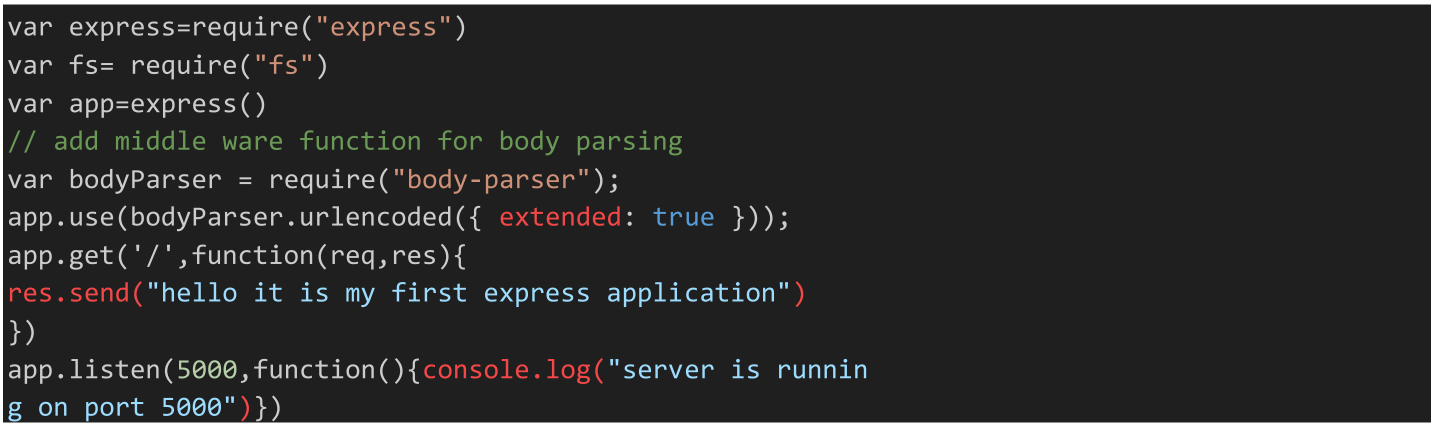
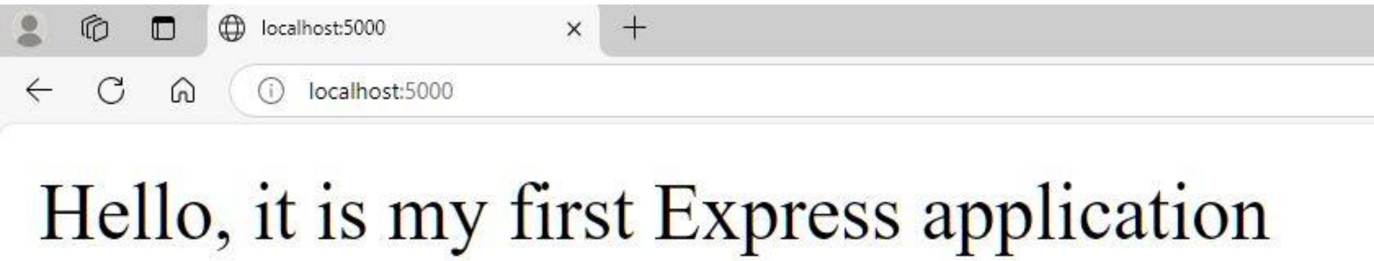
WEEK 4

Exercise 1: write the code for the basic express hello application . NodeJS Core module used: “express”



```
var express=require("express")
var fs= require("fs")
var app=express()
// add middle ware function for body parsing
var bodyParser = require("body-parser");
app.use(bodyParser.urlencoded({ extended: true }));
app.get('/',function(req,res){
res.send("hello it is my first express application")
})
app.listen(5000,function(){console.log("server is runnin
g on port 5000")})
```

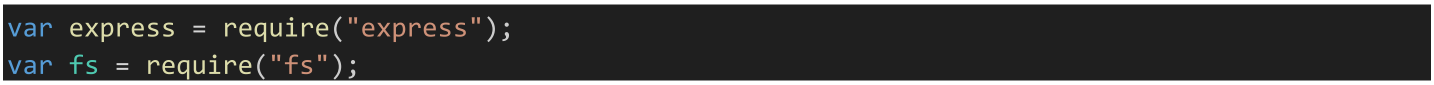
OUTPUT



Hello, it is my first Express application

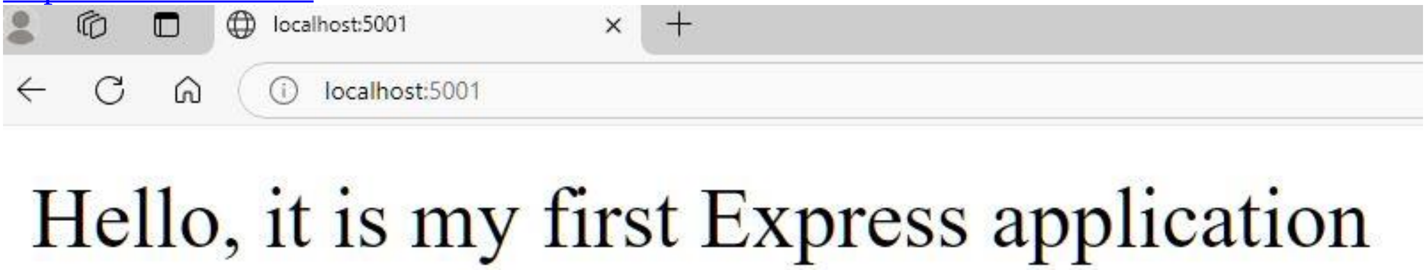
Exercise 2: Now return to Visual Studio code again, press Ctrl C to terminate the running application, and add the following code in index.js. Add two more URLs in the index.js using following code:

app.get('/about',function(req,res)
{ res.send("This is basic express application ")
CN5006 Lab Session Week4 Prepared by Dr. N.Qazi
})
app.get('/users/:userId/books/:bookId', function (req, res) {
res.send(req.params)
})



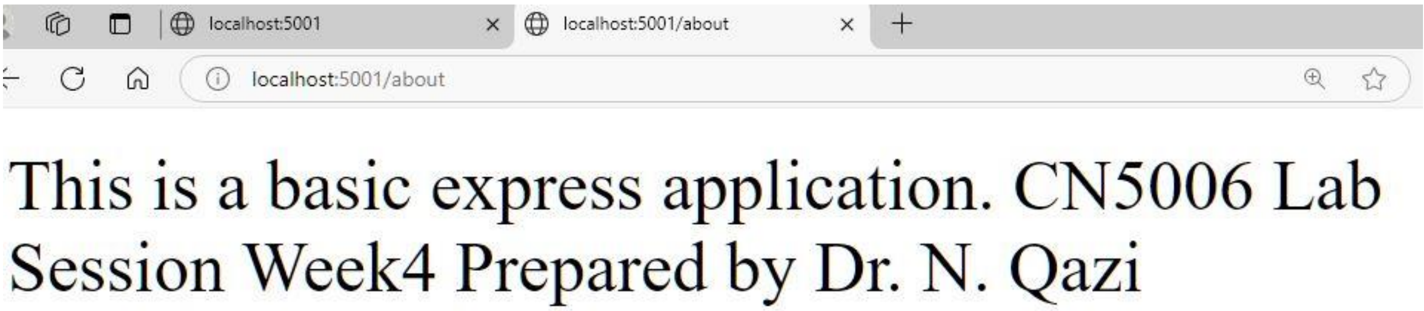
```
var express = require("express");
var fs = require("fs");
var app = express();

// Add middleware function for body parsing
var bodyParser = require("body-parser");
app.use(bodyParser.urlencoded({ extended: true }));

// Root route
app.get('/', function(req, res) {
    res.send("Hello, it is my first Express application");
});

// About route
app.get('/about', function(req, res) {
    res.send("This is a basic express application. CN5006 Lab Session Week4 Prepared by Dr.
N. Qazi");
});

// Dynamic route for user and book IDs
app.get('/users/:userId/books/:bookId', function(req, res) {
    res.send(req.params); // Send the userId and bookId parameters
});

// Function to start the server on an available port
function startServer(port) {
    app.listen(port, function() {
        console.log(`Server is running on port ${port}`);
    }).on('error', (err) => {
        if (err.code === 'EADDRINUSE') {
            console.log(`Port ${port} is in use, trying another port...`);
            startServer(port + 1);  // Try the next port
        } else {
            console.error('Server error:', err);
        }
    });
}

// Start server on port 5000, will retry if the port is in use
startServer(5000);
```

http://localhost:5001



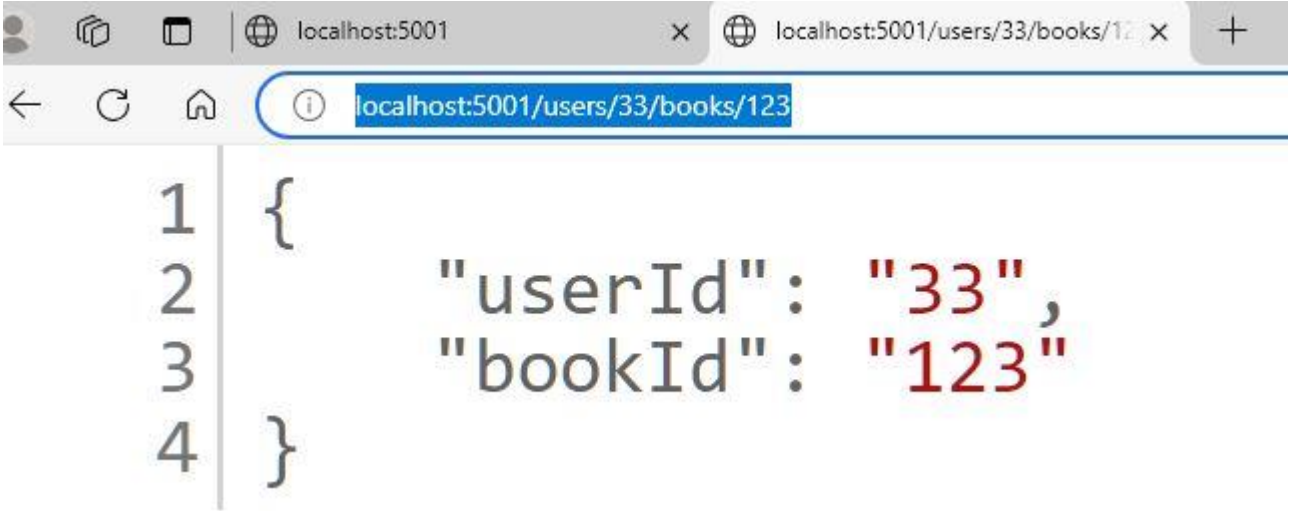
Hello, it is my first Express application

http://localhost:5001/about



This is a basic express application. CN5006 Lab Session Week4 Prepared by Dr. N. Qazi

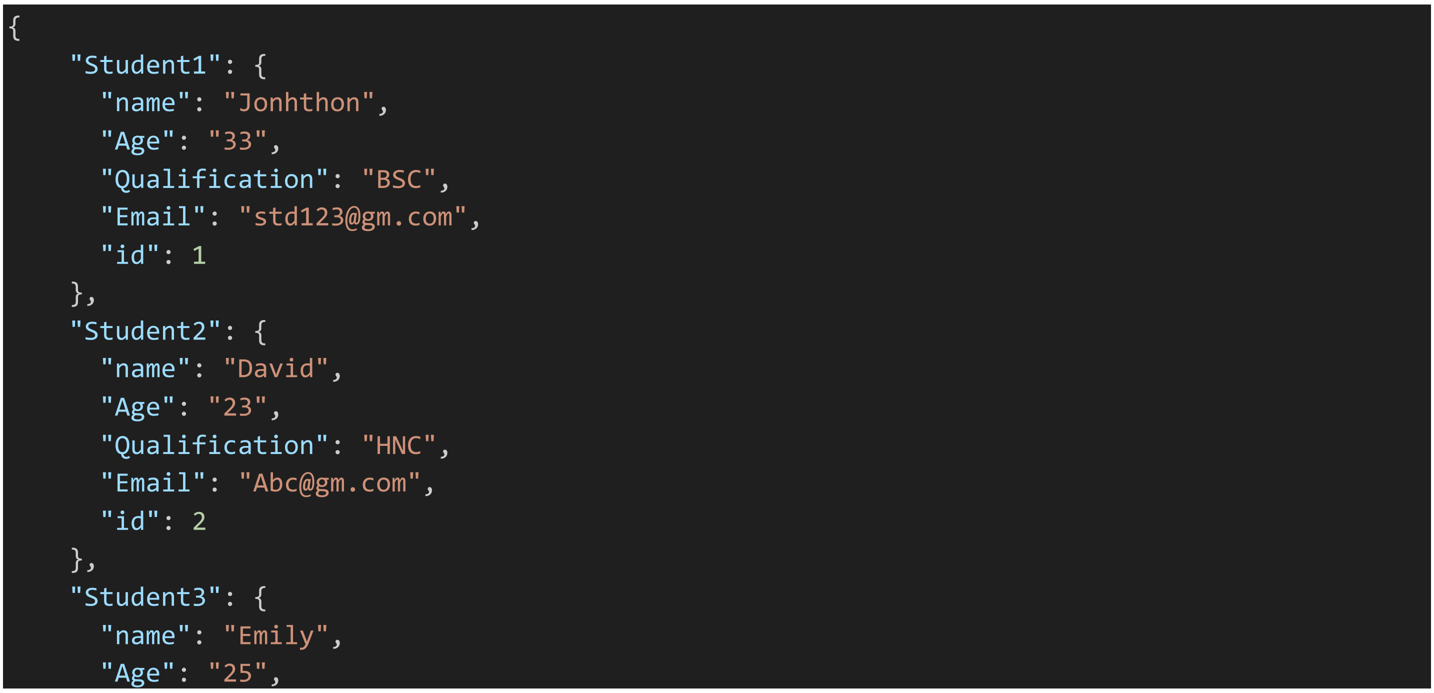
http://localhost:5001/users/33/books/123


localhost:5001
localhost:5001/users/33/books/12
localhost:5001/users/33/books/123

```
{
        "userId": "33",
        "bookId": "123"
}
```

Exercise 3

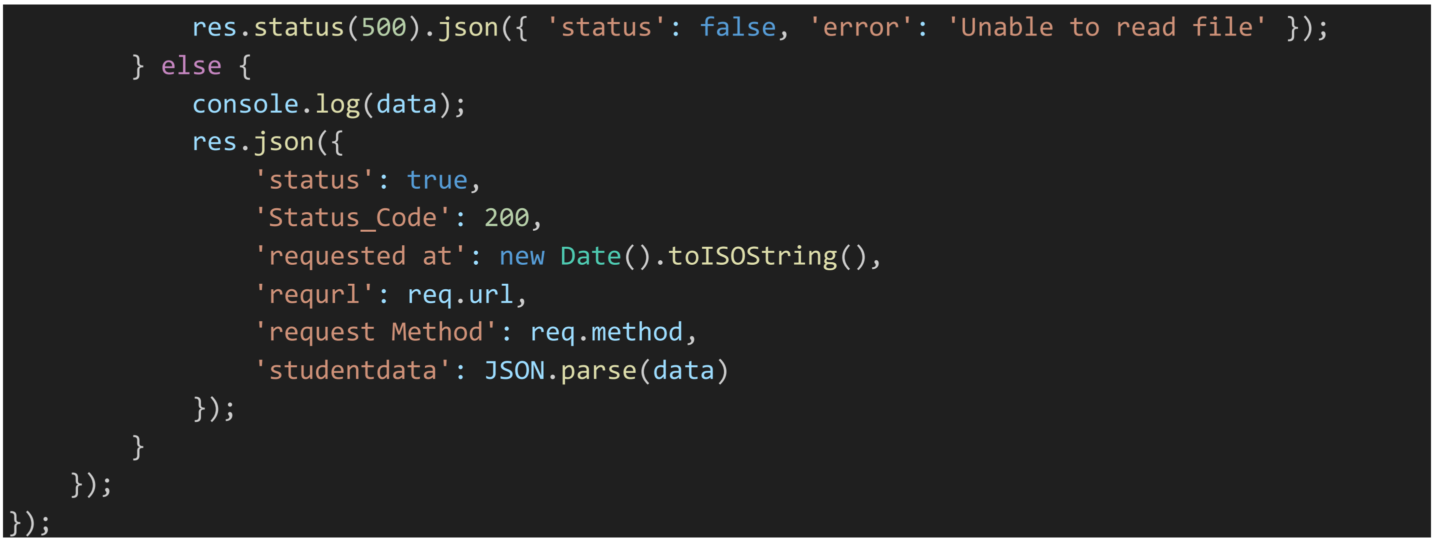
Step 1: Create a file name Student.json and add the following code to it:



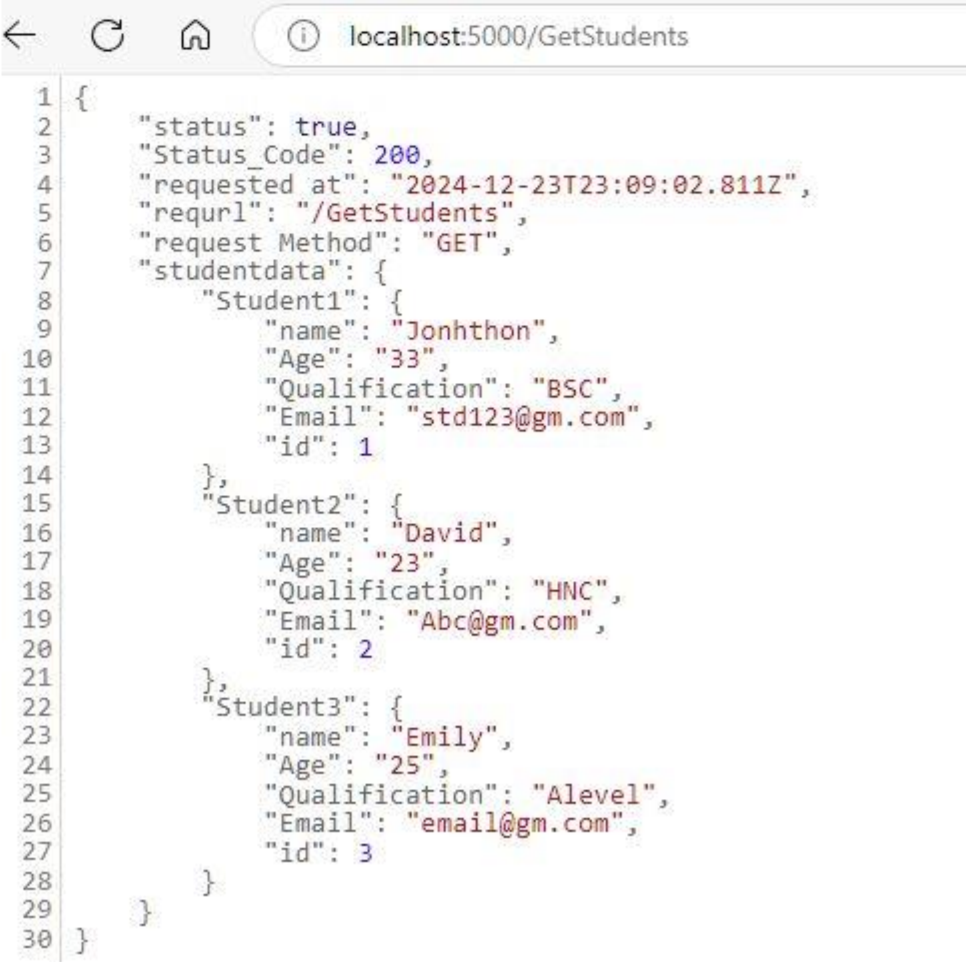
```
{
    "Student1": {
      "name": "Jonhthon",
      "Age": "33",
      "Qualification": "BSC",
      "Email": "std123@gm.com",
      "id": 1
    },
    "Student2": {
      "name": "David",
      "Age": "23",
      "Qualification": "HNC",
      "Email": "Abc@gm.com",
      "id": 2
    },
    "Student3": {
      "name": "Emily",
      "Age": "25",
      "Qualification": "Alevel",
      "Email": "email@gm.com",
      "id": 3
    }
  }
```

Step 2: Read this JSON file in your web application appending the following code in your index.js

```
var express = require("express");
var fs = require("fs");
var app = express();

// Add middleware function for body parsing
var bodyParser = require("body-parser");
app.use(bodyParser.urlencoded({ extended: true }));

// Root route
app.get('/', function(req, res) {
    res.send("Hello, it is my first Express application");
});

// About route
app.get('/about', function(req, res) {
    res.send("This is a basic express application. CN5006 Lab Session Week4 Prepared by Dr. N. Qazi");
});

// Dynamic route for user and book IDs
app.get('/users/:userId/books/:bookId', function(req, res) {
    res.send(req.params); // Send the userId and bookId parameters
});

// Function to start the server on an available port
function startServer(port) {
    app.listen(port, function() {
        console.log(`Server is running on port ${port}`);
    }).on('error', (err) => {
        if (err.code === 'EADDRINUSE') {
            console.log(`Port ${port} is in use, trying another port...`);
            startServer(port + 1);  // Try the next port
        } else {
            console.error('Server error:', err);
        }
    });
}

// Start server on port 5000, will retry if the port is in use
startServer(5000);

// Get students route to read the student data from a JSON file
app.get('/GetStudents', function (req, res) {
    fs.readFile(__dirname + "/" + "Student.json", 'utf8', function (err, data) {
        if (err) {
            console.error('Error reading file:', err);
```



```
            res.status(500).json({ 'status': false, 'error': 'Unable to read file' });
        } else {
            console.log(data);
            res.json({
                'status': true,
                'Status_Code': 200,
                'requested at': new Date().toISOString(),
                'requrl': req.url,
                'request Method': req.method,
                'studentdata': JSON.parse(data)
            });
        }
    });
});
```

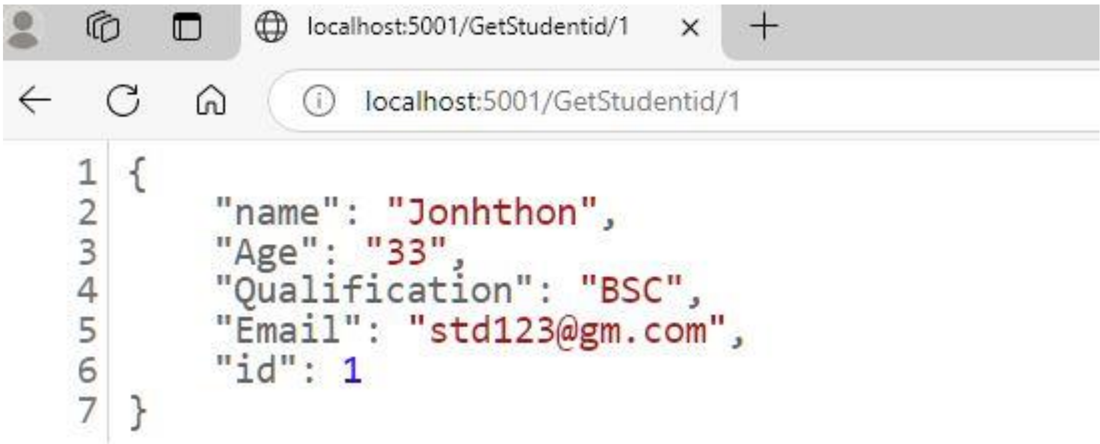


localhost:5000/GetStudents

```
{
    "status": true,
    "Status_Code": 200,
    "requested at": "2024-12-23T23:09:02.811Z",
    "requrl": "/GetStudents",
    "request Method": "GET",
    "studentdata": {
        "Student1": {
            "name": "Jonhthon",
            "Age": "33",
            "Qualification": "BSC",
            "Email": "std123@gm.com",
            "id": 1
        },
        "Student2": {
            "name": "David",
            "Age": "23",
            "Qualification": "HNC",
            "Email": "Abc@gm.com",
            "id": 2
        },
        "Student3": {
            "name": "Emily",
            "Age": "25",
            "Qualification": "Alevel",
            "Email": "email@gm.com",
            "id": 3
        }
    }
}
```

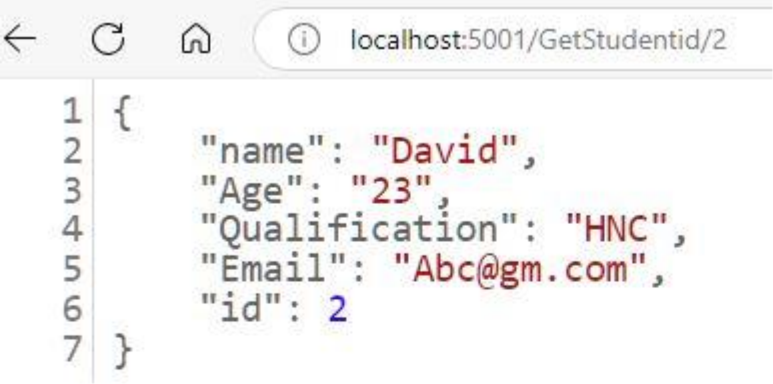
Step 3 Searching the JSON File :Press ctrl C and add more code in the index.js

http://localhost:5001/GetStudentid/1



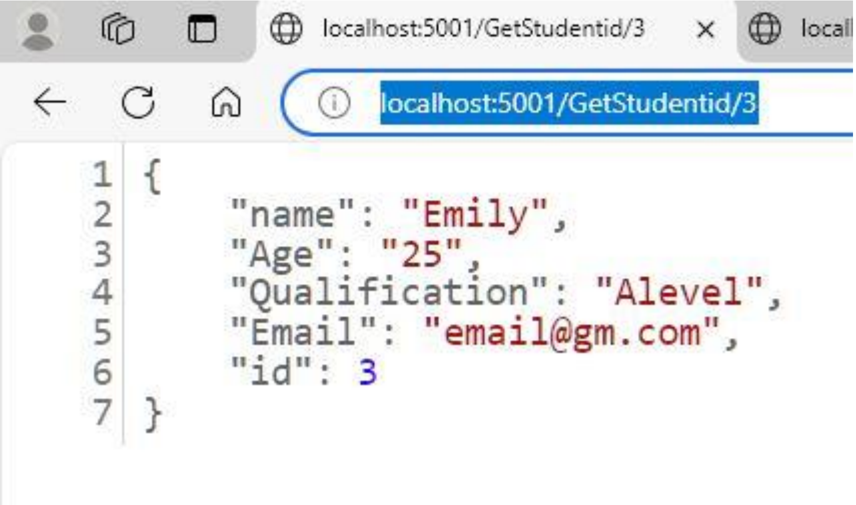
```
{
    "name": "Jonhthon",
    "Age": "33",
    "Qualification": "BSC",
    "Email": "std123@gm.com",
    "id": 1
}
```

http://localhost:5001/GetStudentid/2



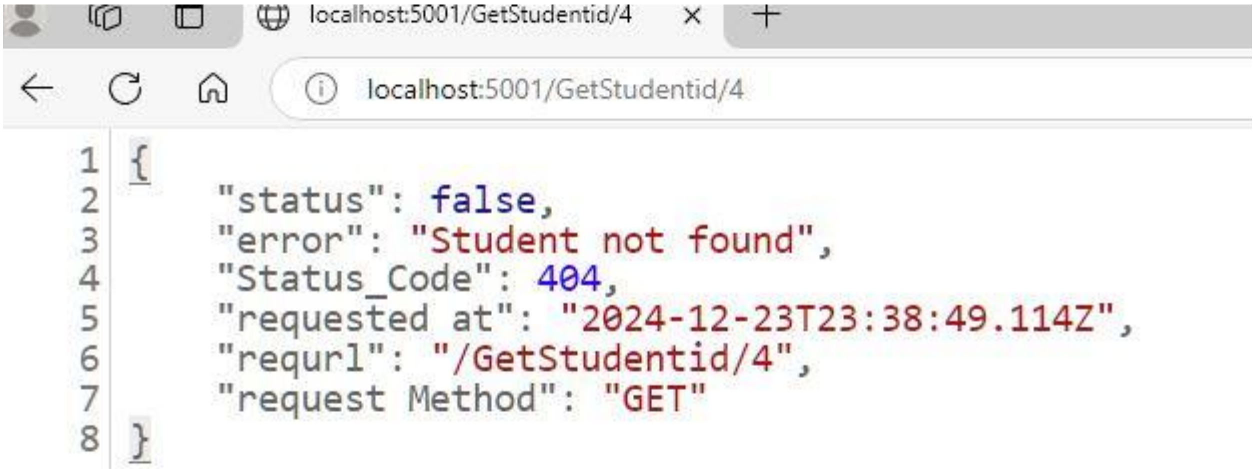
```
{
    "name": "David",
    "Age": "23",
    "Qualification": "HNC",
    "Email": "Abc@gm.com",
    "id": 2
}
```

http://localhost:5001/GetStudentid/3



```
{
    "name": "Emily",
    "Age": "25",
    "Qualification": "Alevel",
    "Email": "email@gm.com",
    "id": 3
}
```

http://localhost:5001/GetStudentid/4



localhost:5001/GetStudentid/4

```
{
    "status": false,
    "error": "Student not found",
    "Status_Code": 404,
    "requested at": "2024-12-23T23:38:49.114Z",
    "requrl": "/GetStudentid/4",
    "request Method": "GET"
}
```

Report what did you observe for following GetStudent/1, GetStudent/2 ,GetStudent/4 one by one and explain your output

1. **For `/GetStudentid/1`:**
   - The request successfully finds and returns the data for Jonhthon.
2. **For `/GetStudentid/2`:**
   - The request successfully finds and returns the data for David.
3. **For `/GetStudentid/4`:**
   - Since there is no `Student4` in the JSON data, the server responds with a `404 Not Found` error, indicating that the student doesn't exist.

The code logic works as expected, providing correct student data when the ID exists and returning an error when the ID is not found in the data.

Exercise 4

Step 1. use of Post Method Create a HTML file name it StudentInfo.html and write following code

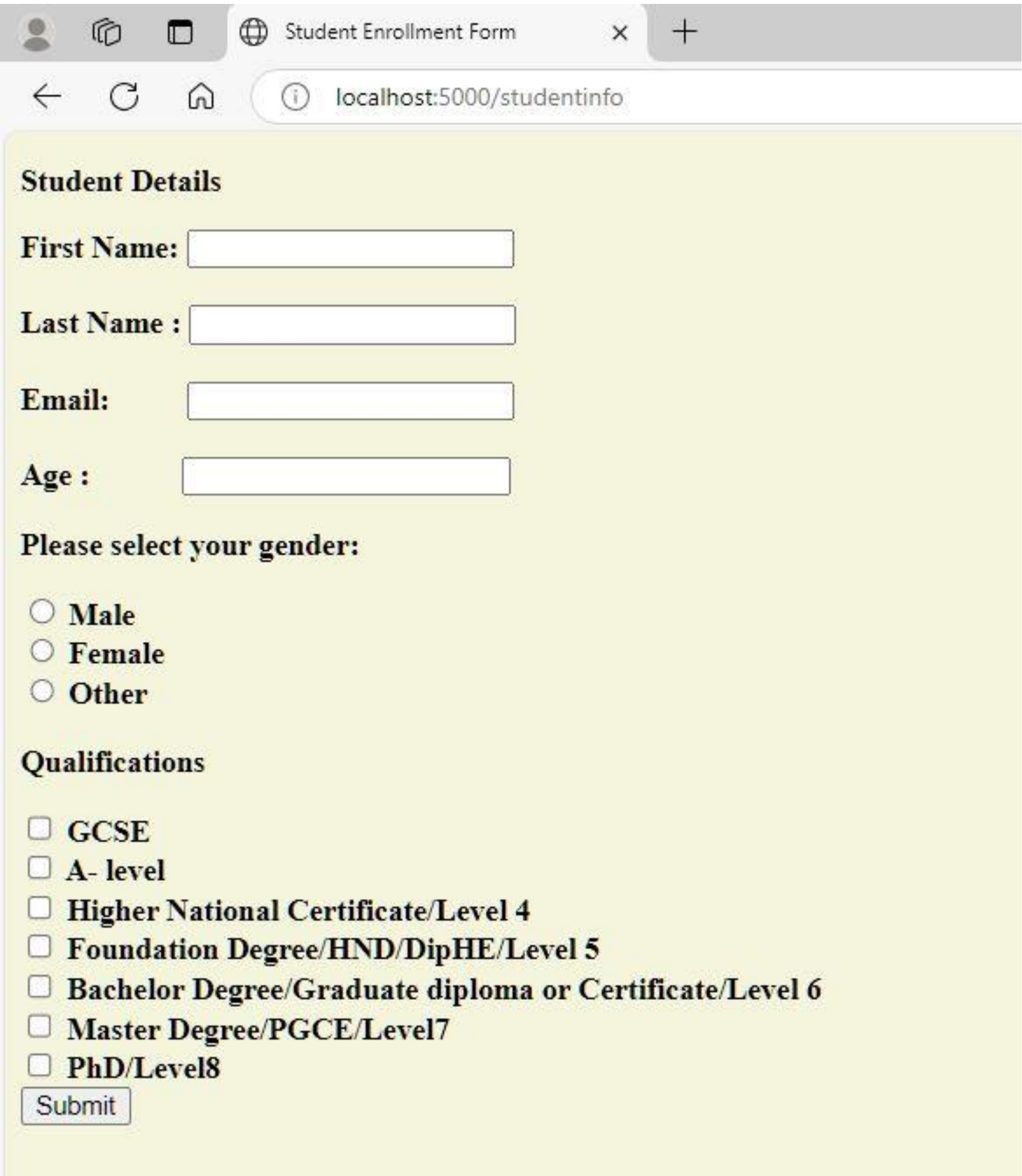
```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta charset="utf-8" />
<title>Student Enrollment Form</title>
</head>

<body bgcolor= beige>
    <form action="/submit-data" method="post" >
    <p><Strong> Student Details </p>
    First Name: <input name="firstName" type="text" /> <br/><br/>
    Last Name : <input name="lastName" type="text" /> <br/><br/>
    Email: &emsp;&emsp; <input type="email" id="emailid"
    name="email"/><br/> <br/>
    Age : &emsp;&emsp; &ensp;<input name="myAge" type="text" /> <br/>
```

```html
    <p>Please select your gender:</p>
    <input type="radio" id="male" name="gender" value="male">
    <label for="male">Male</label><br>
    <input type="radio" id="female" name="gender" value="female">
    <label for="female">Female</label><br>
    <input type="radio" id="other" name="gender" value="other">
    <label for="other">Other</label>
    <p> Qualifications </p>
    <input type="checkbox" name="Qual" value="GCSE"> GCSE <br/>
    <input type="checkbox" name="Qual" value="A-level"> A- level <br>
    <input type="checkbox" name="Qual" value="Higher National Certificate">
    Higher National Certificate/Level 4 <br/>
    <input type="checkbox" name="Qual" value="HND"> Foundation
    Degree/HND/DipHE/Level 5 <br/>
    <input type="checkbox" name="Qual" value="B.Sc"> Bachelor Degree/Graduate
    diploma or
    Certificate/Level 6 <br/>
    <input type="checkbox" name="Qual" value="Master Degree"> Master
    Degree/PGCE/Level7 <br/>
    <input type="checkbox" name="Qual" value="PhD"> PhD/Level8 <br/>
    <input type="submit" />
    </form>
    </body>
    </html>
```

Step 2 append following method in index.js

```javascript
app.get('/studentinfo',function(req,res)
{
res.sendFile('StudentInfo.html', { root: __dirname });
})
app.post('/submit-data', function (req, res) {
var name = req.body.firstName + ' ' + req.body.lastName+
' ';
var Age= req.body.myAge+ ' Gender: ' + req.body.gender+'
'
Qual= ' Qualification'+ req.body.Qual
console.log(req.body.Qual)
res.send({
status: true,
message: 'form Details', data: {
name: name, age:Age, Qualification:Qual,
}
});
});
```

http://localhost:5000/studentinfo you will see a form , fill the form and click submit



Student Details

First Name: ▭

Last Name : ▭

Email: ▭

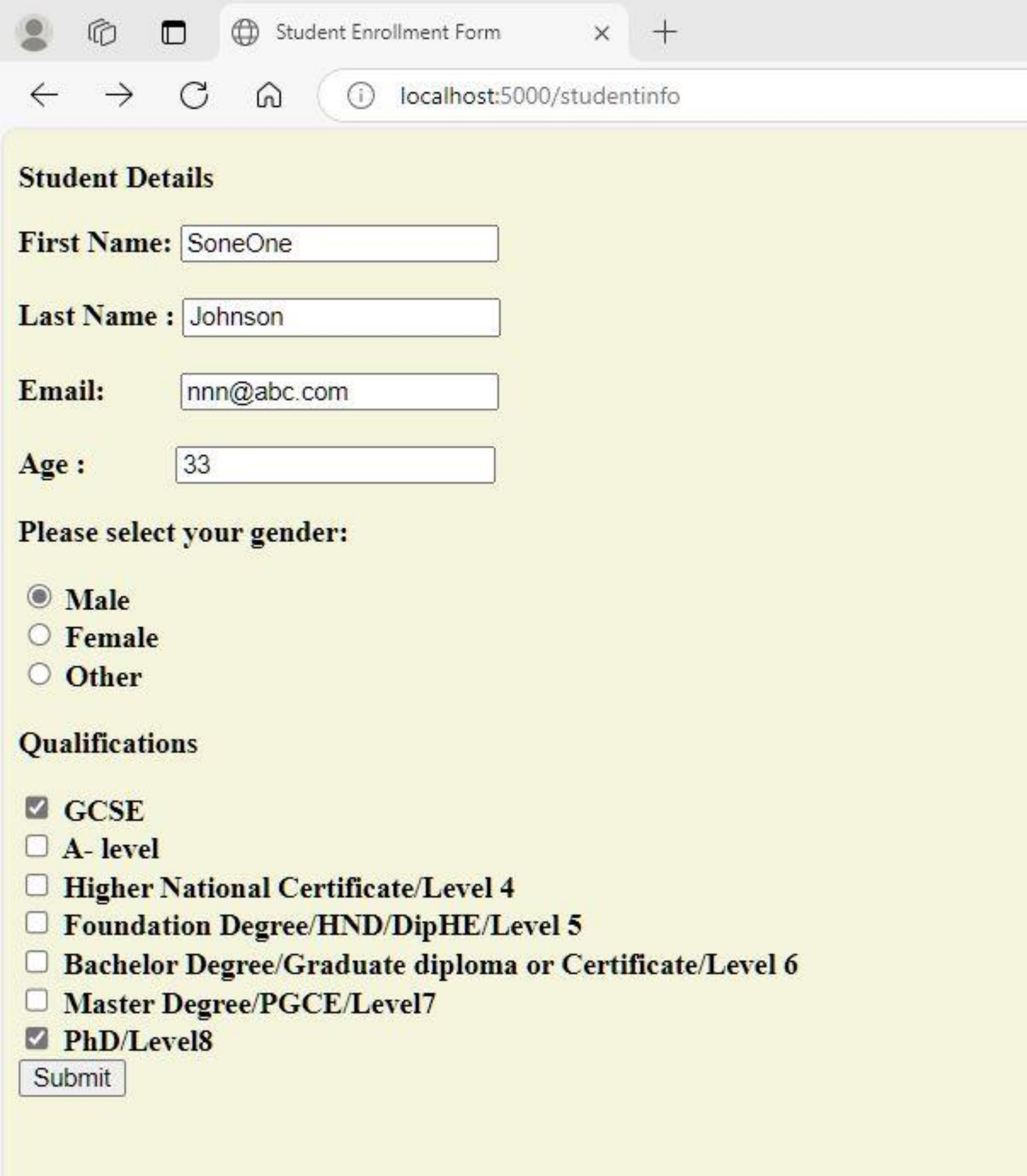
Age : ▭

Please select your gender:

○ **Male**
○ **Female**
○ **Other**

Qualifications

□ **GCSE**
□ **A- level**
□ **Higher National Certificate/Level 4**
□ **Foundation Degree/HND/DipHE/Level 5**
□ **Bachelor Degree/Graduate diploma or Certificate/Level 6**
□ **Master Degree/PGCE/Level7**
□ **PhD/Level8**



Student Details

First Name: SoneOne

Last Name : Johnson

Email: nnn@abc.com

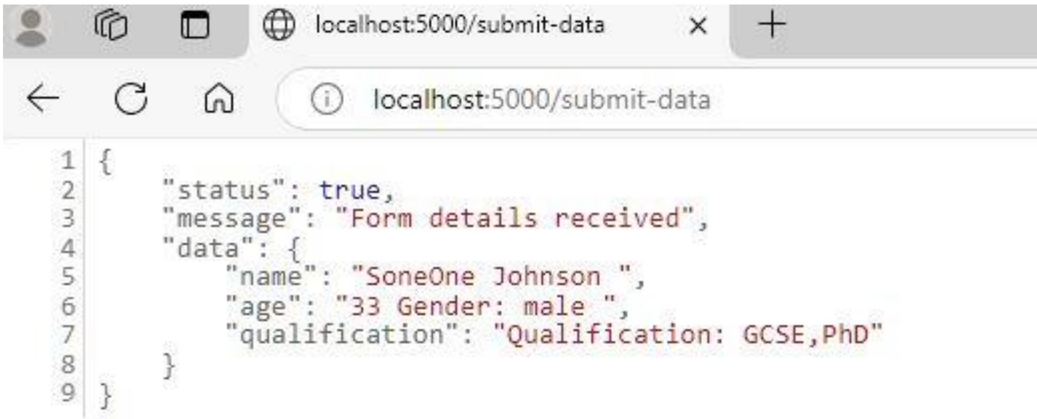
Age : 33

Please select your gender:

● **Male**
○ **Female**
○ **Other**

Qualifications

☑ **GCSE**
□ **A- level**
□ **Higher National Certificate/Level 4**
□ **Foundation Degree/HND/DipHE/Level 5**
□ **Bachelor Degree/Graduate diploma or Certificate/Level 6**
□ **Master Degree/PGCE/Level7**
☑ **PhD/Level8**



```
{
    "status": true,
    "message": "Form details received",
    "data": {
        "name": "SoneOne Johnson ",
        "age": "33 Gender: male ",
        "qualification": "Qualification: GCSE,PhD"
    }
}
```