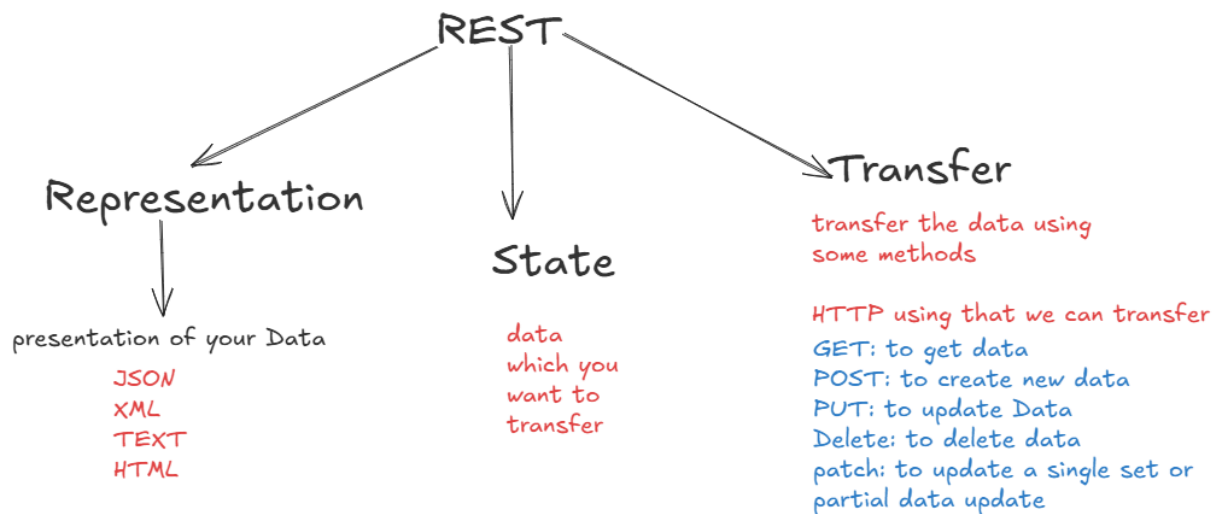
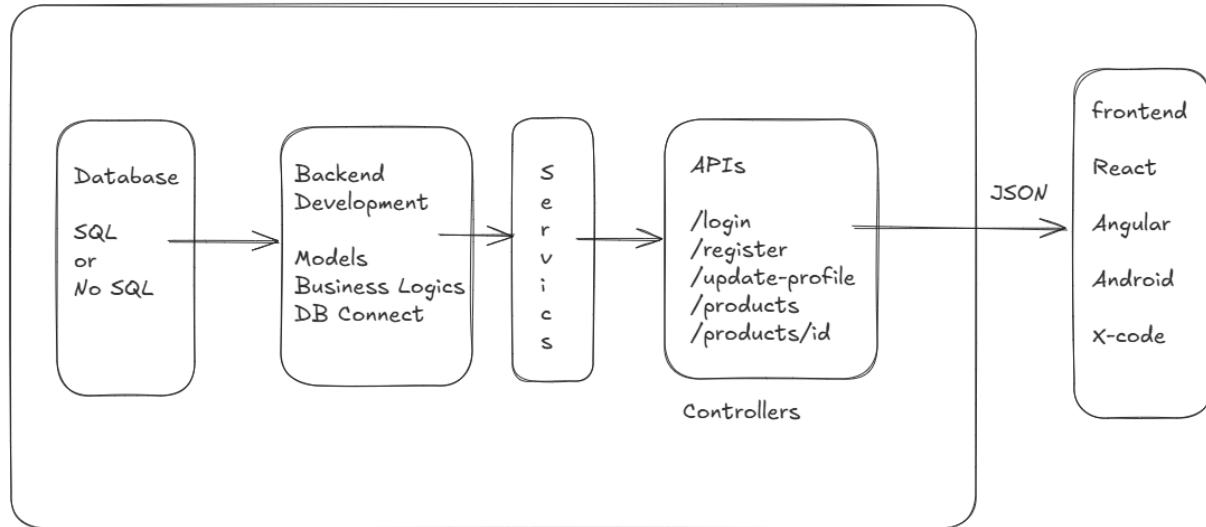


REST API



Let'S Create JSON-SERVER to Test APIS.

Create users.json file

```
{
  "users": [
    {
      "id": "1",
      "name": "Alex",
      "email": "alex@gmail.com",
      "address": "UK"
    }
  ]
}
```

```
},
{
  "id": "2",
  "name": "Bobby",
  "email": "bob@gmail.com",
  "address": "US"
},
{
  "id": "3",
  "name": "Catty",
  "email": "catty@gmail.com",
  "address": "US"
},
{
  "id": "4",
  "name": "test",
  "email": "test@gmail.com",
  "address": "US"
}
]
```

Open terminal execute below commands:

```
D:\SFJ Solutions\Mindsprint-Foundation-batch3\DayWiseUpdates-Git\day-8-API\server>npm i -g json-server
changed 45 packages in 5s

14 packages are looking for funding
  run `npm fund` for details
```

Check version:

```
C:\Users\NEW>json-server --version
1.0.0-beta.3
```

To start the server execute below command:

```
D:\SFJ Solutions\Mindsprint-Foundation-batch3\DayWiseUpdates-Git\day-8-API\server>json-server users.json
JSON Server started on PORT :3000
Press CTRL-C to stop
Watching users.json...
```

```
( ~ ~ ☺ ~ ~ )
```

Index:

<http://localhost:3000/>

Static files:

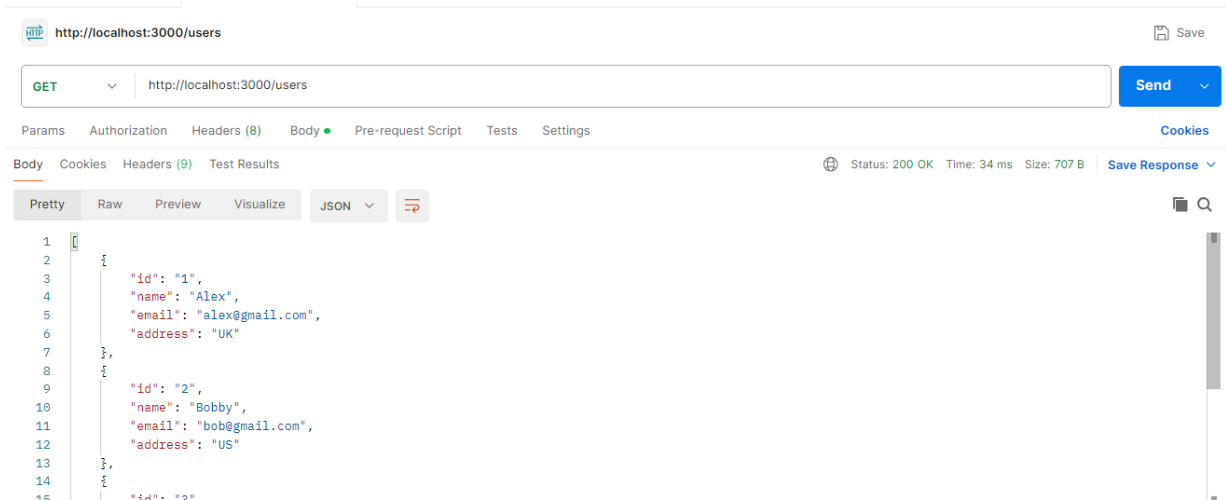
Serving ./public directory if it exists

Endpoints:

<http://localhost:3000/users>

the endpoint added here is your API.

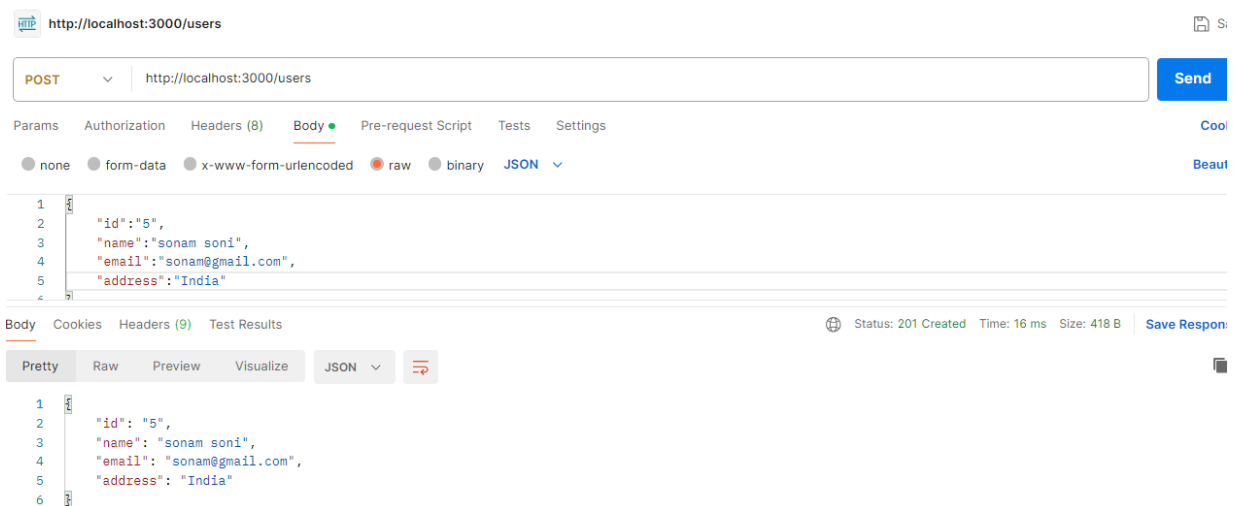
Get Request:



The screenshot shows a REST client interface with the URL `http://localhost:3000/users` and the method `GET`. The response status is `200 OK` with a time of `34 ms` and a size of `707 B`. The response body is displayed in JSON format, showing an array of two user objects:

```
[{"id": "1", "name": "Alex", "email": "alex@gmail.com", "address": "UK"}, {"id": "2", "name": "Bobby", "email": "bobb@gmail.com", "address": "US"}]
```

POST request: make sure you select body --> row data --> JSON --> Type JSON Data as shown here



The screenshot shows a REST client interface with the URL `http://localhost:3000/users` and the method `POST`. The request body is set to `JSON` and contains the following data:

```
{ "id": "5", "name": "sonam soni", "email": "sonam@gmail.com", "address": "India" }
```

The response status is `201 Created` with a time of `16 ms` and a size of `418 B`.

Update Id 4

PUT http://localhost:3000/users/4

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary JSON

```
1 {
2   "name": "John Doe",
3   "email": "john@gmail.com",
4   "address": "Philipines"
5 }
```

Body Cookies Headers (9) Test Results

Status: 200 OK Time: 14 ms Size: 415 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "name": "John Doe",
3   "email": "john@gmail.com",
4   "address": "Philipines",
5   "id": "4"
6 }
```

Delete Id 5

DELETE http://localhost:3000/users/5

Params Authorization Headers (8) Body Pre-request Script Tests Settings

Body Cookies Headers (9) Test Results

Status: 200 OK Time: 18 ms Size: 413 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": "5",
3   "name": "sonam soni",
4   "email": "sonam@gmail.com",
5   "address": "India"
6 }
```

You can see the deleted response in response body

Patch to add new field in existing object:

PATCH http://localhost:3000/users/3

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary JSON

```
1 {
2   "website": "www.blabla.com"
3 }
```

Body Cookies Headers (9) Test Results

Status: 200 OK Time: 8 ms Size: 437 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": "3",
3   "name": "Catty",
4   "email": "catty@gmail.com",
5   "address": "US",
6   "website": "www.blabla.com"
7 }
```

Let's Work With React

How to create React App: `npm create vite@latest`

give the project name: first-name

select react

select language: javascript

press enter

You can see the folder created named first-app (which is your React project)

then move to the folder: cd first-app

install dependencies: npm install

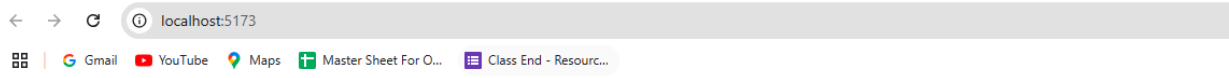
run app: npm run dev

```
> first-app@0.0.0 dev
> vite

VITE v6.1.1 ready in 488 ms

→ Local:   http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help
```

open the link and check the default output



Vite + React

count is 0

Edit src/App.jsx and save to test HMR

Click on the Vite and React logos to learn more