

# RESTFUL OPERATIONS

Y.Sahasra  
2303A51269  
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" }
];
// 1 GET - View all students

```

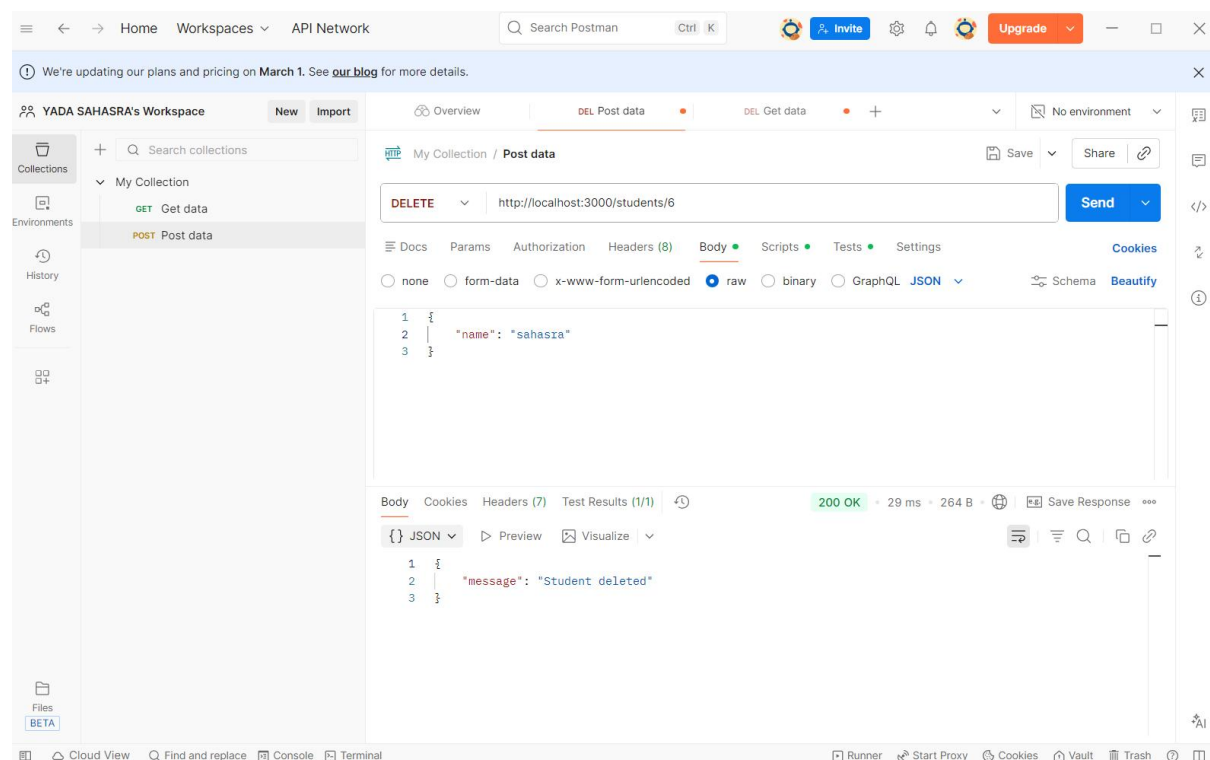
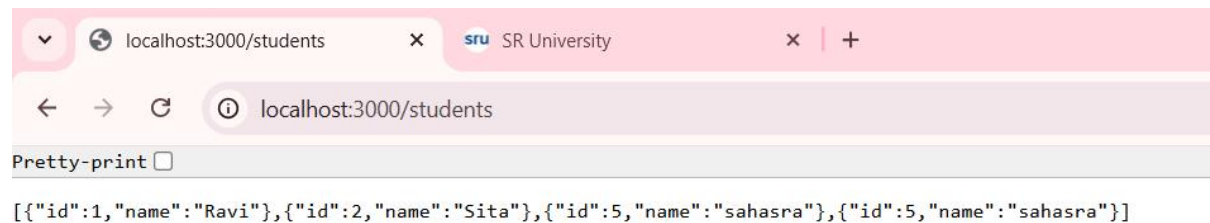
```
app.get('/students', (req, res) => {  
  res.json(students);  
});
```

GET OPERATION:

/ 1 GET - View all students

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

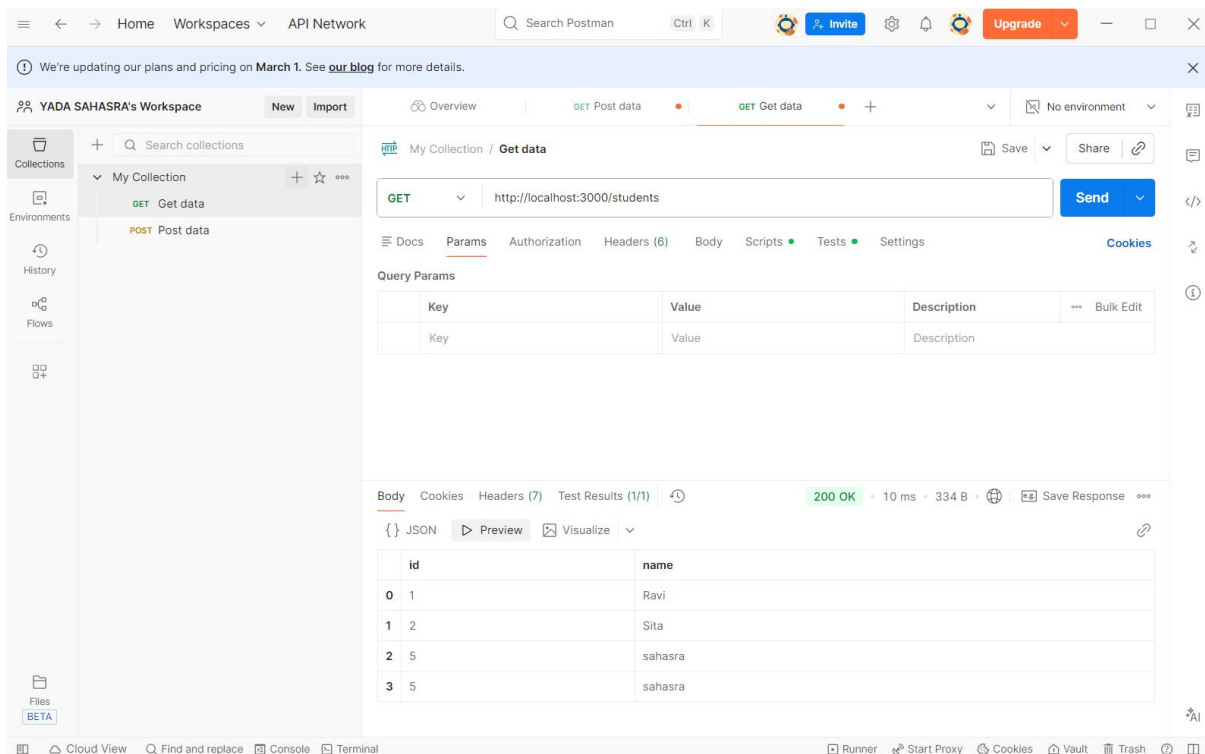
OUTPUT:



```
// 2 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:



The screenshot shows the Postman interface with a collection named 'My Collection' containing a request 'Get data'. The request is a GET method to the URL 'http://localhost:3000/students'. The response is a 200 OK status with a 10 ms response time and 334 B body. The body is a JSON array of student objects.

id	name
0	1
1	2
2	5
3	5

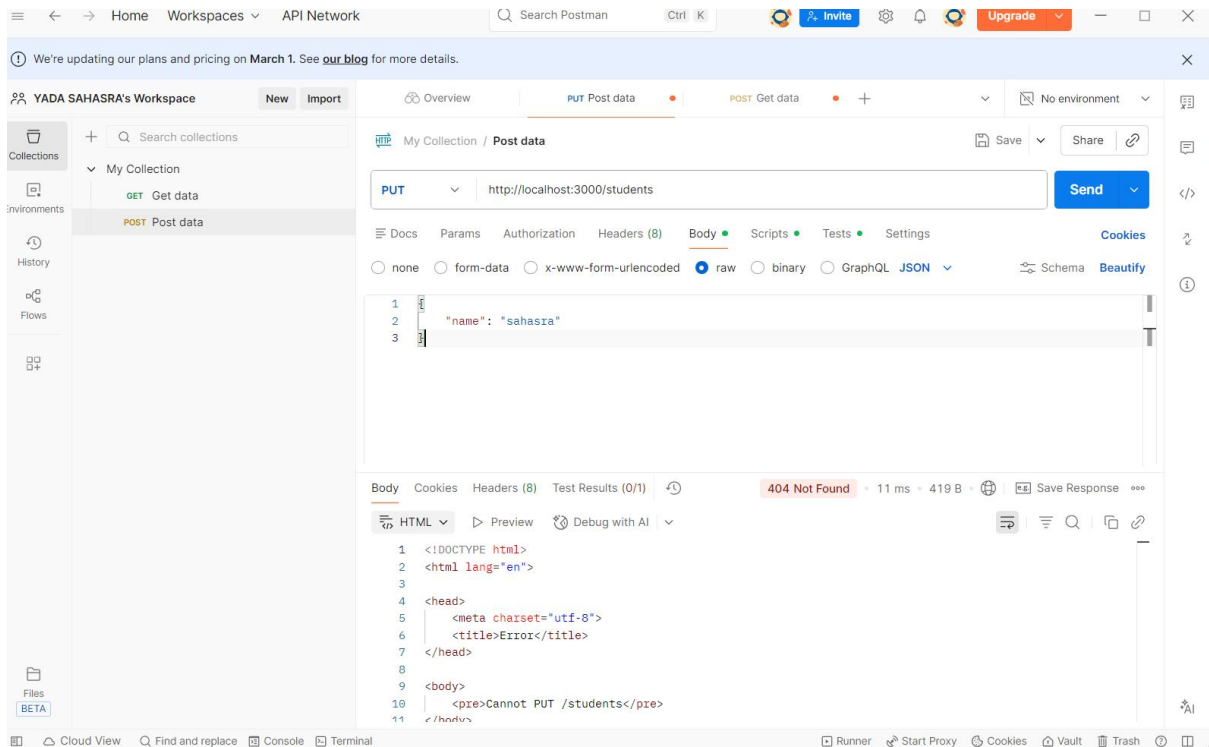
```
/ 4 PUT - Update student
```

```
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:





// 5 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:

Home Workspaces API Network Search Postman Ctrl K

We're updating our plans and pricing on March 1. See [our blog](#) for more details.

YADA SAHASRA's Workspace New Import Overview DEL Post data POST Get data No environment

My Collection / Post data Save Share

DELETE http://localhost:3000/students/6 Send

Docs Params Authorization Headers (8) Body Scripts Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Schema Beautify

```
1 {
2   |   "name": "sahasra"
3 }
```

Body Cookies Headers (7) Test Results (1/1) 200 OK 7 ms 264 B Save Response

JSON Preview Visualize

```
1 {
2   |   "message": "Student deleted"
3 }
```

Cloud View Find and replace Console Terminal Runner Start Proxy Cookies Vault Trash

localhost:3000/students SR University

localhost:3000/students

Pretty-print

```
[{"id":1,"name":"Ravi"}, {"id":2,"name":"Sita"}, {"id":5,"name":"sahasra"}, {"id":5,"name":"sahasra"}]
```