

Practical No 10

Aim : Write A Program To Demonstrate RESTFul API In Express.js

(GET, POST, PUT, DELETE)

```
import express from "express"
```

```
const port = 4001
```

```
const app = express()  
app.use(express.json())
```

```
let users = [  
  {  
    id: "user1",  
    username: "Ashif Shaikh",  
    class: "SY CS",  
    rollNo: 32  
  },  
  {  
    id: "user2",  
    username: "Arshad Shaikh",  
    class: "SY CS",  
    rollNo: 31  
  },  
  {  
    id: "user3",  
    username: "Farhan Shaikh",  
    class: "SY CS",  
    rollNo: 53  
  },  
  {  
    id: "user4",  
    username: "Raza Shaikh",  
    class: "SY CS",  
    rollNo: 49  
  },  
  {  
    id: "user5",  
    username: "Rounak Pal",  
    class: "SY CS",  
    rollNo: 46  
  },  
  {  
    id: "user6",  
    username: "Harish Pandey",  
    class: "SY CS",  
    rollNo: 24  
  },  
  {  
    id: "user7",
```

```

    username: "Harsh Salvi",
    class: "SY CS",
    rollNo: 29
  },
  {
    id: "user8",
    username: "Sachin D. Jaiswal",
    class: "SY CS",
    rollNo: 16
  },
  {
    id: "user9",
    username: "Vaibhav Soyane",
    class: "SY CS",
    rollNo: 45
  },
  {
    id: "user10",
    username: "Mayur Suryavanshi",
    class: "SY CS",
    rollNo: 42
  },
]

app.get("/", (req, res) => {
  res.status(200).send("<h1>Home Page</h1>")
})

app.get("/users", (req, res) => {
  res.status(200).json(users)
})

app.post("/users", (req, res) => {
  const user = req.body
  users.push(user)
  res.status(201).json("User Added Successfull.")
})

app.put("/users", (req, res) => {
  const user = req.body
  users = users.filter((u) => u.id !== user.id)
  users.push(user)
  res.status(201).json("User Updated Succesfull.")
})

app.delete("/users", (req, res) => {
  const id = req.body.id
  users = users.filter((u) => u.id !== id)
  res.status(201).json("User Deleted Succesfull.")
})

app.listen(port, () => {
  console.log("server started")
})

```

GET

localhost:4001/users

GET localhost:4001/users

Params Authorization Headers (7) Body Scripts Settings

Body Cookies Headers (7) Test Results

{ } JSON Preview Visualize

```
1 [
2   {
3     "id": "user1",
4     "username": "Ashif Shaikh",
5     "class": "SY CS",
6     "rollNo": 32
7   },
8   {
9     "id": "user2",
10    "username": "Arshad Shaikh",
11    "class": "SY CS",
12    "rollNo": 31
13  },
14  {
15    "id": "user3",
16    "username": "Farhan Shaikh",
17    "class": "SY CS",
18    "rollNo": 53
19  },
```

```
20  {
21    "id": "user4",
22    "username": "Raza Shaikh",
23    "class": "SY CS",
24    "rollNo": 49
25  },
26  {
27    "id": "user5",
28    "username": "Rounak Pal",
29    "class": "SY CS",
30    "rollNo": 46
31  },
32  {
33    "id": "user6",
34    "username": "Harish Pandey",
35    "class": "SY CS",
36    "rollNo": 24
37  },
38  {
39    "id": "user7",
40    "username": "Harsh Salvi",
41    "class": "SY CS",
42    "rollNo": 29
43  },
44  {
45    "id": "user8",
46    "username": "Sachin D. Jaiswal",
47    "class": "SY CS",
48    "rollNo": 16
49  },
```

POST

localhost:4001/users

POST localhost:4001/users

Params Authorization Headers (9) Body Scripts

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw

```
1 {
2   "id": "user11",
3   "username": "John Doe",
4   "class": "SY CS",
5   "rollNo": 0
6 }
```

Body Cookies Headers (7) Test Results

{ } JSON Preview Visualize

```
1 "User Added Successful."
```

```
44  {
45    "id": "user8",
46    "username": "Sachin D. Jaiswal",
47    "class": "SY CS",
48    "rollNo": 16
49  },
50  {
51    "id": "user9",
52    "username": "Vaibhav Soyane",
53    "class": "SY CS",
54    "rollNo": 45
55  },
56  {
57    "id": "user10",
58    "username": "Mayur Suryavanshi",
59    "class": "SY CS",
60    "rollNo": 42
61  },
62  {
63    "id": "user11",
64    "username": "John Doe",
65    "class": "SY CS",
66    "rollNo": 0
67  },
68 }
```

PUT

HTTP localhost:4001/users

PUT localhost:4001/users

Params Authorization Headers (9) **Body** Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐

```
1 {
2   "id": "user11",
3   "username": "John Doe",
4   "class": "SY CS",
5   "rollNo": 60
6 }
```

Body Cookies Headers (7) Test Results ↻

{ } JSON Preview Visualize

1 "User Updated Succesfull."

```
44 {
45   "id": "user8",
46   "username": "Sachin D. Jaiswal",
47   "class": "SY CS",
48   "rollNo": 16
49 },
50 {
51   "id": "user9",
52   "username": "Vaibhav Soyane",
53   "class": "SY CS",
54   "rollNo": 45
55 },
56 {
57   "id": "user10",
58   "username": "Mayur Suryavanshi",
59   "class": "SY CS",
60   "rollNo": 42
61 },
62 {
63   "id": "user11",
64   "username": "John Doe",
65   "class": "SY CS",
66   "rollNo": 60
67 }
68 ]
```

DELETE

HTTP localhost:4001/users

DELETE localhost:4001/users

Params Authorization Headers (9) **Body** Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐

```
1 {
2   "id": "user11"
3 }
```

Body Cookies Headers (7) Test Results ↻

{ } JSON Preview Visualize

1 "User Deleted Succesfull."

```
38 {
39   "id": "user7",
40   "username": "Harsh Salvi",
41   "class": "SY CS",
42   "rollNo": 29
43 },
44 {
45   "id": "user8",
46   "username": "Sachin D. Jaiswal",
47   "class": "SY CS",
48   "rollNo": 16
49 },
50 {
51   "id": "user9",
52   "username": "Vaibhav Soyane",
53   "class": "SY CS",
54   "rollNo": 45
55 },
56 {
57   "id": "user10",
58   "username": "Mayur Suryavanshi",
59   "class": "SY CS",
60   "rollNo": 42
61 }
62 ]
```