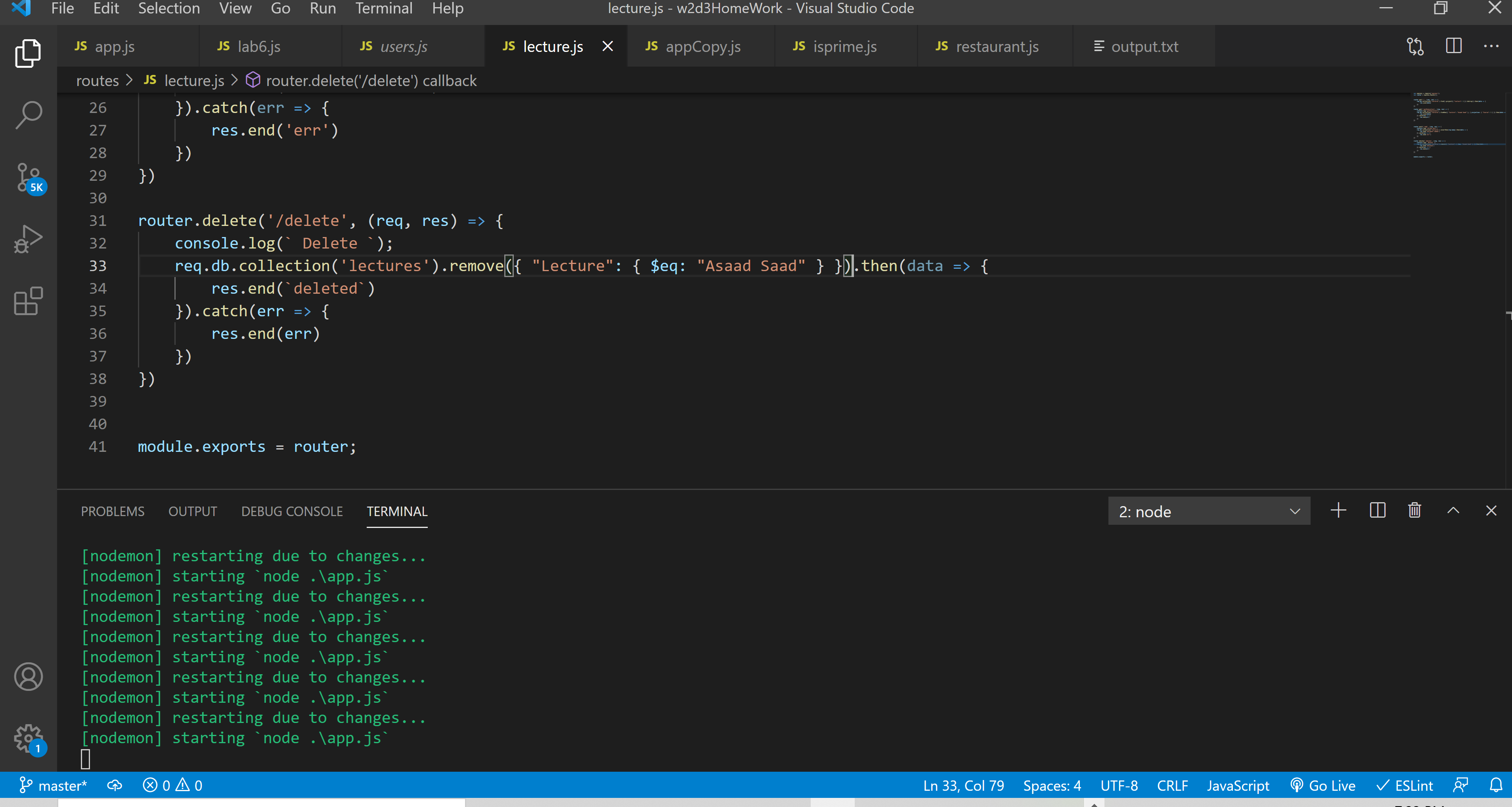
**Add:**





Delete:





5-) Considering the following restaurants collection that has information about all

restaurants in the USA. Import the data into a local/cloud DB server.

{

"address": { "building": "1007",

"coord": [ -73.856077, 40.848447 ],

"street": "Morris Park Ave",

"zipcode": "10462" },

"district": "Bronx",

"cuisine": "Bakery",

"grades": [ {"date": {"$date": 1393804800000}, "grade": "A", "score": 2},

{"date": {"$date": 1378857600000}, "grade": "A", "score": 6},

{"date": {"$date": 1358985600000}, "grade": "A", "score": 10},

{"date": {"$date": 1322006400000}, "grade": "A", "score": 9},

{"date": {"$date": 1299715200000}, "grade": "B", "score": 14}],

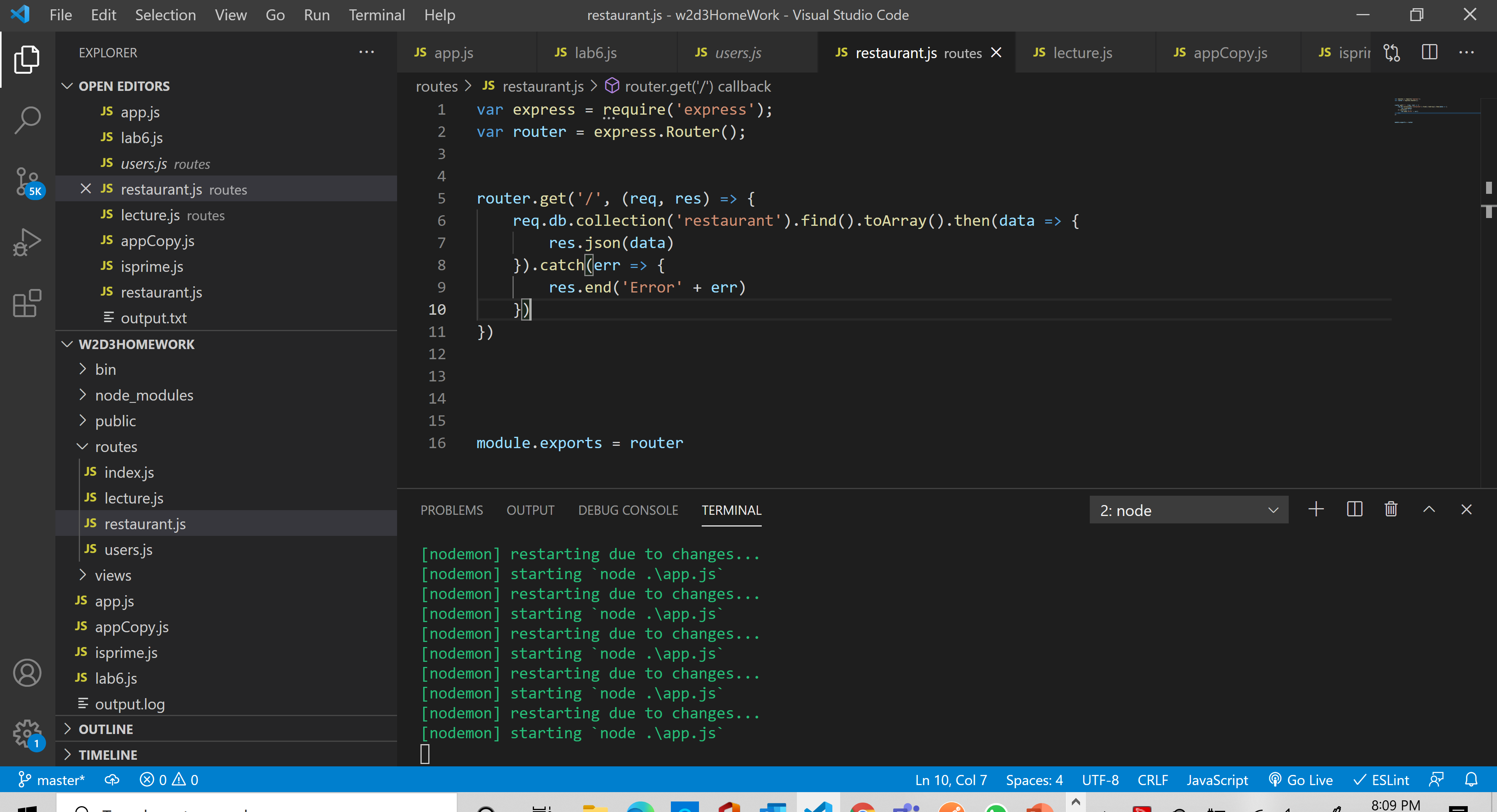
"name": "Morris Park Bake Shop",

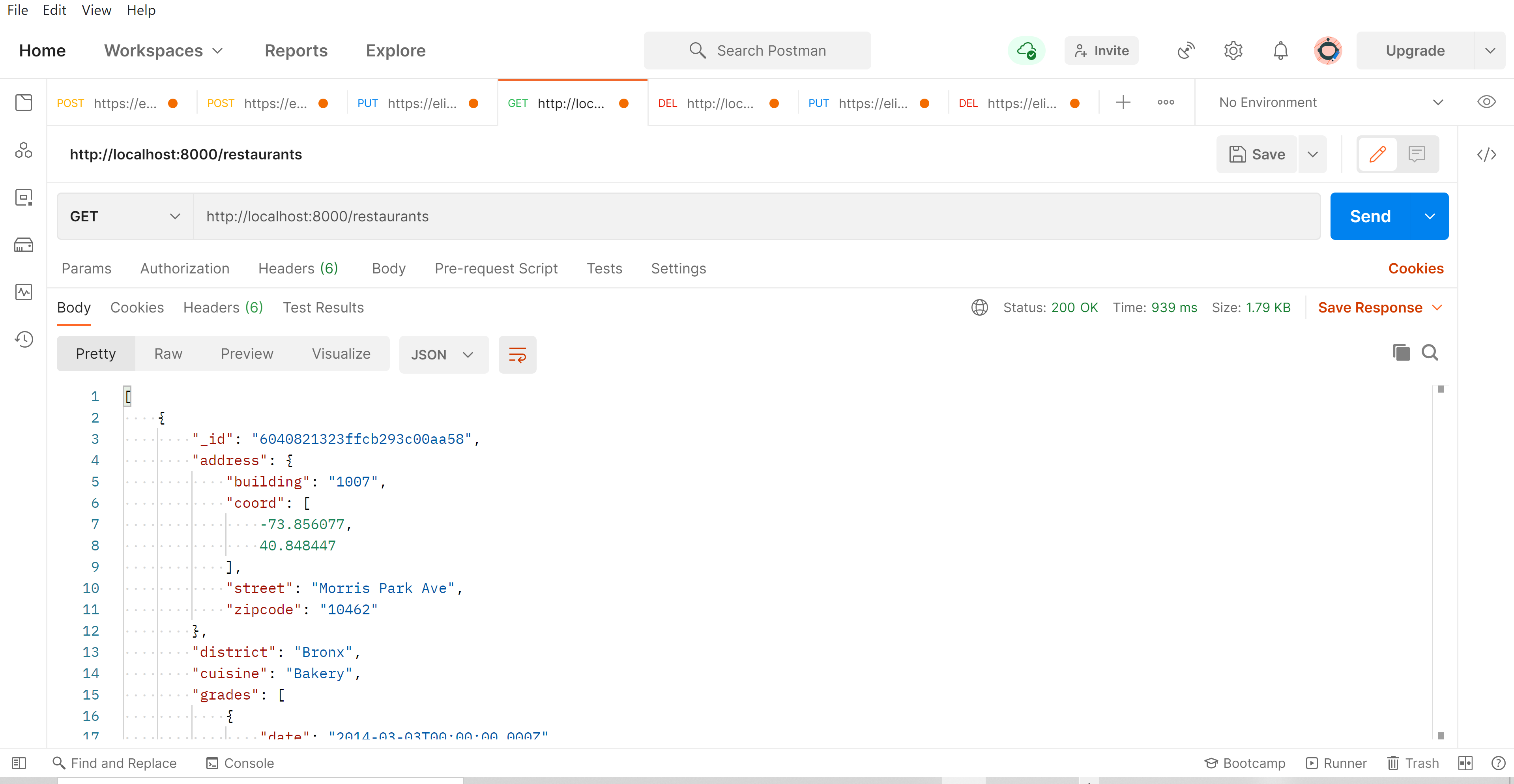
"restaurant\_id": "30075445"

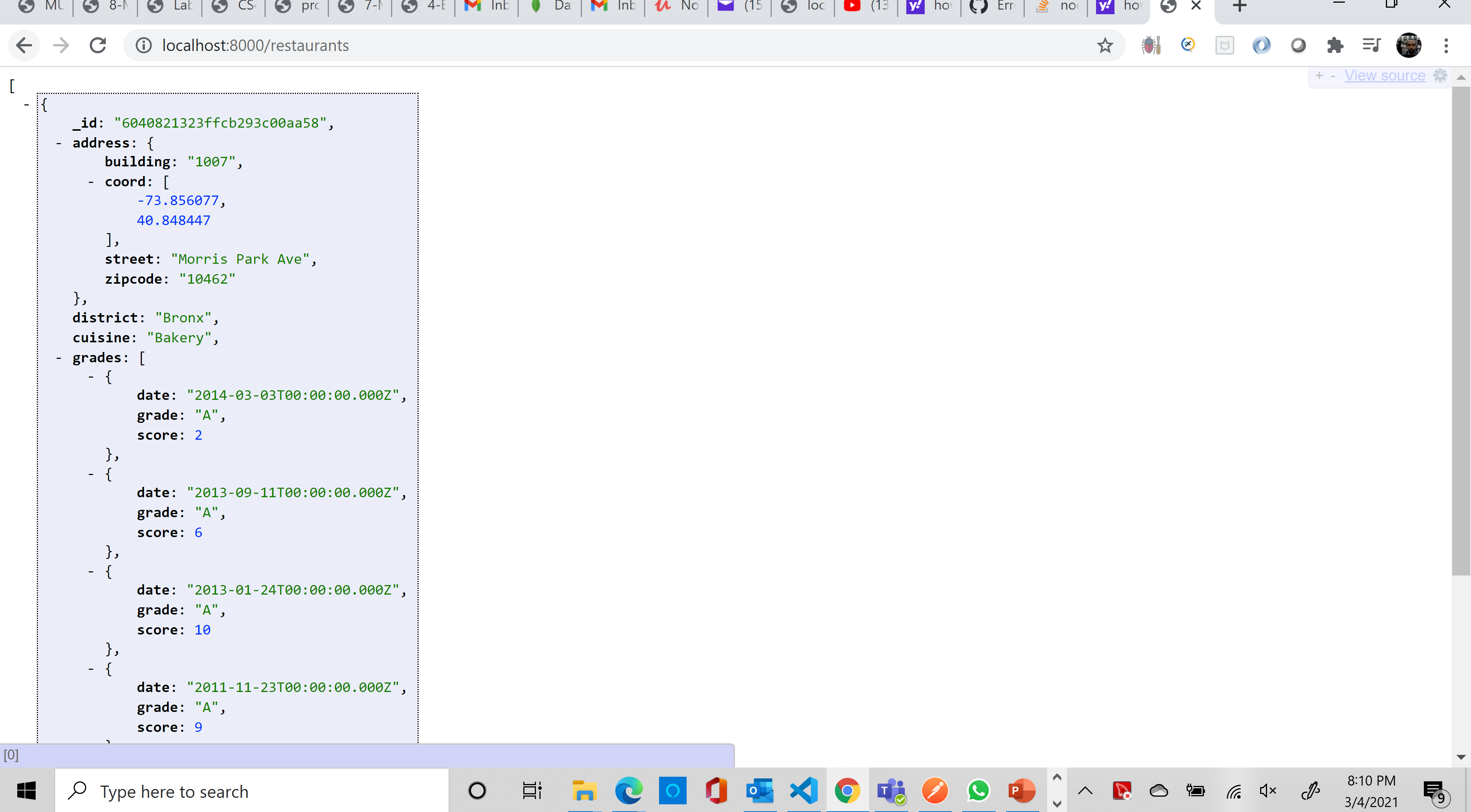
}

1. Write a MongoDB query to display all the documents in the collection

restaurants.







2. Write a MongoDB query to display the

fields restaurant\_id, name, district and cuisine for all the documents in the

collection restaurant.

router.get('/filter', (req, res) => {

    req.db.collection('restaurant').find().project({"restaurant\_id": 1 , "name": 1, "district ": 1, "cuisine": 1}).toArray().then(data => {

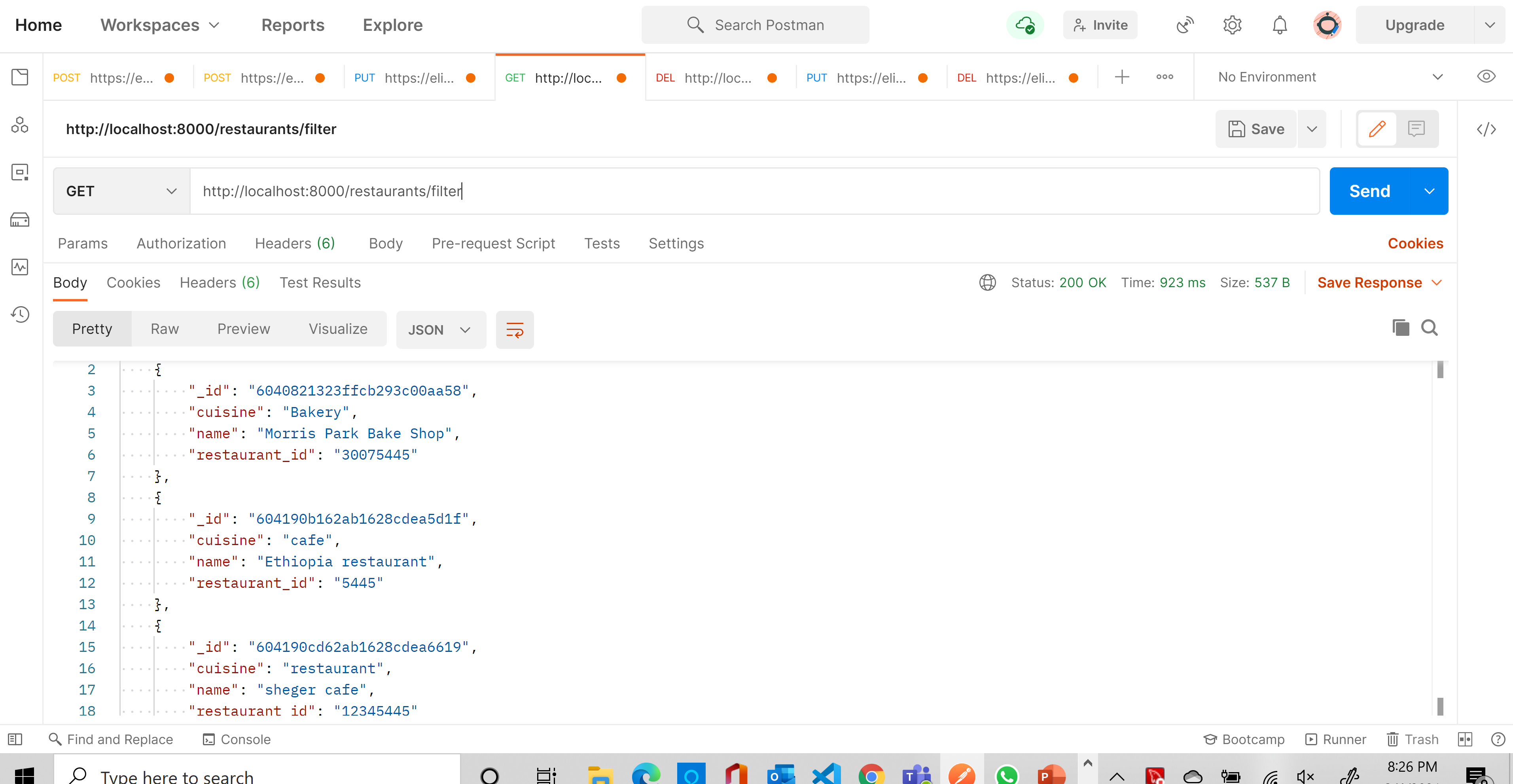
        res.json(data)

    }).catch(err => {

        res.end('Error' + err)

    })

})



3. Write a MongoDB query to display the

fields restaurant\_id, name, district and cuisine, but exclude the field \_id for all the

documents in the collection restaurant.

router.get('/filter', (req, res) => {

    req.db.collection('restaurant').find().project({ "name": 1, "district ": 1, "cuisine": 1, "restaurant\_id": 1 ,"\_id":0}).toArray().then(data => {

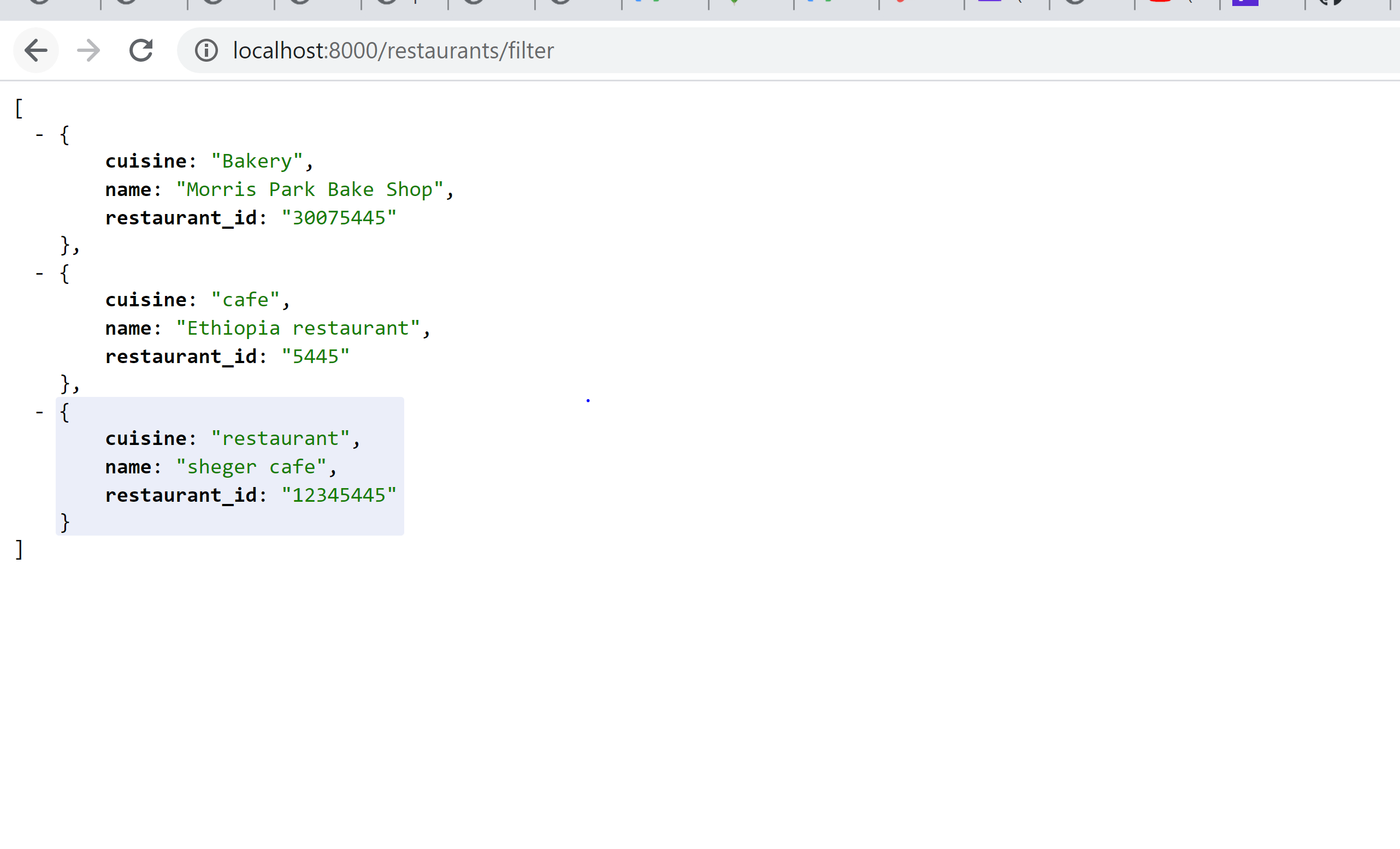
        res.json(data)

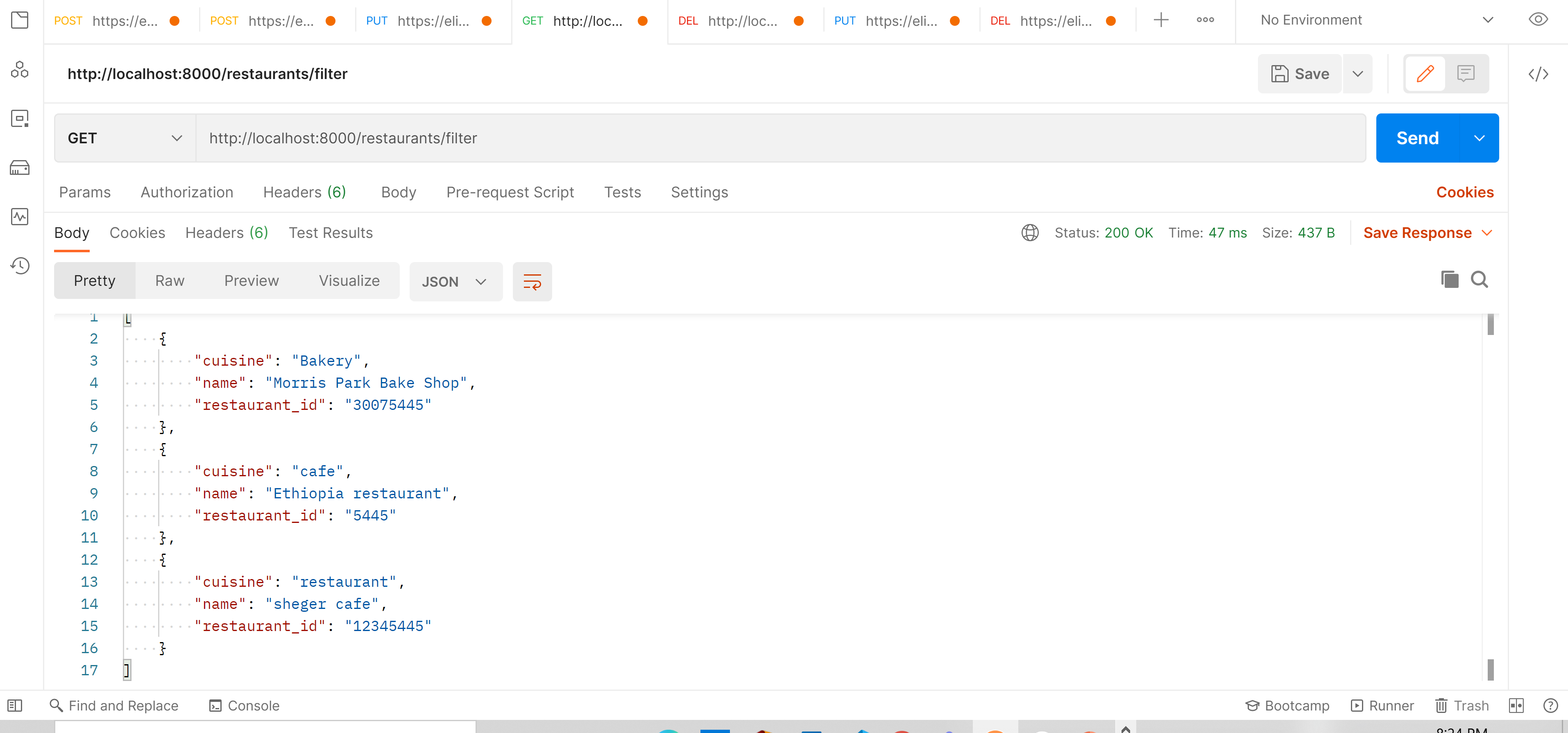
    }).catch(err => {

        res.end('Error' + err)

    })

})





4. Write a MongoDB query to display all the restaurant which is in

the district "Bronx".

router.get('/filter', (req, res) => {

    req.db.collection('restaurant').find({"district": {$eq:"Bronx"}}).toArray().then(data => {

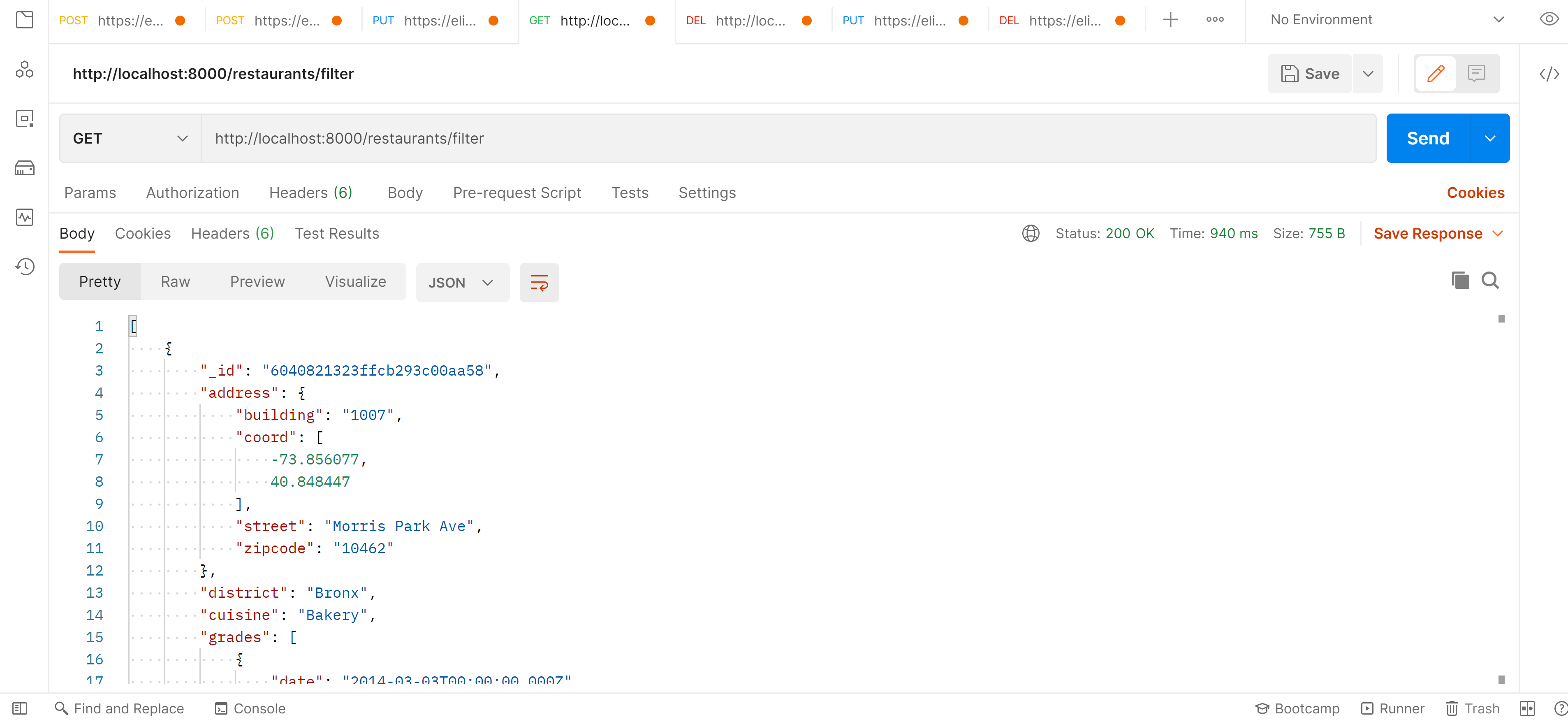
        res.json(data)

    }).catch(err => {

        res.end('Error' + err)

    })

})



5. Write a MongoDB query to display the first 5 restaurant which is in

the district "Bronx".

router.get('/limit', (req, res) => {

    req.db.collection('restaurant').find({"district": {$eq:"Bronx"}}).limit(5).toArray().then(data => {

        res.json(data)

    }).catch(err => {

        res.end('Error' + err)

    })

})

