T.Y.B.Tech CSE 2023-24

Name : Parimal Shrikant Kolhe
Roll No : PB_01
Sub : FSD

Sem : 5
Panel : B
Batch : B1

Lab Assignment 6

Aim: Develop a set of REST API using Express and Node.

Objectives:

1. To define HTTP GET and POST operations.
2. To understand and make use of 'REST', 'a REST endpoint', 'API Integration', and 'API Invocation'
3. To understand the use of a REST Client to make POST and GET requests to an API.

Theory:

1. What is REST API?

REST stands for Representational State Transfer. It's an architectural style for designing networked applications. RESTful APIs (Application Programming Interfaces) adhere to the principles of REST, which focus on using standard HTTP methods for communication between clients and servers.

A REST API is a set of rules and conventions that allow different computer systems to communicate with each other over the internet. It operates based on the principles of REST, utilizing HTTP requests to perform actions like GET (retrieve a resource), POST (create a resource), PUT (update a resource), DELETE (remove a resource), etc. REST APIs typically return data in a format like JSON or XML, which can be easily consumed by other systems.

2. Main purpose of REST API.

The primary purpose of a REST API is to enable communication and interaction between different software systems in a standardized, scalable, and flexible manner. It allows different applications, regardless of their underlying programming languages or platforms, to access and exchange data with each other over the internet. REST APIs provide a structured way for clients (such as web applications, mobile apps, or other services) to make requests to servers, retrieve data, and perform various actions on resources.

By following the REST principles, these APIs ensure interoperability, simplicity, reliability, and scalability in the communication between software components, making them a popular choice for building modern web services and applications.

FAQ:

1. What are HTTP Request types?

HTTP Request Types:

1. GET:

Purpose: Retrieve data from a specified resource.

Usage: Used for reading data without causing any side effects or modifications on the server.

2. POST:

Purpose: Submit data to be processed to a specified resource.

Usage: Often used for creating new resources or sending data that needs to be processed.

3. PUT:

Purpose: Update a resource (if it exists) at a specified URL. It can also create a new resource if it doesn't exist.

Usage: Used to completely replace the target resource with the request payload.

4. DELETE:

Purpose: Remove a specified resource.

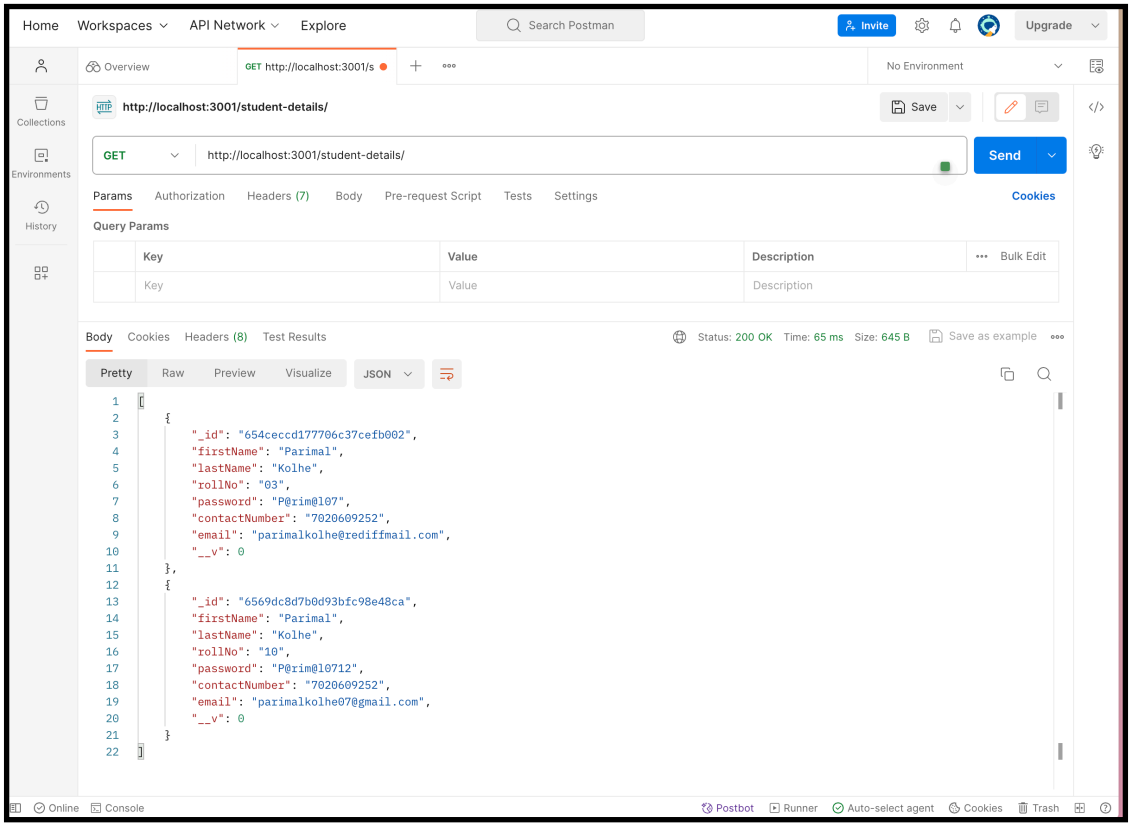
Usage: Used to delete the specified resource.

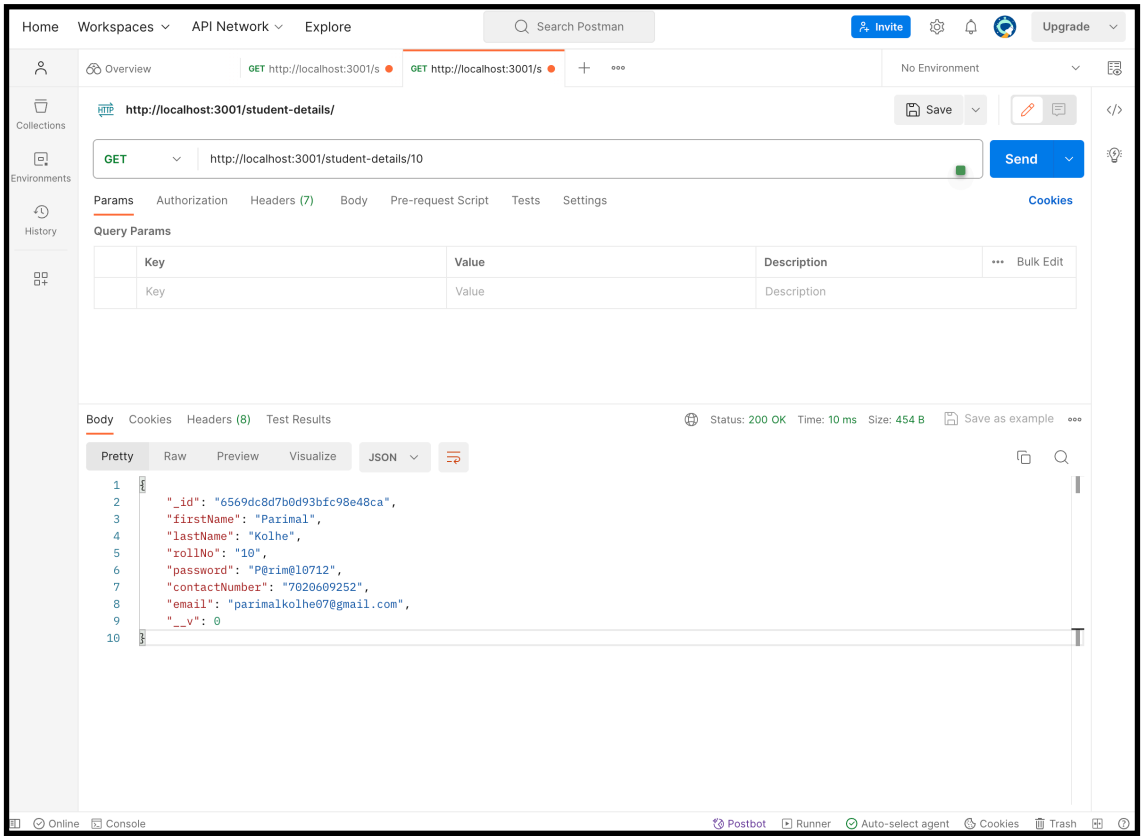
5. PATCH:
Purpose: Apply partial modifications to a resource.
Usage: Used when only a part of the resource needs to be updated.
6. OPTIONS:
Purpose: Fetch information about the communication options available for the target resource.
Usage: Helps in understanding the allowed methods on a resource, its supported MIME types, etc.

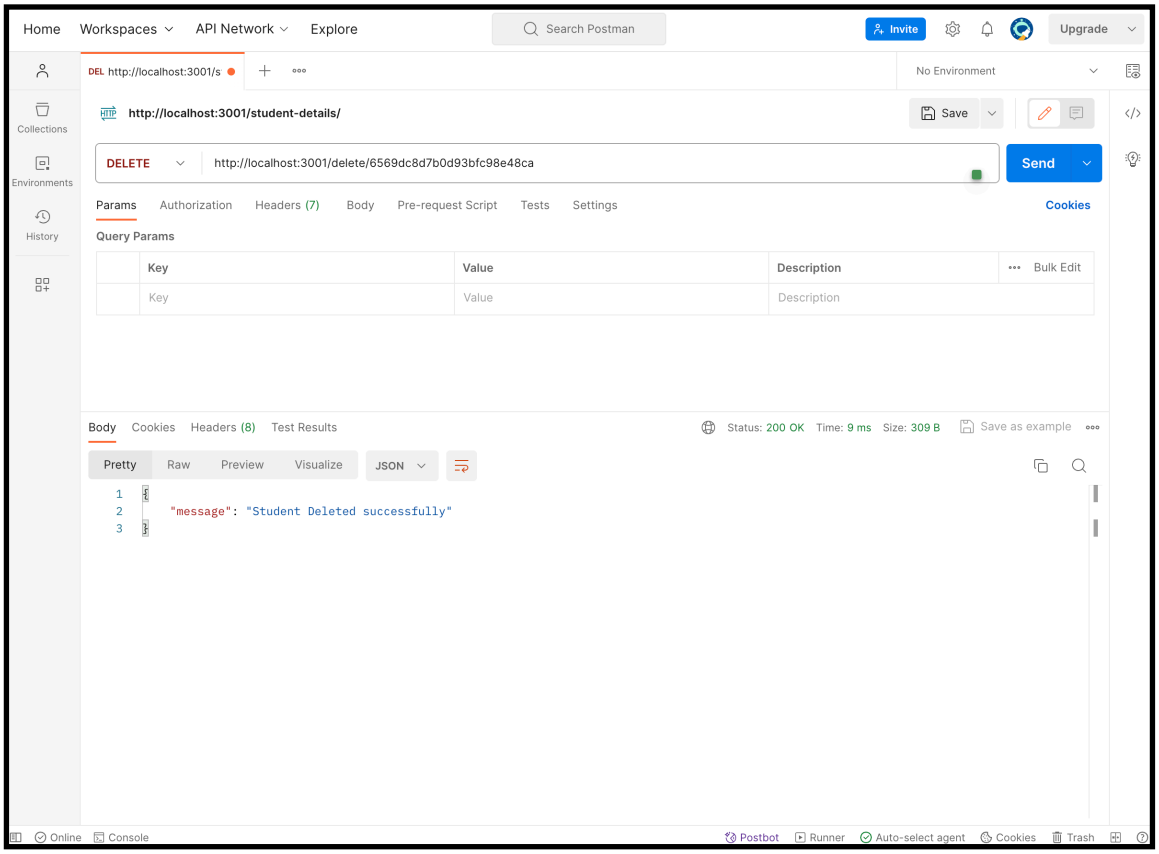
7. HEAD:
Purpose: Retrieve only the headers for a resource without the body content.
Usage: Used to check resource metadata (like headers, status code) without transferring the entire body.

Each HTTP method carries a specific purpose and usage, allowing developers to perform various operations on resources while adhering to RESTful principles.

Output :




Home
Workspaces
API Network
Explore
Search Postman
Invite
Upgrade
Overview
GET http://localhost:3001/s
GET http://localhost:3001/s
No Environment
Collections
Environments
History
http://localhost:3001/student-details/
Save
GET
http://localhost:3001/student-details/10
Send
Params
Authorization
Headers (7)
Body
Pre-request Script
Tests
Settings
Cookies
Query Params
Key
Value
Description
Bulk Edit
Key
Value
Description
Body
Cookies
Headers (8)
Test Results
Status: 200 OK
Time: 10 ms
Size: 454 B
Save as example
Pretty
Raw
Preview
Visualize
JSON
{
"_id": "6569dc8d7b0d93bfc98e48ca",
"firstName": "Parimal",
"lastName": "Kolhe",
"rollNo": "10",
"password": "P@rim@l0712",
"contactNumber": "7020609252",
"email": "parimalkolhe07@gmail.com",
"__v": 0
}
Online
Console
Postbot
Runner
Auto-select agent
Cookies
Trash


Home
Workspaces
API Network
Explore
Search Postman
Invite
Upgrade
DEL http://localhost:3001/s
No Environment
Collections
Environments
History
http://localhost:3001/student-details/
Save
DELETE
http://localhost:3001/delete/6569dc8d7b0d93bfc98e48ca
Send
Params
Authorization
Headers (7)
Body
Pre-request Script
Tests
Settings
Cookies
Query Params
Key
Value
Description
Bulk Edit
Key
Value
Description
Body
Cookies
Headers (8)
Test Results
Status: 200 OK
Time: 9 ms
Size: 309 B
Save as example
Pretty
Raw
Preview
Visualize
JSON
{
"message": "Student Deleted successfully"
}
Online
Console
Postbot
Runner
Auto-select agent
Cookies
Trash