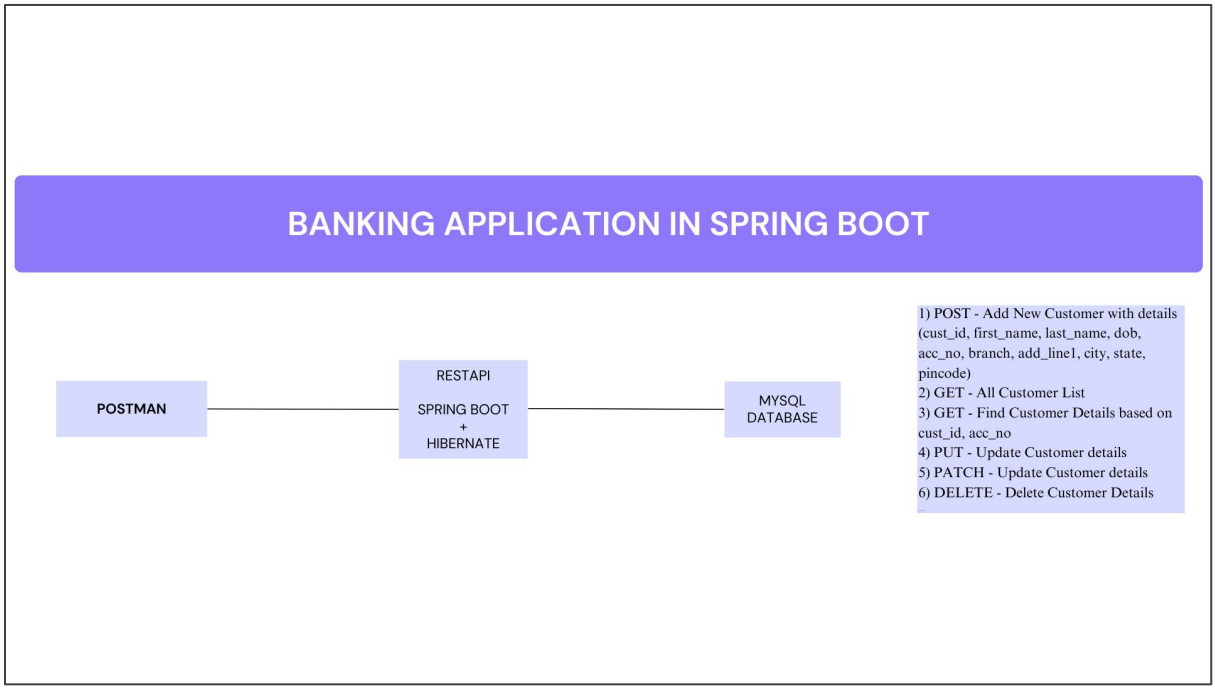
Verinite Technologies Pvt Ltd
Banking Application Spring Boot



Spring Boot Annotations and Their Internals

Here’s a rundown of common Spring Boot annotations and what they do:

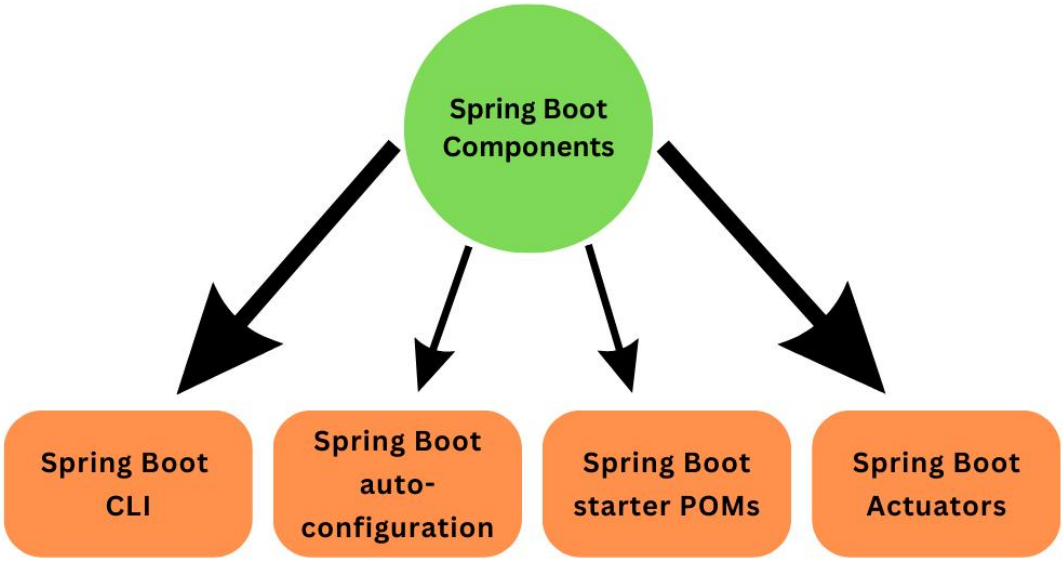
@SpringBootApplication:

Combines @Configuration, @EnableAutoConfiguration, and @ComponentScan.

- @Configuration indicates that the class has @Bean definition methods.
- @EnableAutoConfiguration enables auto-configuration of the Spring application context.
- @ComponentScan scans for components, configurations, and services in the specified package.

1. @RestController**:**Combines @Controller and @ResponseBody. It marks the class as a controller where @RequestMapping methods are used to handle web requests.
2. @Service**:**Indicates that a class is a service component in the business logic layer. Spring will handle it as a Spring-managed bean.
3. @Repository**:**Indicates that a class is a Data Access Object (DAO). It also helps in translating database-related exceptions into Spring's DataAccessException.
4. @Entity**:**Marks a class as a JPA entity. It will be mapped to a database table.
5. @Table**:**Specifies the name of the database table to which an entity is mapped.
6. @Id**:**Specifies the primary key of an entity.
7. @GeneratedValue**:**Defines the strategy for generating primary key values (e.g., auto increment).
8. @Column**:**Specifies the details of the column to which a field will be mapped.
9. @Autowired**:**Allows Spring to resolve and inject collaborating beans into your bean.
10. @Value**:**Injects values into fields from property files.
11. @Transactional**:**Defines the scope of a single database transaction.
12. @ExceptionHandler**:**Used in controllers to handle exceptions and return custom error responses.

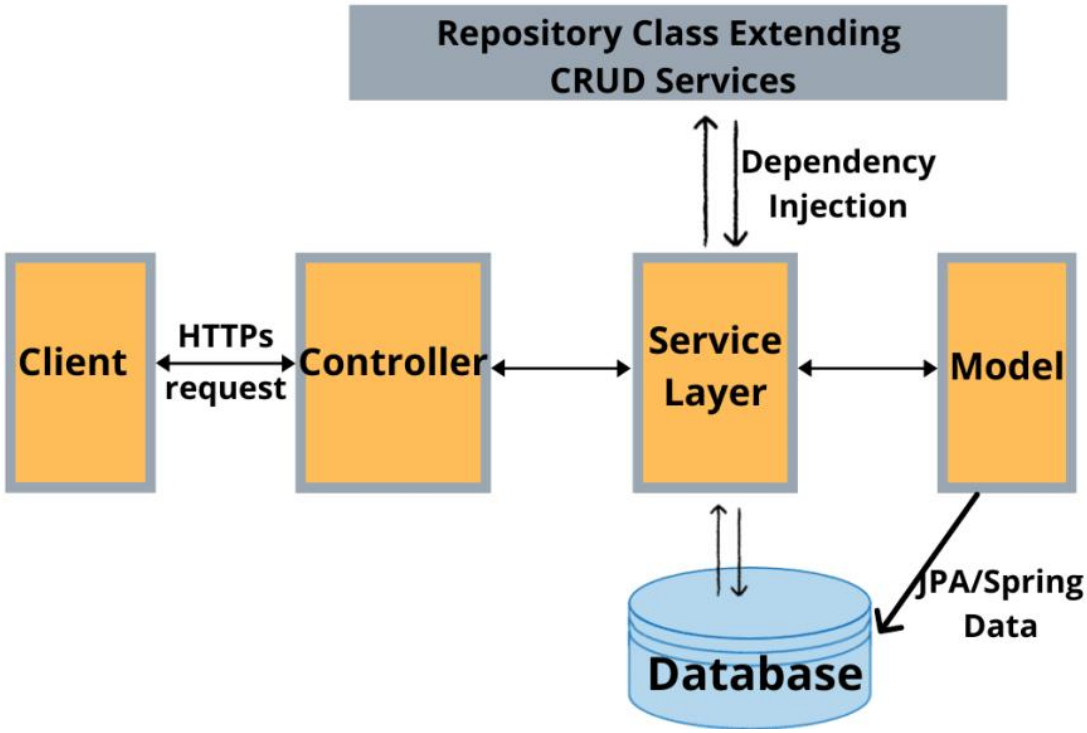
Spring Boot key components



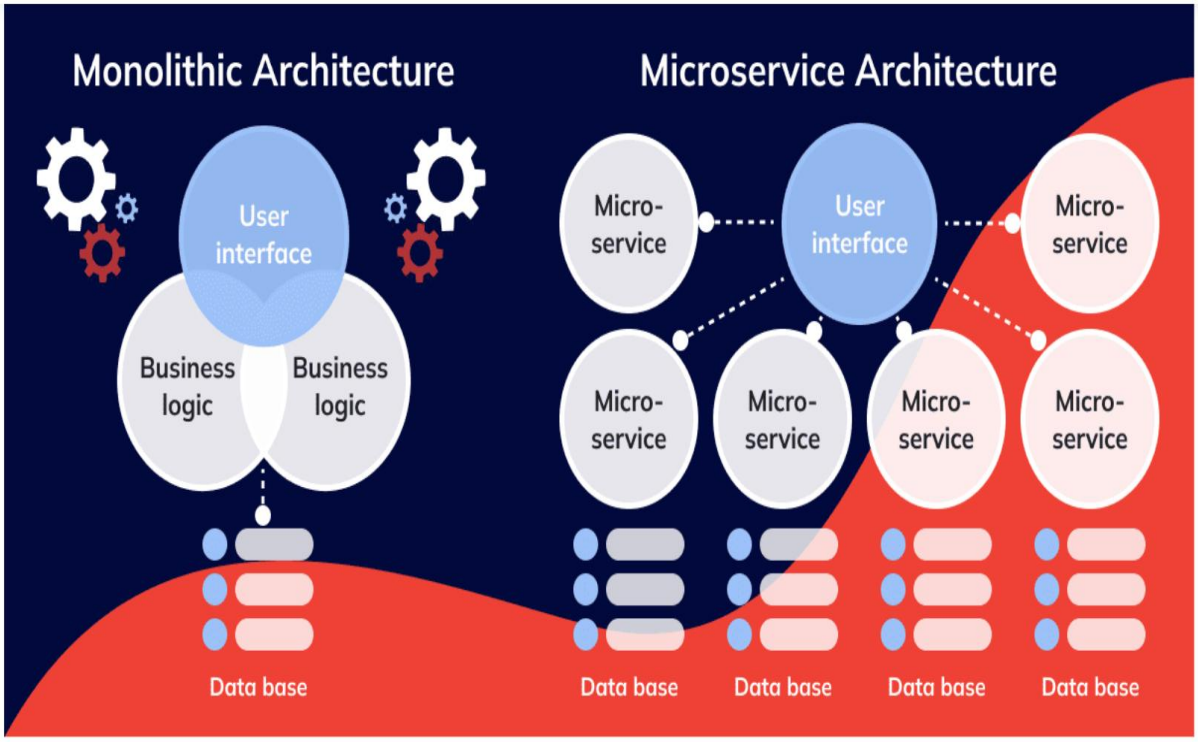
Different starters that are available in Spring Boot

Name	Description
spring-boot-starter-web	Build web applications and RESTful applications
spring-boot-starter-web- services	For building applications exposing SOAP web services
spring-boot-starter-test	Write great unit and integration tests
spring-boot-starter-jdbc	Traditional JDBC applications
spring-boot-starter-hateoas	Make your services more RESTful by adding HATEOAS features
spring-boot-starter-security	Authentication and authorization using Spring Security
spring-boot-starter-data- jpa	Spring Data JPA with Hibernate
spring-boot-starter-cache	Enabling the Spring Framework's caching support
spring-boot-starter-data- rest	Expose simple REST services using Spring Data REST

Spring Boot Flow Architecture



Monolithic Architecture Vs Microservice Architecture



Summary of differences: monolithic vs. microservices

Categories	Monolithic Architecture	Microservices Architecture
Design	Single, unified structure	Disassembled into smaller, independent units
Development	Simple, all functions managed in one place	Modular, each service handles specific functions independently
Deployment	Deployment as a single unit	Independently deployable modules
Debugging	Easier debugging as it operates as a single unit	More complex debugging due to distributed nature
Modification	Changes affect entire stack	Changes can be made to individual services without affecting others
Investment	Lower initial investment, simpler infrastructure	Higher initial investment, but offers scalability and flexibility benefits in the long run

Postman

Summary of All Endpoints

1. Add Account (POST)

Method: POST

URL: http://localhost:8080/api/accounts

Headers: Content-Type: application/json

Body:json

```
Code { "first_name": "Deepti", "last_name": "Wani", "dob": "2003-09-15", "accno": "123456789", "branch": "Main Branch", "address_line1": "123 Main St", "city": "San", "state": "CA", "pincode": 90001 }
```

2. Get Account by Customer ID (GET)

Method: GET

URL: http://localhost:8080/api/accounts/{cust_id}

3. Get Account by Account Number (GET)

Method: GET

URL: http://localhost:8080/api/accounts/account/{accno}

4. Get All Accounts (GET)

Method: GET

URL: http://localhost:8080/api/accounts

5. Update Account (PUT)

Method: PUT

URL: http://localhost:8080/api/accounts/{cust_id}

Headers: Content-Type: application/json

Body: Full account details JSON.

6. Partially Update Account (PATCH)

Method: PATCH

URL: http://localhost:8080/api/accounts/{cust_id}

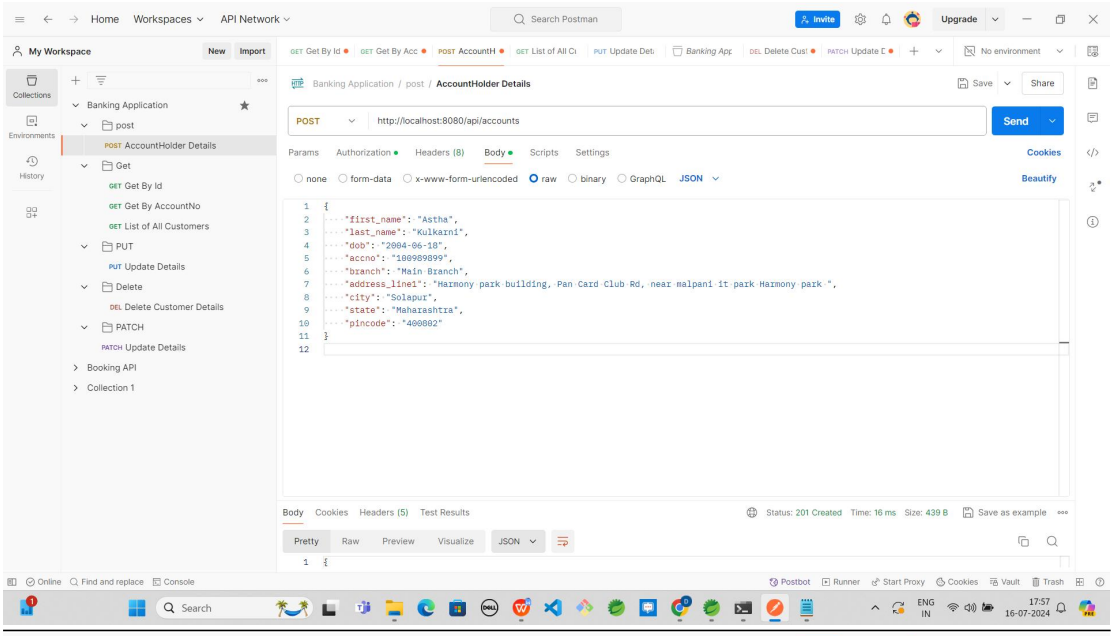
Headers: Content-Type: application/json

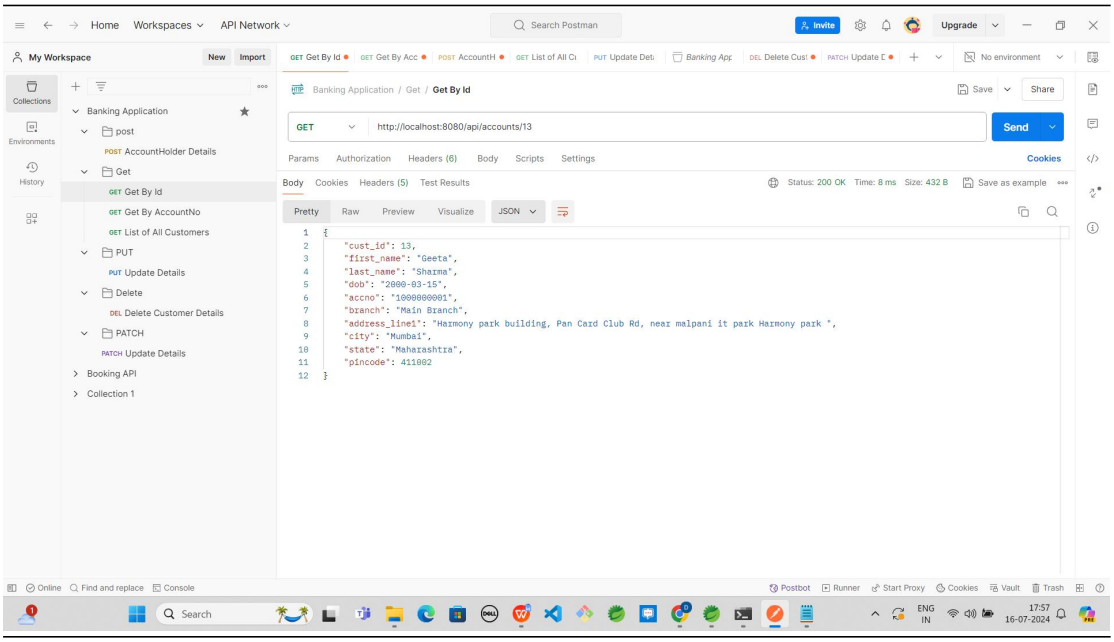
Body: JSON with fields to update.

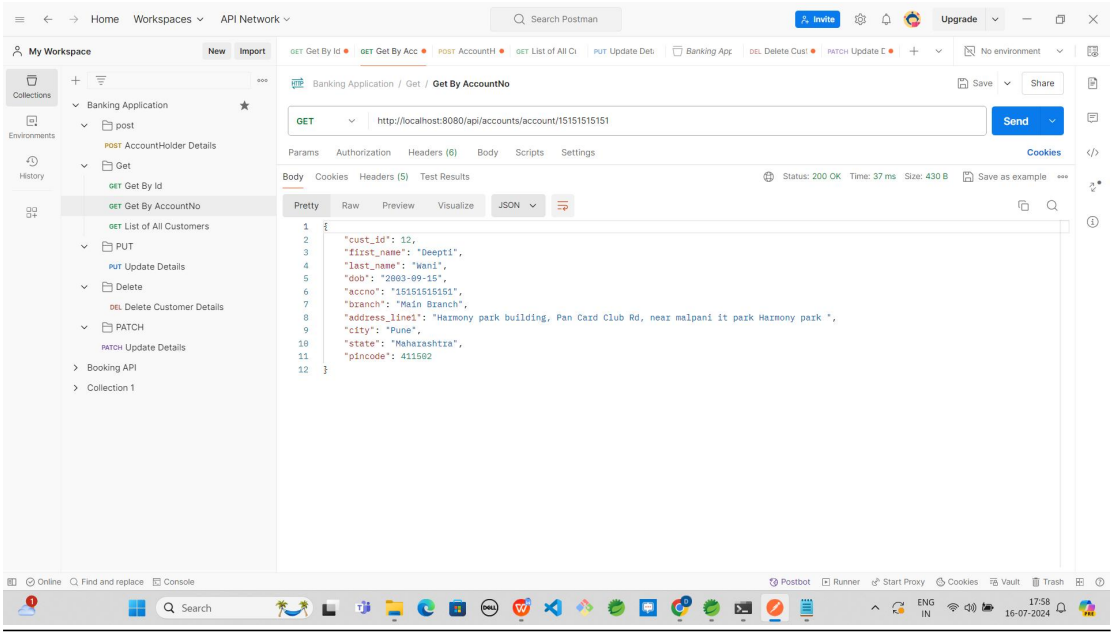
7. Delete Account (DELETE)

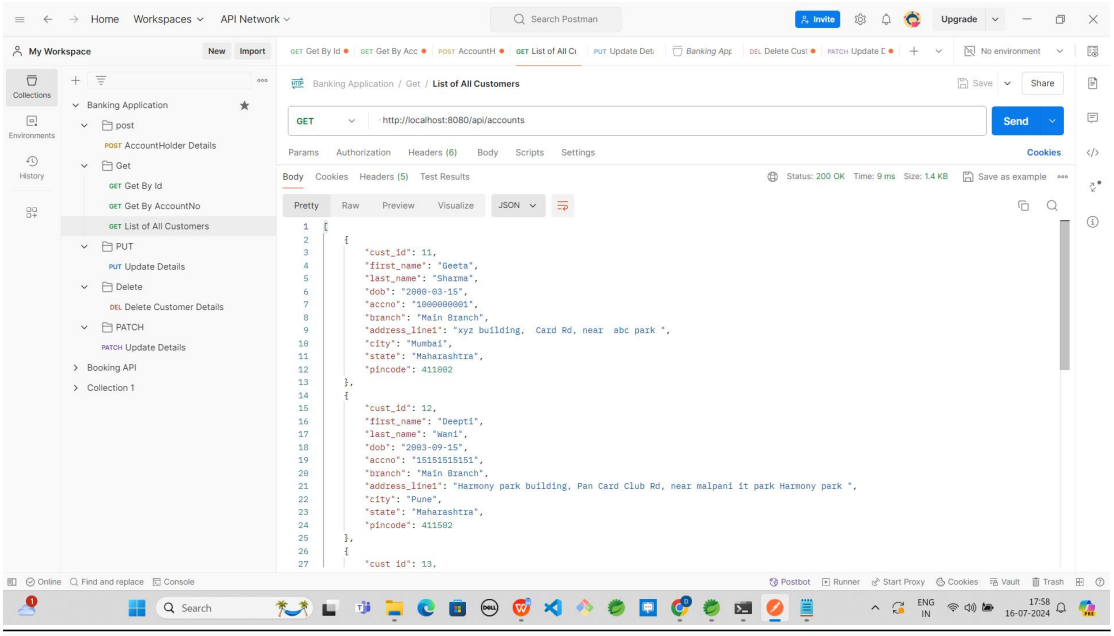
Method: DELETE

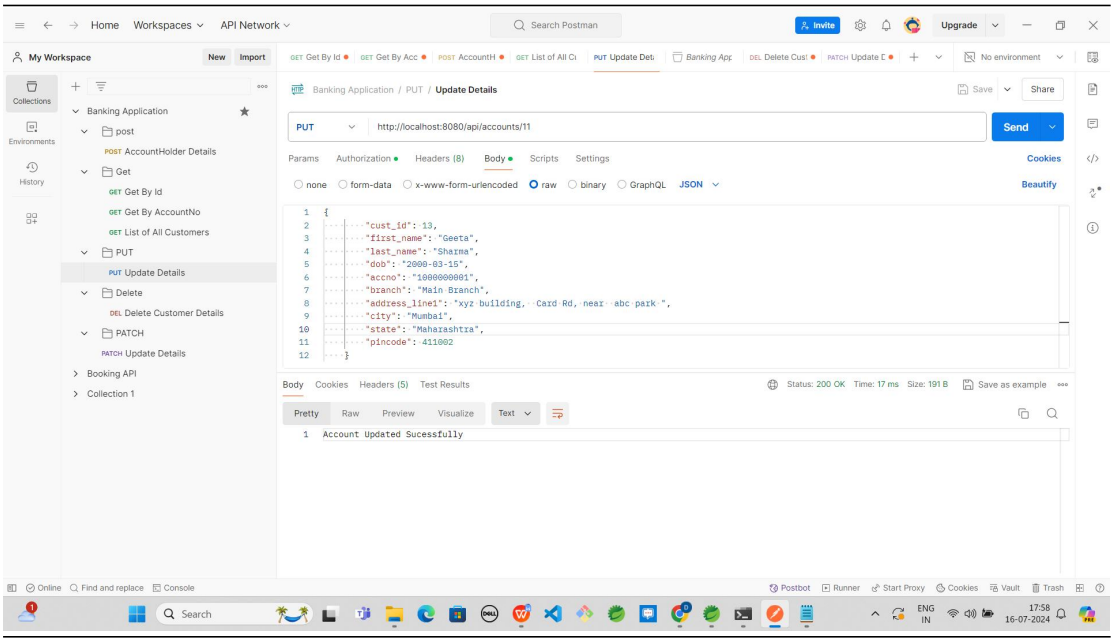
URL: http://localhost:8080/api/accounts/{cust_id}

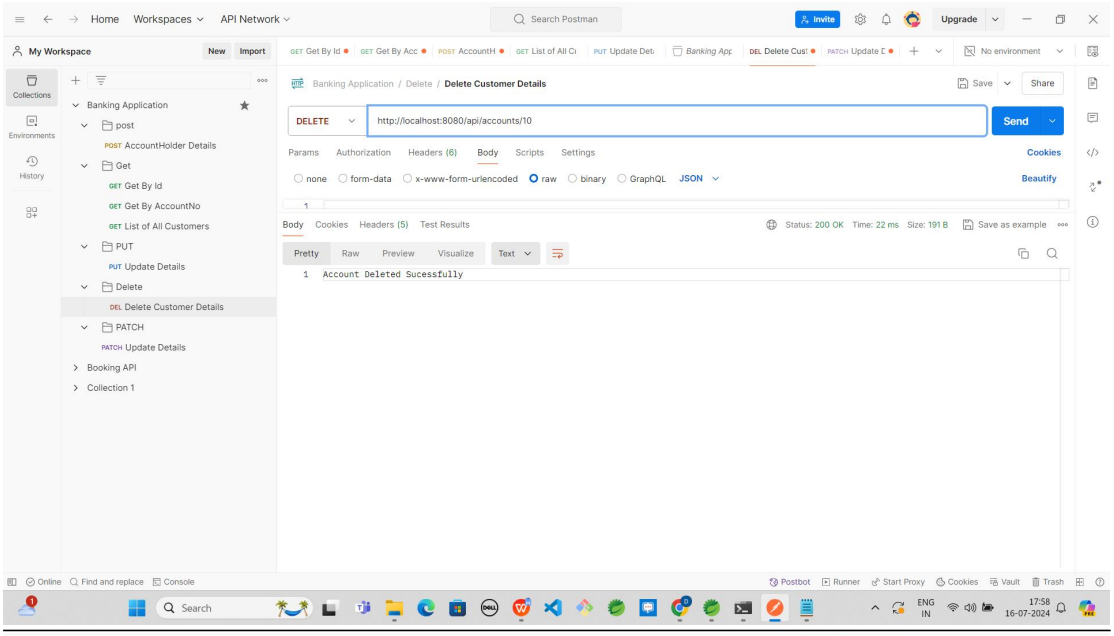

Home
Workspaces
API Network
Search Postman
Invite
Upgrade
My Workspace
New
Import
Collections
Environments
History
Banking Application
post
POST AccountHolder Details
Get
GET Get By Id
GET Get By AccountNo
GET List of All Customers
PUT
PUT Update Details
Delete
DEL Delete Customer Details
PATCH
PATCH Update Details
Booking API
Collection 1
Banking Application / post / AccountHolder Details
Save
Share
POST
http://localhost:8080/api/accounts
Send
Params
Authorization
Headers (8)
Body
Scripts
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"first_name": "Astha",
"last_name": "Kulkarni",
"dob": "2004-06-18",
"accno": "100969899",
"branch": "Main Branch",
"address_line1": "Harmony park building, Pan Card Club Rd, near malpani it park Harmony park ",
"city": "Solapur",
"state": "Maharashtra",
"pincode": "400002"
}
Body
Cookies
Headers (5)
Test Results
Status: 201 Created
Time: 16 ms
Size: 439 B
Save as example
Pretty
Raw
Preview
Visualize
JSON
Online
Find and replace
Console
Postbot
Runner
Start Proxy
Cookies
Vault
Trash
Search
ENG
IN
17:57
16-07-2024

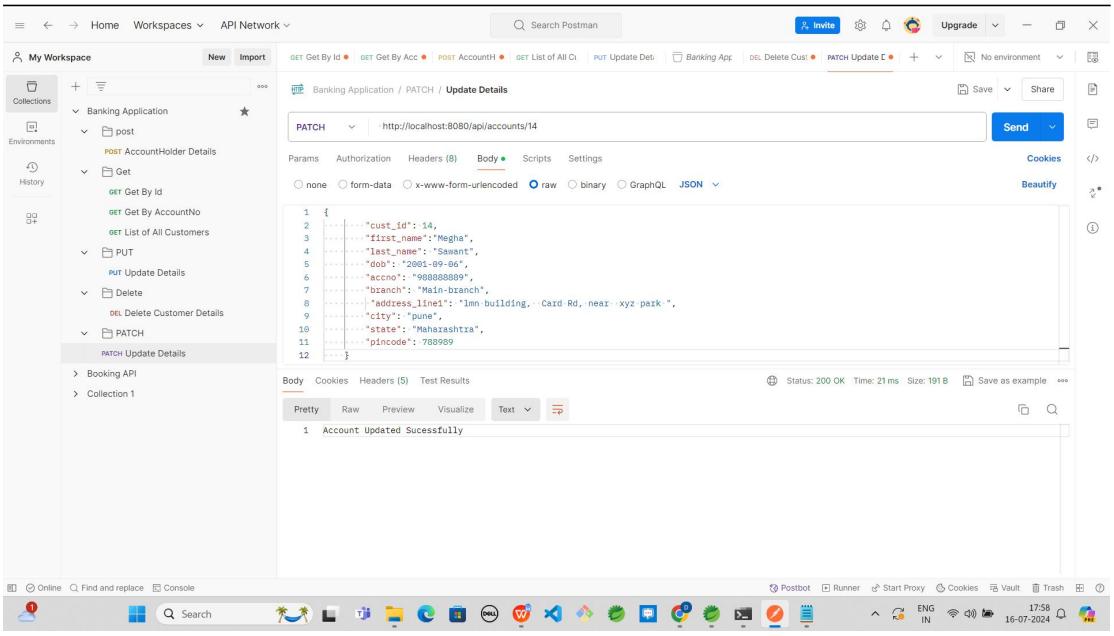

Home
Workspaces
API Network
Search Postman
Invite
Upgrade
My Workspace
New
Import
Collections
Environments
History
Banking Application
post
POST AccountHolder Details
Get
GET Get By Id
GET Get By AccountNo
GET List of All Customers
PUT
PUT Update Details
Delete
DEL Delete Customer Details
PATCH
PATCH Update Details
Booking API
Collection 1
Banking Application / Get / Get By Id
Save
Share
GET
http://localhost:8080/api/accounts/13
Send
Params
Authorization
Headers (6)
Body
Scripts
Settings
Cookies
Status: 200 OK
Time: 8 ms
Size: 432 B
Save as example
Pretty
Raw
Preview
Visualize
JSON
{
"cust_id": 13,
"first_name": "Geeta",
"last_name": "Sharma",
"dob": "2000-03-15",
"accno": "1000000001",
"branch": "Main Branch",
"address_line1": "Harmony park building, Pan Card Club Rd, near malpani it park Harmony park ",
"city": "Mumbai",
"state": "Maharashtra",
"pincode": 411002
}
Online
Find and replace
Console
Postbot
Runner
Start Proxy
Cookies
Vault
Trash
Search
ENG
IN
17:57
16-07-2024


Home
Workspaces
API Network
Search Postman
Invite
Upgrade
My Workspace
New
Import
Collections
Environments
History
Banking Application
post
POST AccountHolder Details
Get
GET Get By Id
GET Get By AccountNo
GET List of All Customers
PUT
PUT Update Details
Delete
DEL Delete Customer Details
PATCH
PATCH Update Details
Booking API
Collection 1
Banking Application / Get / Get By AccountNo
Save
Share
GET
http://localhost:8080/api/accounts/account/15151515151
Send
Params
Authorization
Headers (6)
Body
Scripts
Settings
Cookies
Body
Cookies
Headers (5)
Test Results
Status: 200 OK
Time: 37 ms
Size: 430 B
Save as example
Pretty
Raw
Preview
Visualize
JSON
{
"cust_id": 12,
"first_name": "Deepti",
"last_name": "Wani",
"dob": "2003-09-15",
"accno": "15151515151",
"branch": "Main Branch",
"address_line1": "Harmony park building, Pan Card Club Rd, near malpani it park Harmony park ",
"city": "Pune",
"state": "Maharashtra",
"pincode": 411502
}
Online
Find and replace
Console
Postbot
Runner
Start Proxy
Cookies
Vault
Trash
Search
ENG
IN
17:58
16-07-2024


Home
Workspaces
API Network
Search Postman
Invite
Upgrade
My Workspace
New
Import
Banking Application / Get / List of All Customers
Save
Share
GET
http://localhost:8080/api/accounts
Send
Params
Authorization
Headers (6)
Body
Scripts
Settings
Cookies
Body
Cookies
Headers (5)
Test Results
Status: 200 OK
Time: 9 ms
Size: 1.4 KB
Save as example
Pretty
Raw
Preview
Visualize
JSON
Banking Application
post
POST AccountHolder Details
Get
GET Get By Id
GET Get By AccountNo
GET List of All Customers
PUT
PUT Update Details
Delete
DEL Delete Customer Details
PATCH
PATCH Update Details
Booking API
Collection 1
[
{
"cust_id": 11,
"first_name": "Geeta",
"last_name": "Sharma",
"dob": "2000-03-15",
"accno": "1000000001",
"branch": "Main Branch",
"address_line1": "xyz building, Card Rd, near abc park ",
"city": "Mumbai",
"state": "Maharashtra",
"pincode": 411002
},
{
"cust_id": 12,
"first_name": "Deepti",
"last_name": "Wani",
"dob": "2003-09-15",
"accno": "15151515151",
"branch": "Main Branch",
"address_line1": "Harmony park building, Pan Card Club Rd, near malpani it park Harmony park ",
"city": "Pune",
"state": "Maharashtra",
"pincode": 411502
},
{
"cust_id": 13,
Online
Find and replace
Console
Postbot
Runner
Start Proxy
Cookies
Vault
Trash
Search
ENG
IN
17:58
16-07-2024


Home
Workspaces
API Network
Search Postman
Invite
Upgrade
My Workspace
New
Import
Banking Application / PUT / Update Details
Save
Share
PUT
http://localhost:8080/api/accounts/11
Send
Params
Authorization
Headers (8)
Body
Scripts
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"cust_id": 13,
"first_name": "Geeta",
"last_name": "Sharma",
"dob": "2000-03-15",
"accno": "1000000001",
"branch": "Main Branch",
"address_line1": "xyz building, Card Rd, near abc park ",
"city": "Mumbai",
"state": "Maharashtra",
"pincode": 411002
}
Body
Cookies
Headers (5)
Test Results
Status: 200 OK
Time: 17 ms
Size: 191 B
Save as example
Pretty
Raw
Preview
Visualize
Text
Account Updated Sucessfully
Banking Application
post
POST AccountHolder Details
Get
GET Get By Id
GET Get By AccountNo
GET List of All Customers
PUT
PUT Update Details
Delete
DEL Delete Customer Details
PATCH
PATCH Update Details
Booking API
Collection 1
Online
Find and replace
Console
Postbot
Runner
Start Proxy
Cookies
Vault
Trash
Search
ENG
IN
17:58
16-07-2024


Home
Workspaces
API Network
Search Postman
Invite
Upgrade
My Workspace
New
Import
Collections
Environments
History
Banking Application
post
POST AccountHolder Details
Get
GET Get By Id
GET Get By AccountNo
GET List of All Customers
PUT
PUT Update Details
Delete
DEL Delete Customer Details
PATCH
PATCH Update Details
Booking API
Collection 1
Banking Application / Delete / Delete Customer Details
Save
Share
DELETE
http://localhost:8080/api/accounts/10
Send
Params
Authorization
Headers (6)
Body
Scripts
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
Body
Cookies
Headers (5)
Test Results
Status: 200 OK
Time: 22 ms
Size: 191 B
Save as example
Pretty
Raw
Preview
Visualize
Text
1 Account Deleted Sucessfully
Online
Find and replace
Console
Postbot
Runner
Start Proxy
Cookies
Vault
Trash
Search
ENG
IN
17:58
16-07-2024


Home
Workspaces
API Network
Search Postman
Invite
Upgrade
My Workspace
New
Import
Collections
Environments
History
Banking Application
post
POST AccountHolder Details
Get
GET Get By Id
GET Get By AccountNo
GET List of All Customers
PUT
PUT Update Details
Delete
DEL Delete Customer Details
PATCH
PATCH Update Details
Booking API
Collection 1
Banking Application / PATCH / Update Details
Save
Share
PATCH
http://localhost:8080/api/accounts/14
Send
Params
Authorization
Headers (8)
Body
Scripts
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"cust_id": 14,
"first_name":"Megha",
"last_name": "Sawant",
"dob": "2001-09-06",
"accno": "908888809",
"branch": "Main-branch",
"address_line1": "lmn building, Card Rd, near xyz park ",
"city": "pune",
"state": "Maharashtra",
"pincode": 788989
}
Body
Cookies
Headers (5)
Test Results
Status: 200 OK
Time: 21 ms
Size: 191 B
Save as example
Pretty
Raw
Preview
Visualize
Text
1 Account Updated Sucessfully
Online
Find and replace
Console
Postbot
Runner
Start Proxy
Cookies
Vault
Trash
Search
ENG
IN
17:58
16-07-2024