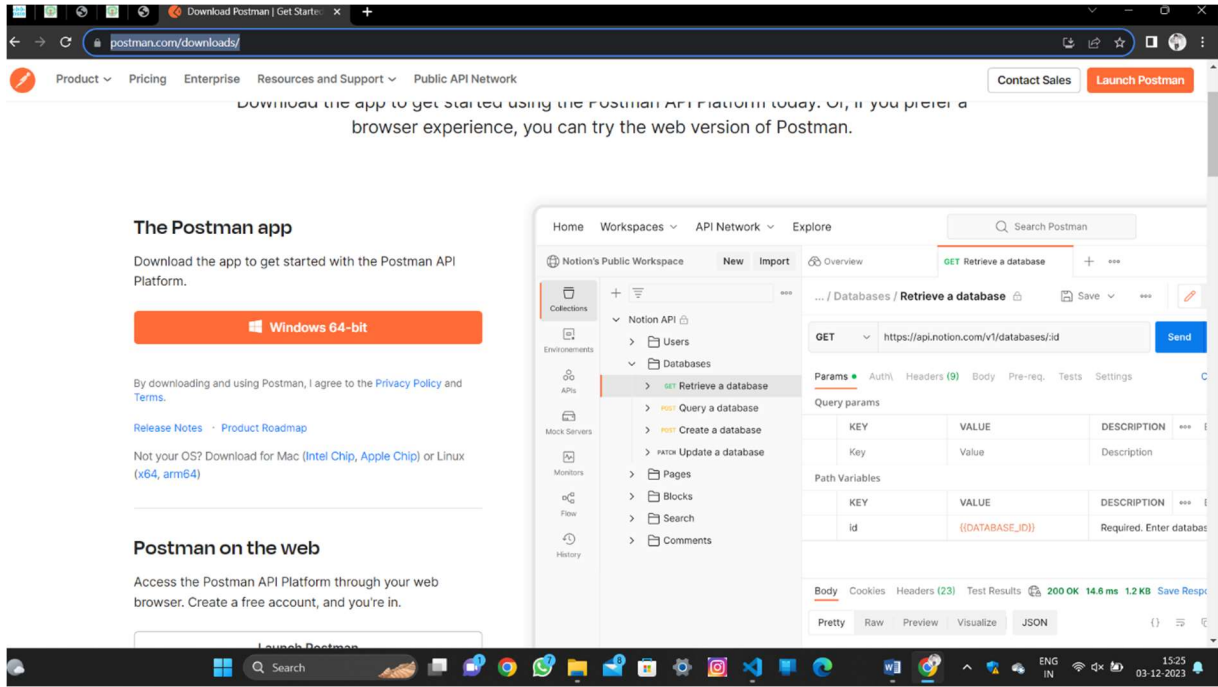
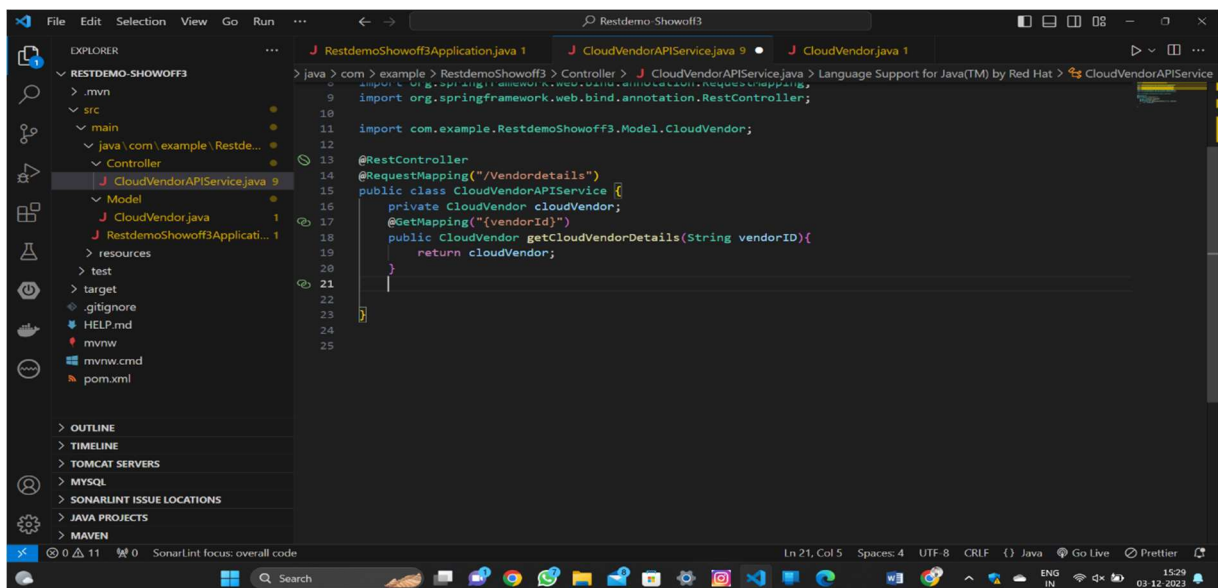


Showoff4_Documentation

1. Download Postman from this link: <https://www.postman.com/downloads/>

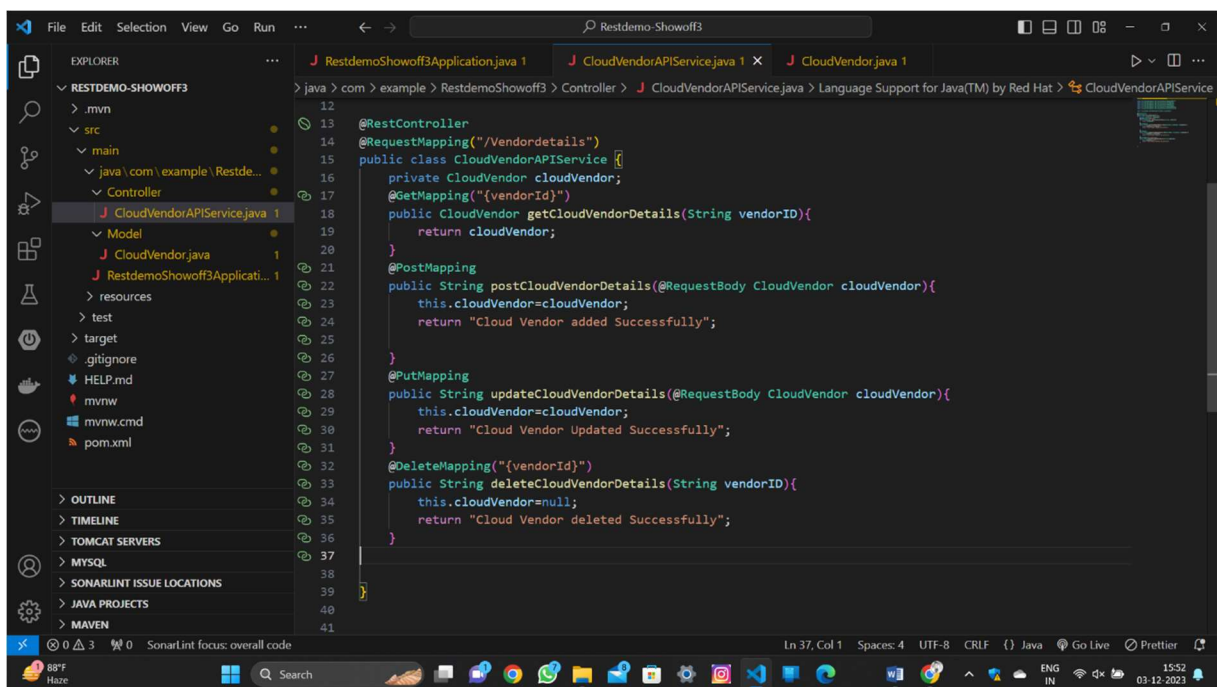


2. Now open previous code in Visualstudioapp, and open CloudVendorAPIService.java module.



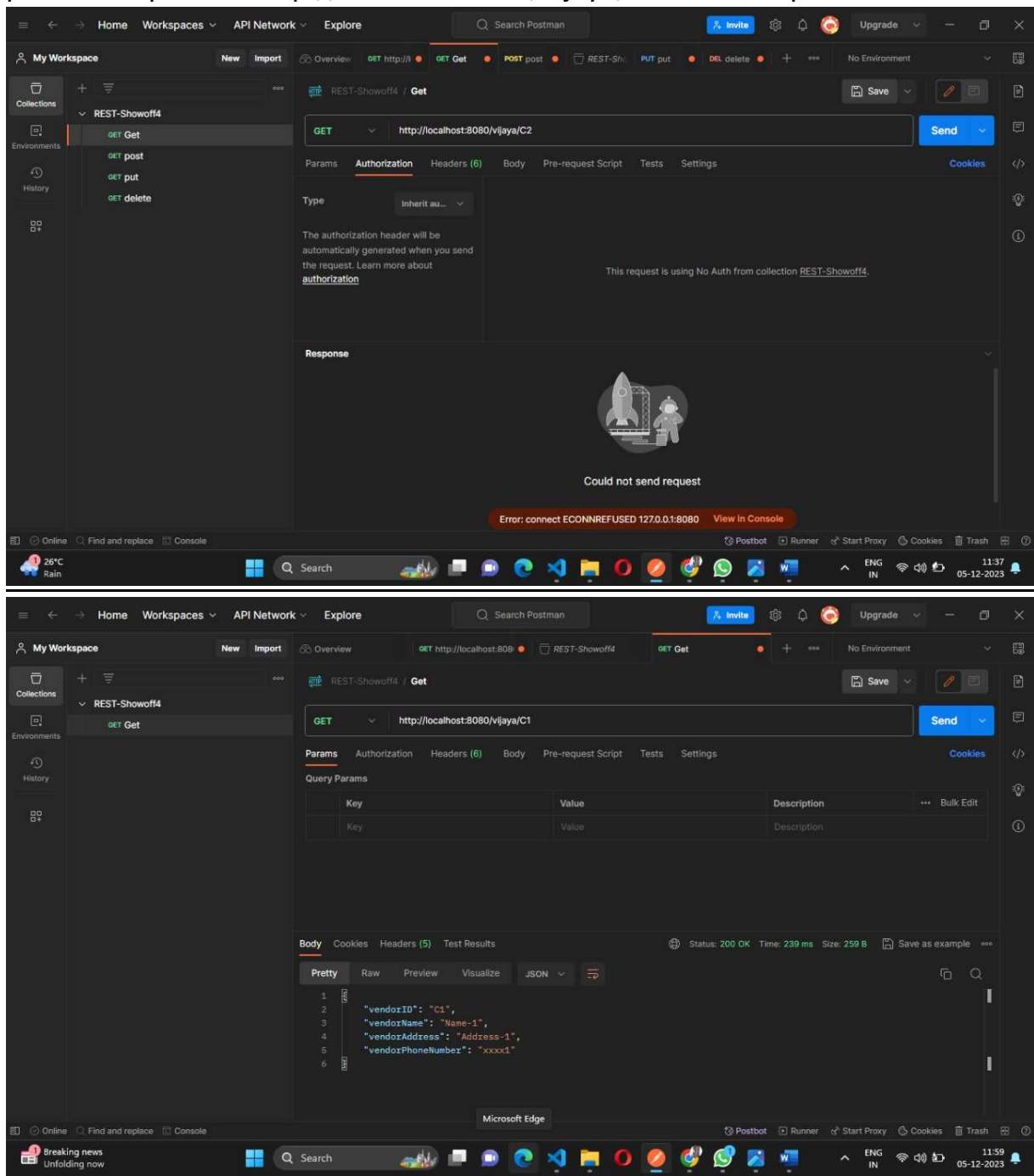
3. Open postman app and create Blank collection and give name to that blank collection as cloudvendorapiservice.java .
4. In the collection cloudvendorapiservice.java add a request and assign the name to that request as Get and save it.
5. Copy the url <http://localhost:8080/vijaya/C1> from the browser and past it in postman and click Send button.
6. It will display the following details.....
7. Again go to VSCode and this methods

- > @PostMapping
- > @PutMapping
- > @DeleteMapping



```
12
13 @RestController
14 @RequestMapping("/VendorDetails")
15 public class CloudVendorAPIService {
16     private CloudVendor cloudVendor;
17     @GetMapping("/{vendorId}")
18     public CloudVendor getCloudVendorDetails(String vendorID){
19         return cloudVendor;
20     }
21     @PostMapping
22     public String postCloudVendorDetails(@RequestBody CloudVendor cloudVendor){
23         this.cloudVendor=cloudVendor;
24         return "Cloud Vendor added Successfully";
25     }
26     @PutMapping
27     public String updateCloudVendorDetails(@RequestBody CloudVendor cloudVendor){
28         this.cloudVendor=cloudVendor;
29         return "Cloud Vendor Updated Successfully";
30     }
31     @DeleteMapping("/{vendorId}")
32     public String deleteCloudVendorDetails(String vendorID){
33         this.cloudVendor=null;
34         return "Cloud Vendor deleted Successfully";
35     }
36 }
37
38
39
40
41
```

8. Open postman and create another new request in Get and name it as Get.
past the copied url `http://localhost:8080/vijaya/C1` in Get request.



9. And past the copied url `http://localhost:8080/vijaya` in post request , Enter the details and click **Body -> raw->Text-> JSON**. click **Send** button.

The image shows two screenshots from a development environment. The top screenshot is from an IDE (likely IntelliJ IDEA) showing the source code of a REST API. The code defines a `RestController` with two endpoints: `getCloudVendorcloudVendorDetails` and `postCloudVendorDetails`. The `postCloudVendorDetails` endpoint is a `POST` request that takes a `CloudVendor` object as input and returns a success message.

```
@GetMapping("/{vendorID}")
public CloudVendor getCloudVendorcloudVendorDetails(String vendorID){
    return new CloudVendor(vendorID:"C1",vendorName:"Name-1",vendorAddress:"Address-1",vendorPhoneNumber:"xxxx1");
}

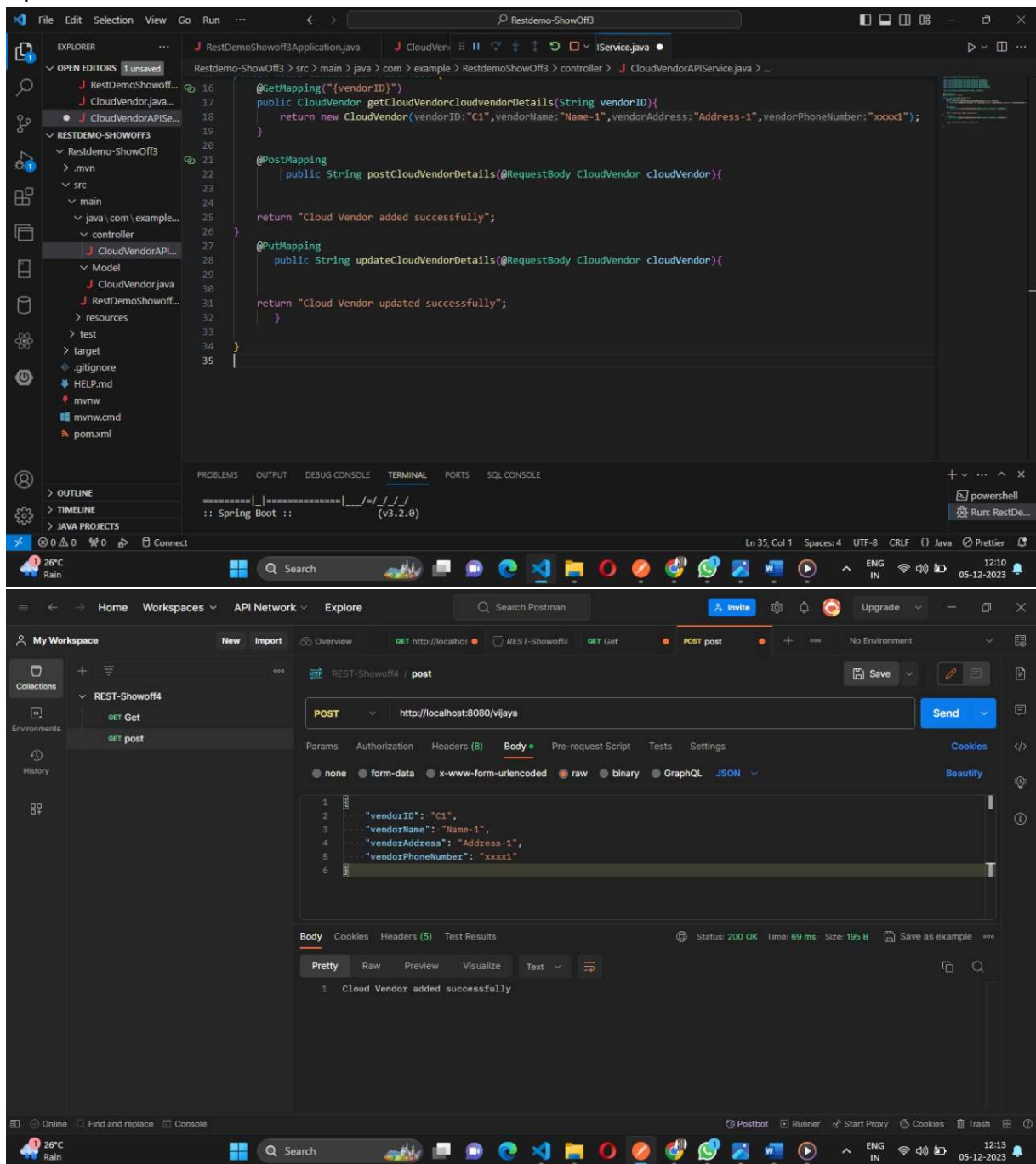
@PostMapping
public String postCloudVendorDetails(@RequestBody CloudVendor cloudVendor){
    return "Cloud Vendor added successfully";
}
```

The bottom screenshot is from Postman, showing a `POST` request to `http://localhost:8080/vijaya`. The request body is a JSON object with the following fields: `vendorID`, `vendorName`, `vendorAddress`, and `vendorPhoneNumber`. The response status is `415 Unsupported Media Type`, and the response body is a JSON object with the following fields: `timestamp`, `status`, `error`, and `path`.

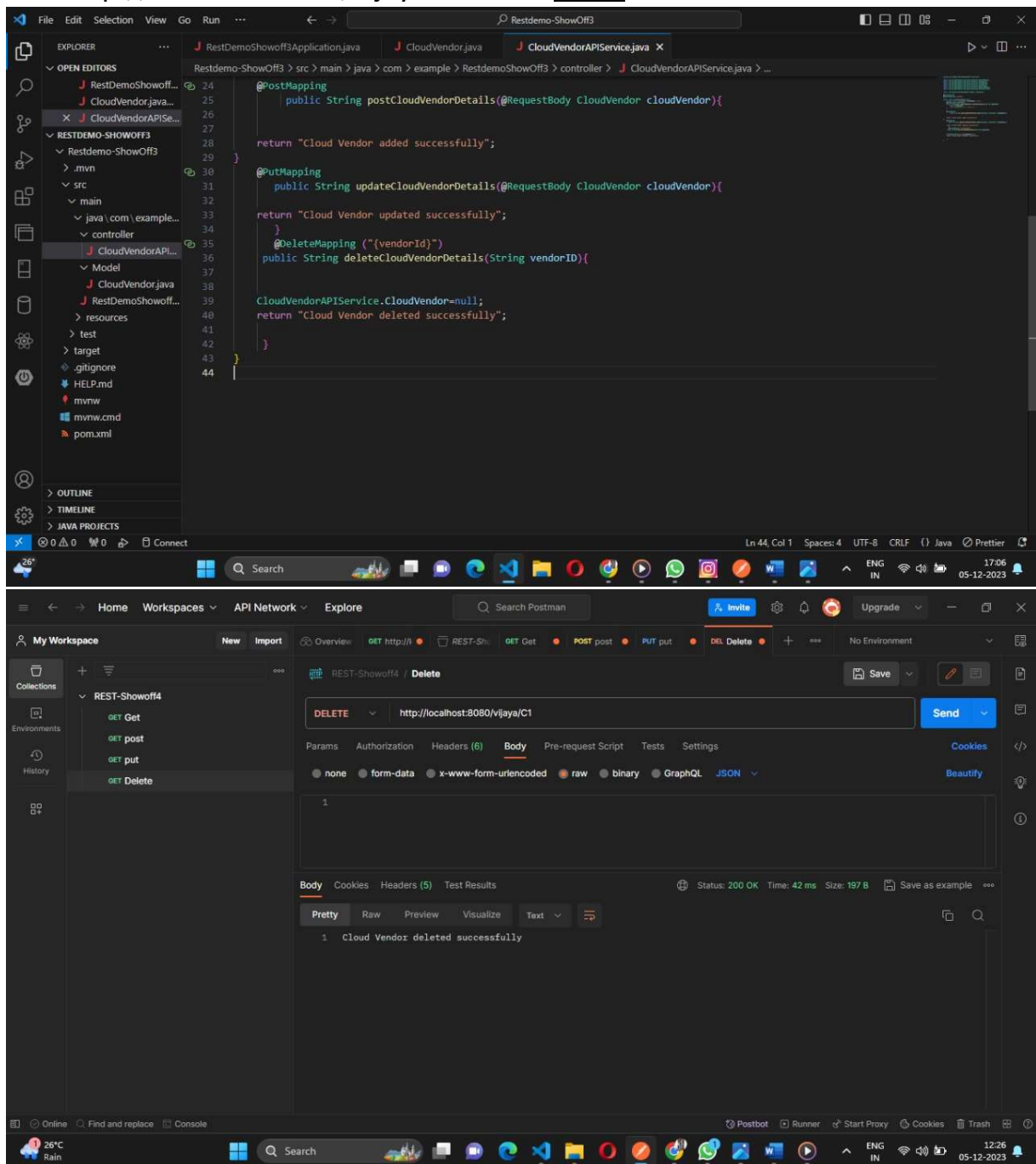
```
{
  "vendorID": "C1",
  "vendorName": "Name-1",
  "vendorAddress": "Address-1",
  "vendorPhoneNumber": "xxxx1"
}
```

```
{
  "timestamp": "2023-12-05T06:37:13.720+00:00",
  "status": 415,
  "error": "Unsupported Media Type",
  "path": "/vijaya"
}
```

10. Create a new request file in Put and name it as Put .Then the details are updated.



11. Create a new request file in Delete and name it as Delete. And enter the URL <http://localhost:8080/vijaya> and click Send button.



12. It will delete the present details .

13. Check in the Get request to confirm whether the details or deleted or not.

