

RESTFUL OPERATIONS

ABBIDI KRISHNA KOUSHIK

2303A51589

B-19

simple Restfull operation

```
const express = require('express');
const app = express();
app.use(express.json());
app.get('/students', (req, res) => {
  res.json([{ id: 1, name: "Ravi" }]);
});
app.post('/students', (req, res) => {
  const student = req.body;
  res.status(201).json({ message: "Student added", data: student });
});
app.listen(3000, () => {
  console.log("Server running on port 3000");
});
```

restfull operation with HTML

```
const express = require('express');
const app = express();
app.use(express.urlencoded({ extended: true }));
app.use(express.json());
let students = [];
app.get('/', (req, res) => {
  res.send(`
<h2>Add Student</h2>
<form method="POST" action="/students">
```

```

<input type="text" name="name" placeholder="Enter name" />
<button type="submit">Add</button>
</form>

`);
});
app.post('/students', (req, res) => {
  const name = req.body.name;
  const newStudent = {
    id: students.length + 1,
    name: name
  };
  students.push(newStudent);
  res.send("Student Added Successfully!");
});
app.get('/students', (req, res) => {
  res.json(students);
});
app.listen(3000, () => {
  console.log("Server running on port 3000");
});

#### resfull operation with POSTMAN desktop app(https://www.postman.com/downloads/)

const express = require('express');
const app = express();
app.use(express.json());
let students = [
  { id: 1, name: "Ravi" },
  { id: 2, name: "Sita" }
];

// 123 GET - View all students

```

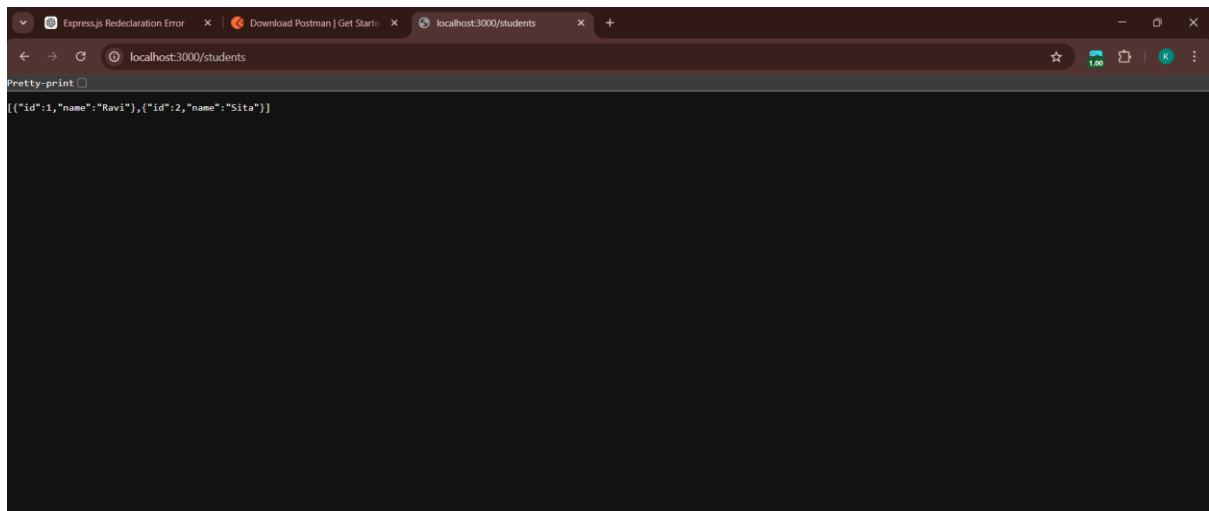
```
app.get('/students', (req, res) => {  
  res.json(students);  
});
```

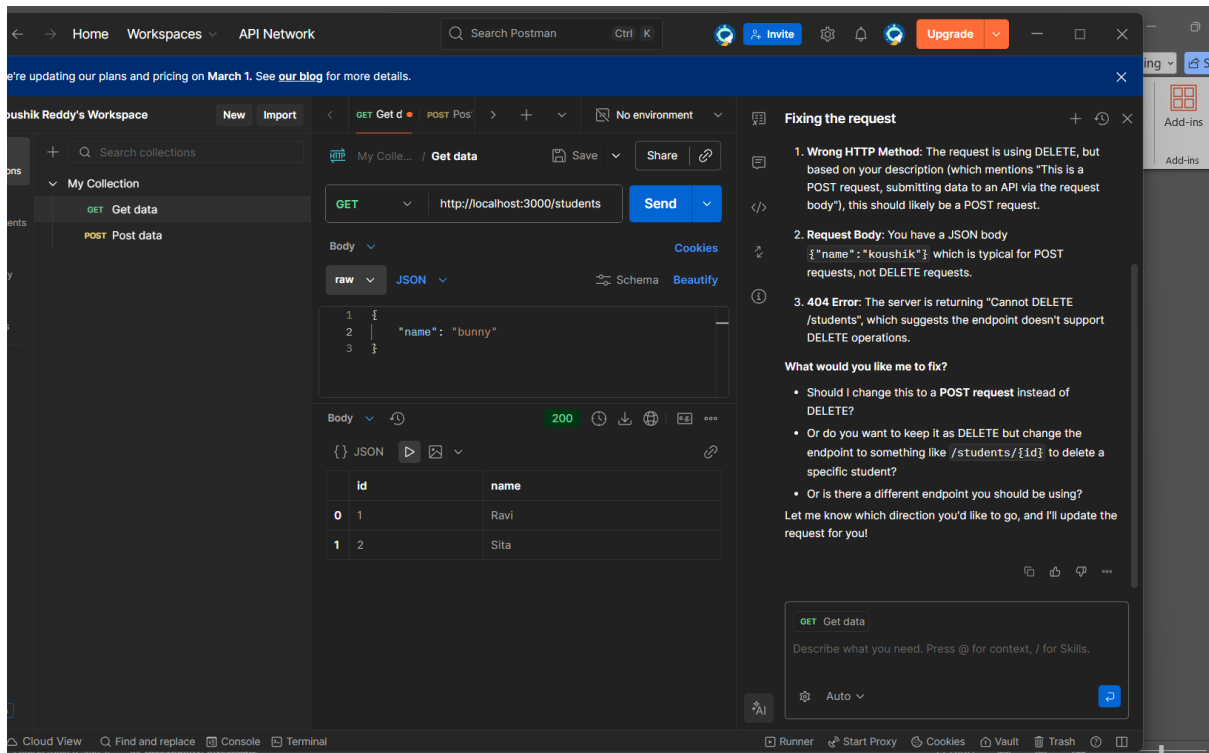
GET OPERATION:

/ GET - View all students

```
app.get('/students', (req, res) => {  
  res.json(students);  
});
```

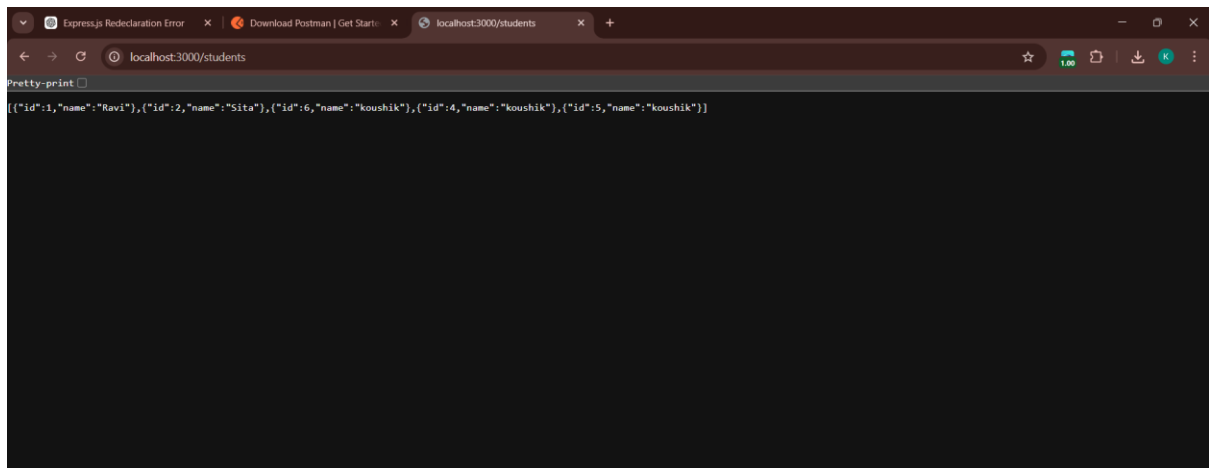
OUTPUT:



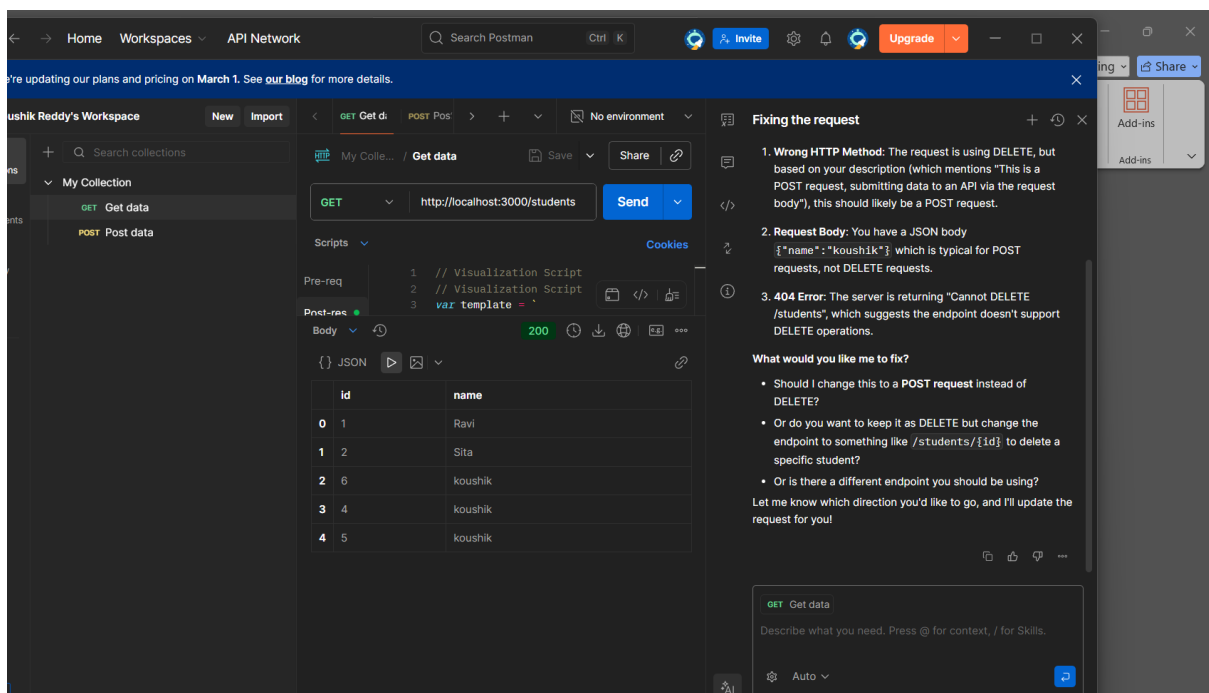


// GET - View single student

```
app.get('/students/:id', (req, res) => {  
  const id = parseInt(req.params.id);  
  const student = students.find(s => s.id === id);  
  if (!student) {  
    return res.status(404).json({ message: "Student not found" });  
  }  
  res.json(student);  
});
```



OUTPUT:



/ 4?? PUT - Update student

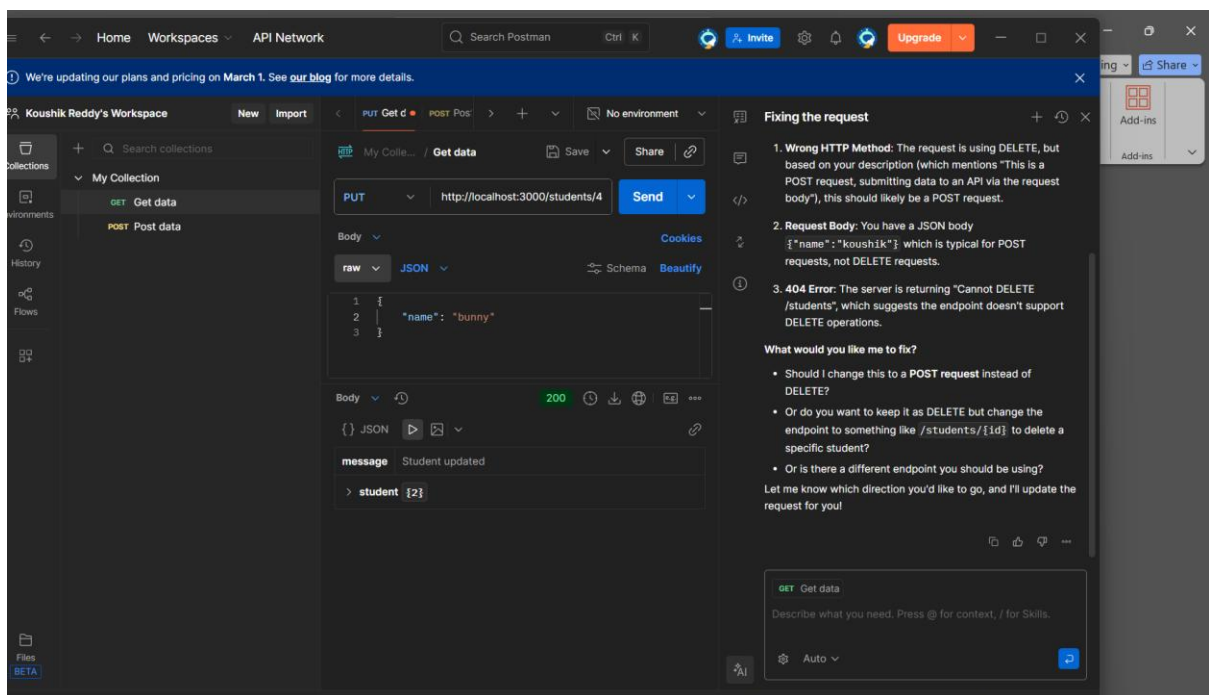
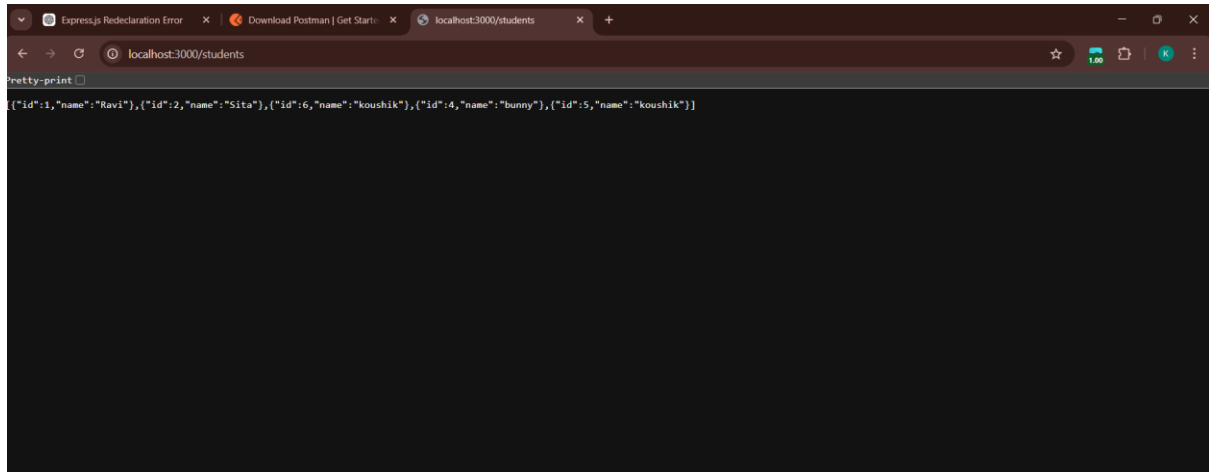
```
app.put('/students/:id', (req, res) => {
  const id = parseInt(req.params.id);
  const { name } = req.body;
  const student = students.find(s => s.id === id);
  if (!student) {
    return res.status(404).json({ message: "Student not found" });
  }
});
```

```
student.name = name;

res.json({ message: "Student updated", student });

});
```

OUTPUT:



```
// 500 DELETE - Remove student

app.delete('/students/:id', (req, res) => {

  const id = parseInt(req.params.id);

  students = students.filter(s => s.id !== id);
```

```
res.json({ message: "Student deleted" });

});

app.listen(3000, () => {

console.log("Server running on port 3000");

});
```

OUTPUT:

