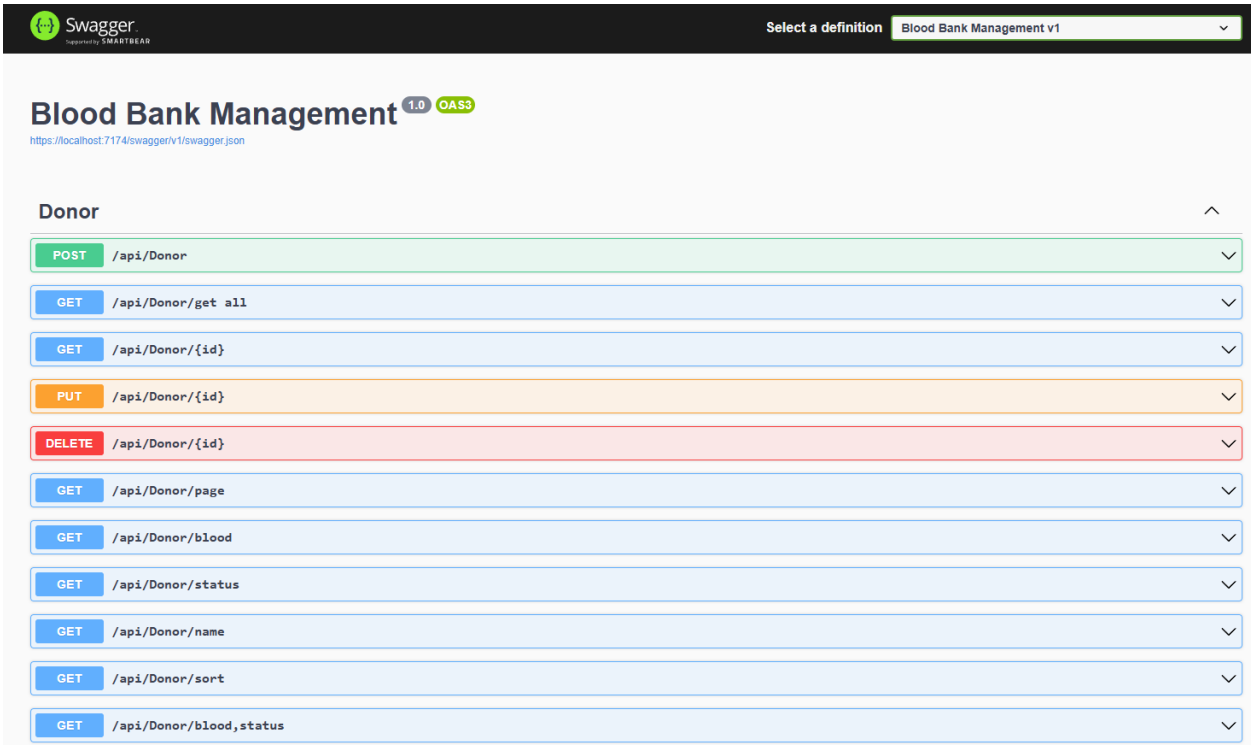
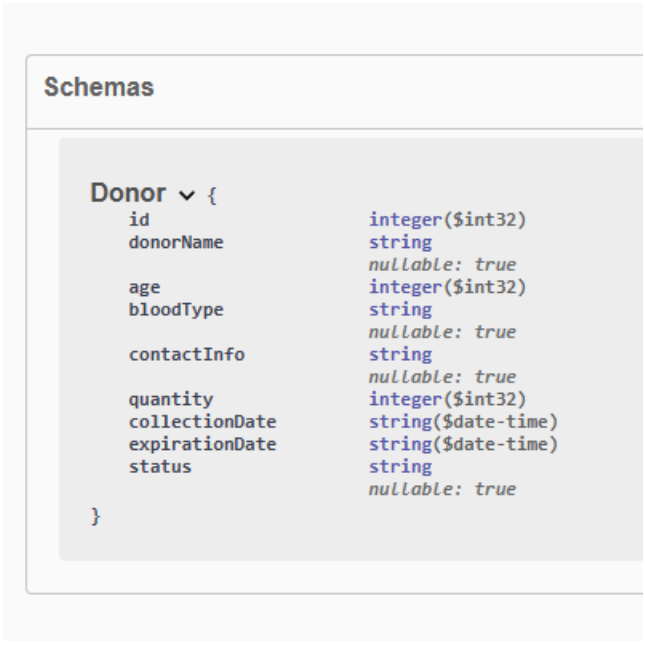
# Blood Bank Management REST API Assignment Explanation Docs




Schema (MODEL), (Class Attributes)

This is the **Swagger UI** interface. It provides a visual representation of the API endpoints, their HTTP methods (e.g., POST, GET, PUT, DELETE), and a way to interact with the API.

## 1. Create Donor (POST /api/donors)  endpoint

POST /api/Donor

This endpoint adds a new donor to the system. The input data undergoes strict validation:

- Donor Name
    - Cannot be null or empty.
    - *Error:* "Donor name is required."
- Age
    - Must be between 18 and 75.
    - *Error:* "Invalid age. Age should be between 18 and 75."
- Blood Type
    - Must be one of: A+, A-, B+, B-, AB+, AB-, O+, O-.
    - *Error:* "Invalid blood type."
- Contact Info
    - Must be a valid 10-digit number.
    - *Error:* "Contact info must be a valid 10-digit phone number."
- Quantity
    - Minimum of 200 mL required.
    - *Error:* "Please enter more than 200 mL."
- Expiration Date
    - Set to 42 days from CollectionDate.
    - Status must be Available, Requested, or Expired.
    - *Error:* "Invalid status."
- Donor ID
    - Automatically assigned as the next sequential ID.
- Response
- Success: Donor is added and returned.
- Error: 400 Bad Request with the validation issue.

Request URL

```
https://localhost:7174/api/Donor
```

Server response

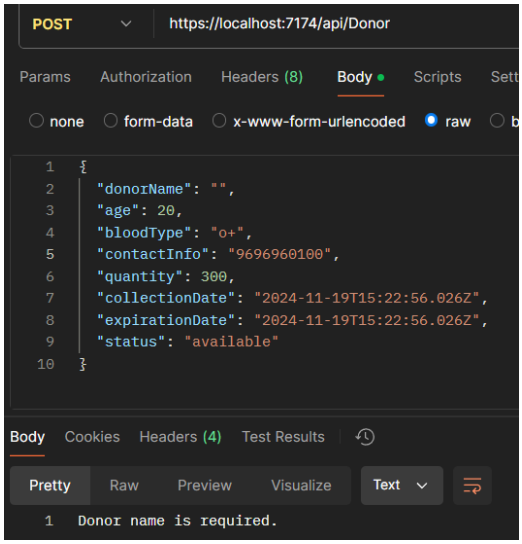
| Code | Details |
| --- | --- |
| 400 *Undocumented* | Error: response status is 400 |

Response body

```
Donor name is required.
```

Response headers

```
content-type: text/plain; charset=utf-8
date: Tue,19 Nov 2024 15:45:45 GMT
server: Kestrel
```




Request URL

```
https://localhost:7174/api/Donor
```

Server response

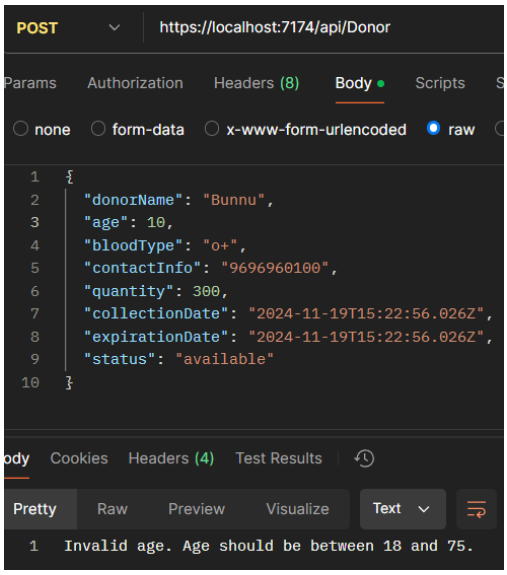
| Code | Details |
| --- | --- |
| 400 *Undocumented* | Error: response status is 400 |

Response body

```
Invalid age. Age should be between 18 and 75.
```

Response headers

```
content-type: text/plain; charset=utf-8
date: Tue,19 Nov 2024 15:37:59 GMT
server: Kestrel
```




Request URL

```
https://localhost:7174/api/Donor
```

Server response

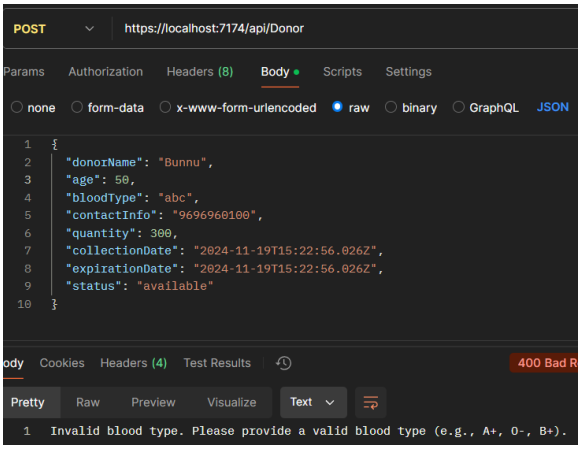
| Code | Details |
| --- | --- |
| 400 *Undocumented* | Error: response status is 400 |

Response body

```
Invalid blood type. Please provide a valid blood type (e.g., A+, O-, B+).
```

Response headers

```
content-type: text/plain; charset=utf-8
date: Tue,19 Nov 2024 15:46:54 GMT
server: Kestrel
```



Request URL

```
https://localhost:7174/api/Donor
```

Server response

| Code | Details |
| --- | --- |
| 400 *Undocumented* | Error: response status is 400 |

Response body

```
Contact info must be a valid 10-digit phone number.
```

Response headers

```
content-type: text/plain; charset=utf-8
date: Tue,19 Nov 2024 15:48:46 GMT
server: Kestrel
```


```
{
  "donorName": "Bunnu",
  "age": 50,
  "bloodType": "A+",
  "contactInfo": "96969601",
  "quantity": 300,
  "collectionDate": "2024-11-19T15:22:56.026Z",
  "expirationDate": "2024-11-19T15:22:56.026Z",
  "status": "available"
}
```


ody Cookies Headers (4) Test Results

Pretty Raw Preview Visualize Text

```
1  Contact info must be a valid 10-digit phone number.
```

Request URL

```
https://localhost:7174/api/Donor
```

Server response

| Code | Details |
| --- | --- |
| 400 *Undocumented* | Error: response status is 400 |

Response body

```
Please enter more than 200ml.
```


```
{
  "donorName": "Bunnu",
  "age": 50,
  "bloodType": "A+",
  "contactInfo": "9696960100",
  "quantity": 0,
  "collectionDate": "2024-11-19T15:22:56.026Z",
  "expirationDate": "2024-11-19T15:22:56.026Z",
  "status": "available"
}
```


ody Cookies Headers (4) Test Results

Pretty Raw Preview Visualize Text

```
1  Please enter more than 200ml.
```

400 *Undocumented* Error: response status is 400

Response body

```
Invalid status. Please provide a valid status (Available, Requested, or Expired).
```

Body Cookies Headers (4) Test Results 400 Bad Request 48 ms

Pretty Raw Preview Visualize Text

```
1  Invalid status. Please provide a valid status (Available, Requested, or Expired).
```

Request body


```
{
  "id": 0,
  "donorName": "sri",
  "age": 20,
  "bloodType": "o+",
  "contactInfo": "9696960100",
  "quantity": 300,
  "collectionDate": "2024-11-19T15:22:56.026Z",
  "expirationDate": "2024-11-19T15:22:56.026Z",
  "status": "available"
}
```


Request URL

```
https://localhost:7174/api/Donor
```

Server response

| Code | Details |
| --- | --- |
| 201 *Undocumented* | Response body |


```
[
  {
    "id": 1,
    "donorName": "Srijith",
    "age": 20,
    "bloodType": "O+",
    "contactInfo": "123123",
    "quantity": 500,
    "collectionDate": "2024-11-19T21:00:45.7857695+05:30",
    "expirationDate": "2024-12-31T21:00:45.7866727+05:30",
    "status": "Available"
  },
  {
    "id": 2,
    "donorName": "sri",
    "age": 20,
    "bloodType": "o+",
    "contactInfo": "9696960100",
    "quantity": 300,
    "collectionDate": "2024-11-19T15:22:56.026Z",
    "expirationDate": "2024-12-31T15:22:56.026Z",
    "status": "available"
  }
]
```



```
{
  "donorName": "Bunnu",
  "age": 50,
  "bloodType": "A+",
  "contactInfo": "9696960100",
  "quantity": 500,
  "collectionDate": "2024-11-19T15:22:56.026Z",
  "expirationDate": "2024-11-19T15:22:56.026Z",
  "status": "Requested"
}
```

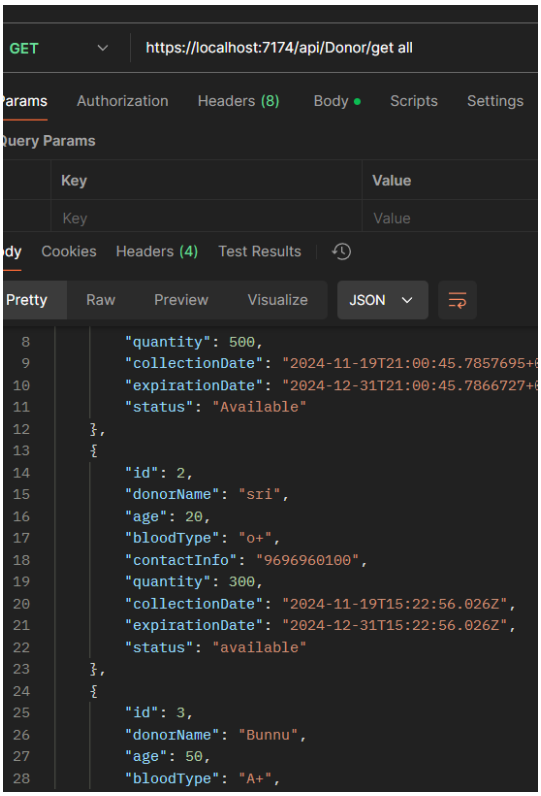

dy Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

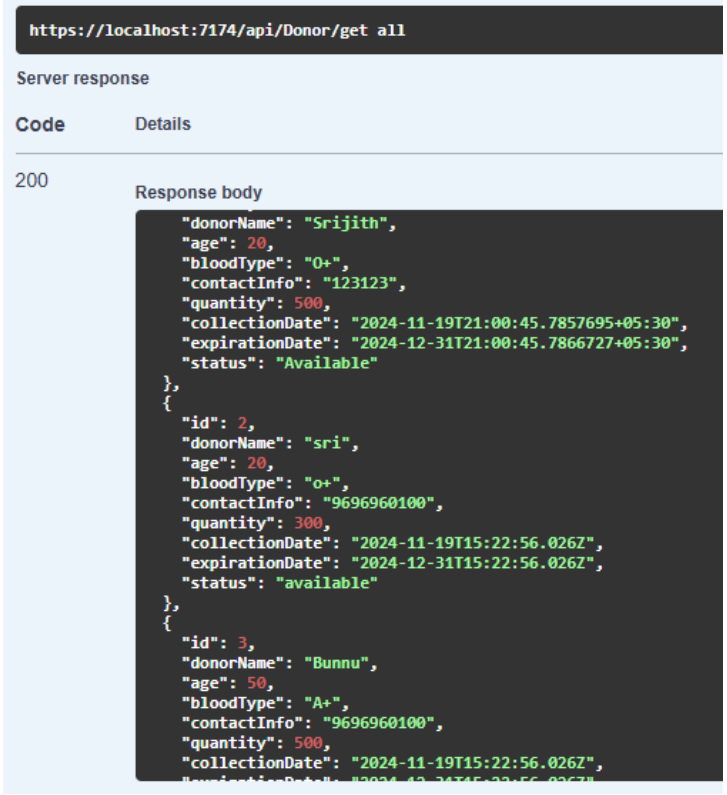

```
[
    {
        "id": 3,
        "donorName": "Bunnu",
        "age": 50,
        "bloodType": "A+",
        "contactInfo": "9696960100",
        "quantity": 500,
        "collectionDate": "2024-11-19T15:22:56.026Z",
        "expirationDate": "2024-12-31T15:22:56.026Z",
        "status": "Requested"
    }
]
```

## GET /api/Donor/get_all

This endpoint retrieves a list of all donors from the system.



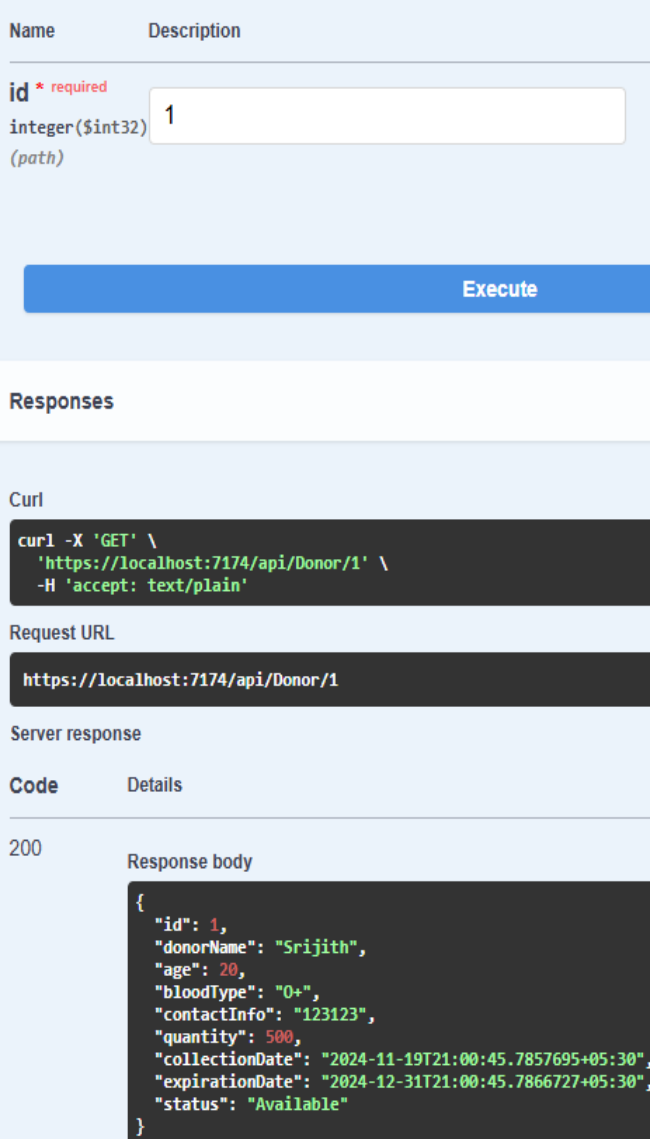

GET in POSTMAN API

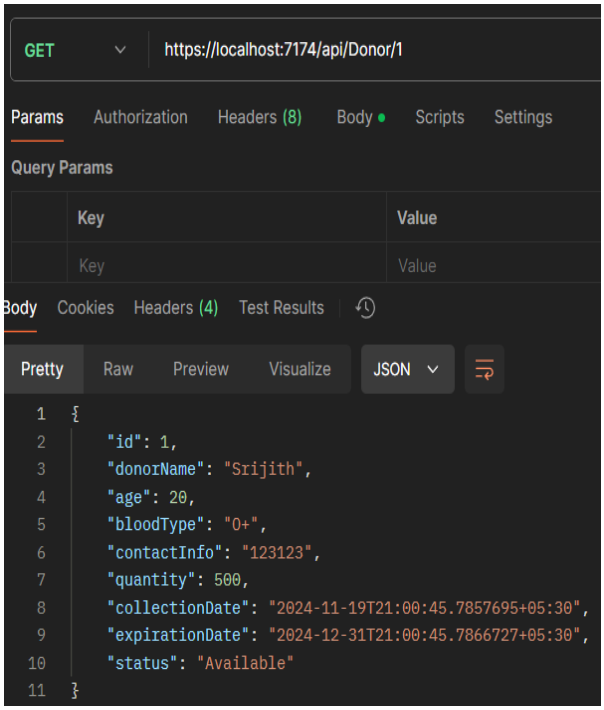



GET in SWAGGER

**Retrieves the details of a specific donor using their unique ID.**

- Validation:
    - Ensures the provided id exists in the system.
    - Returns 404 Not Found if the donor is not found.

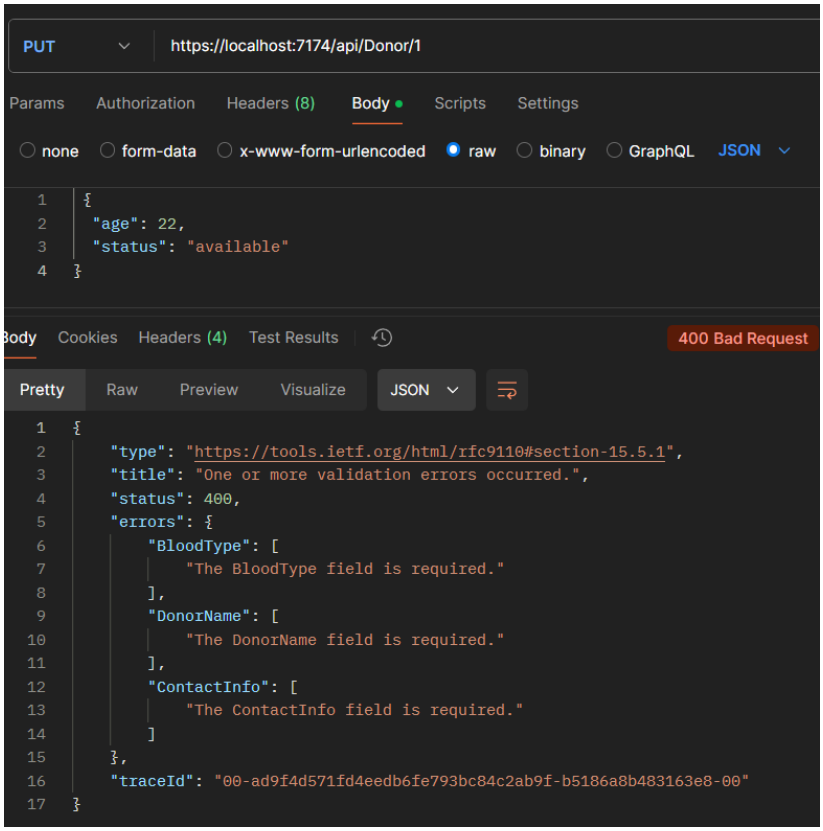


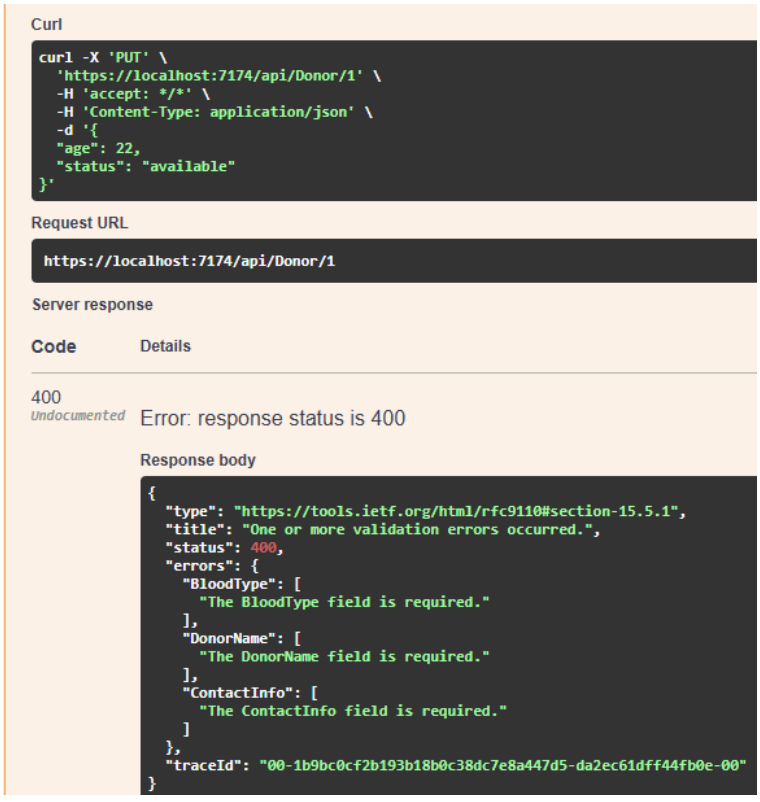



**PUT /api/Donor/{id}**

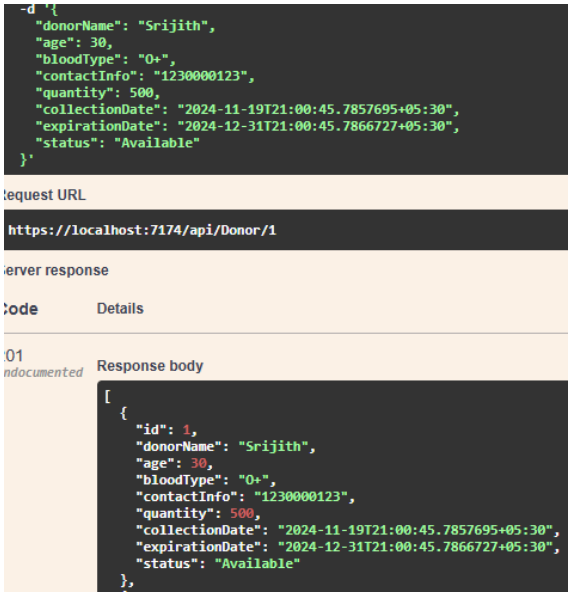
Updates the details of an existing donor.

- **Validation:**
  - Verifies if the id exists in the system.
  - Checks for all input validations, similar to the POST method (e.g., valid name, age, blood type, etc.).
  - Returns 400 Bad Request for invalid data.
- **Necessity:** Use this to modify details like contact info, blood quantity, or status.



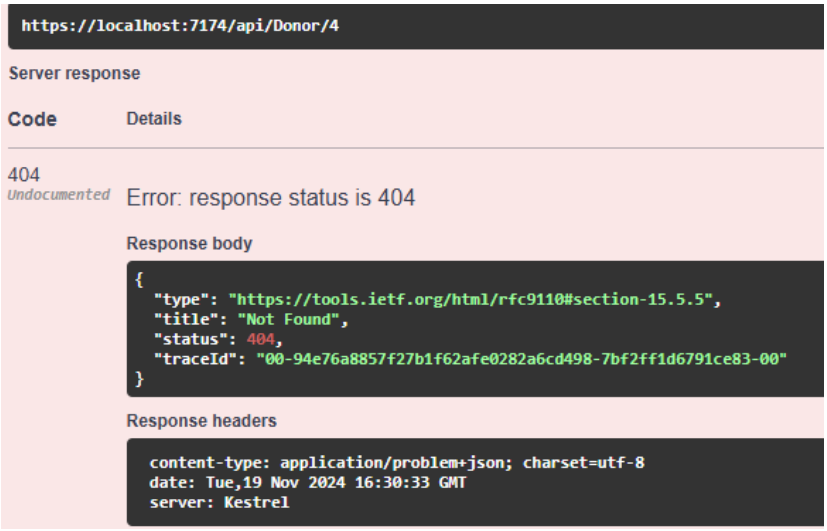

Success Case ==➔

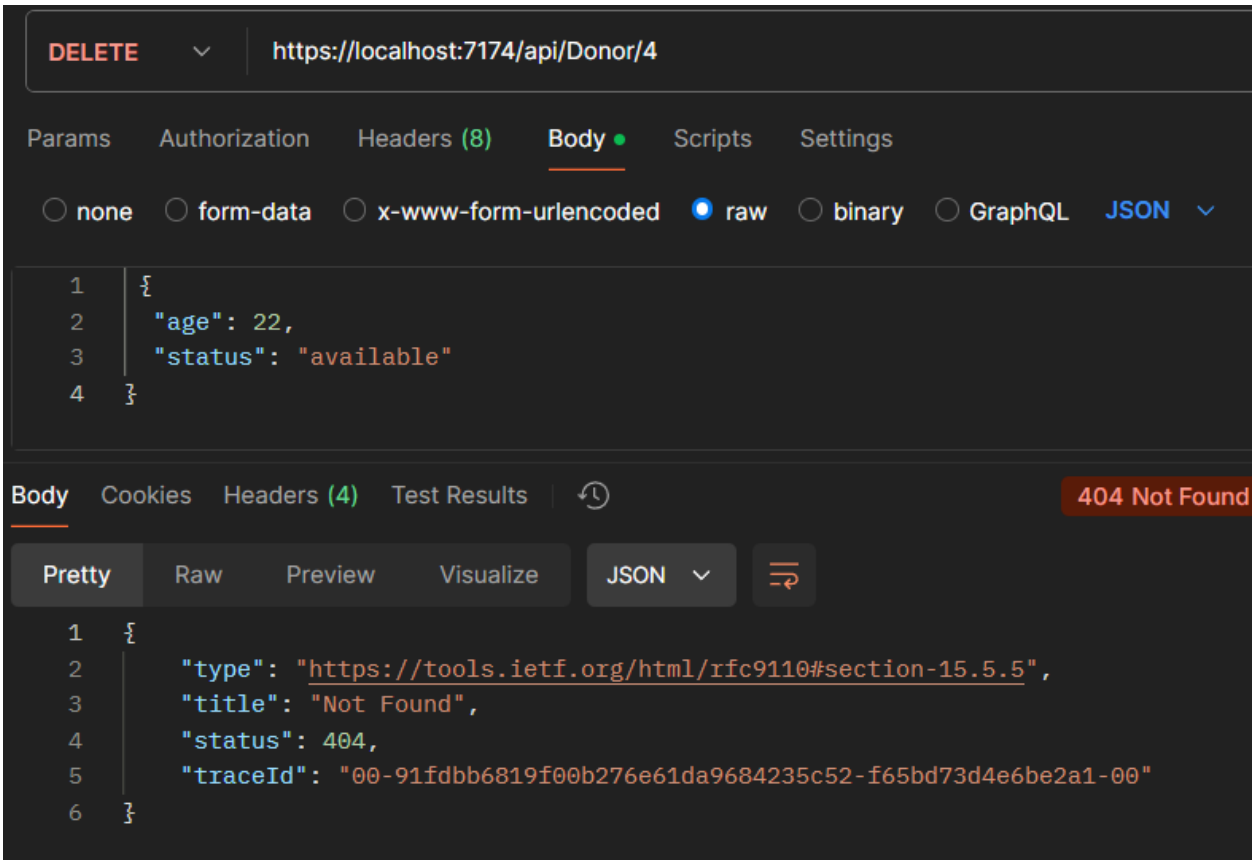


**DELETE /api/Donor/{id}**

Deletes a donor record by their unique ID.

- **Validation:**
    - Ensures the donor exists before attempting deletion.
    - Returns 404 Not Found if the donor is not found.
- **Necessity:** Useful for data cleanup or removing inactive donors.

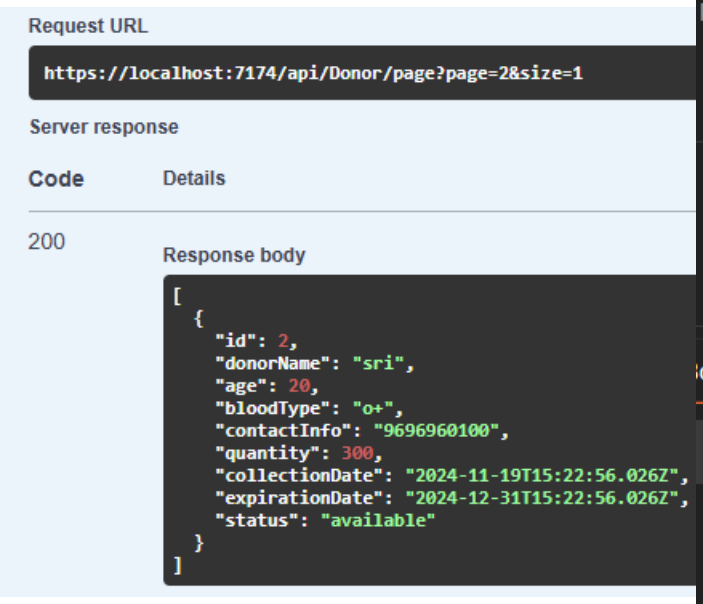


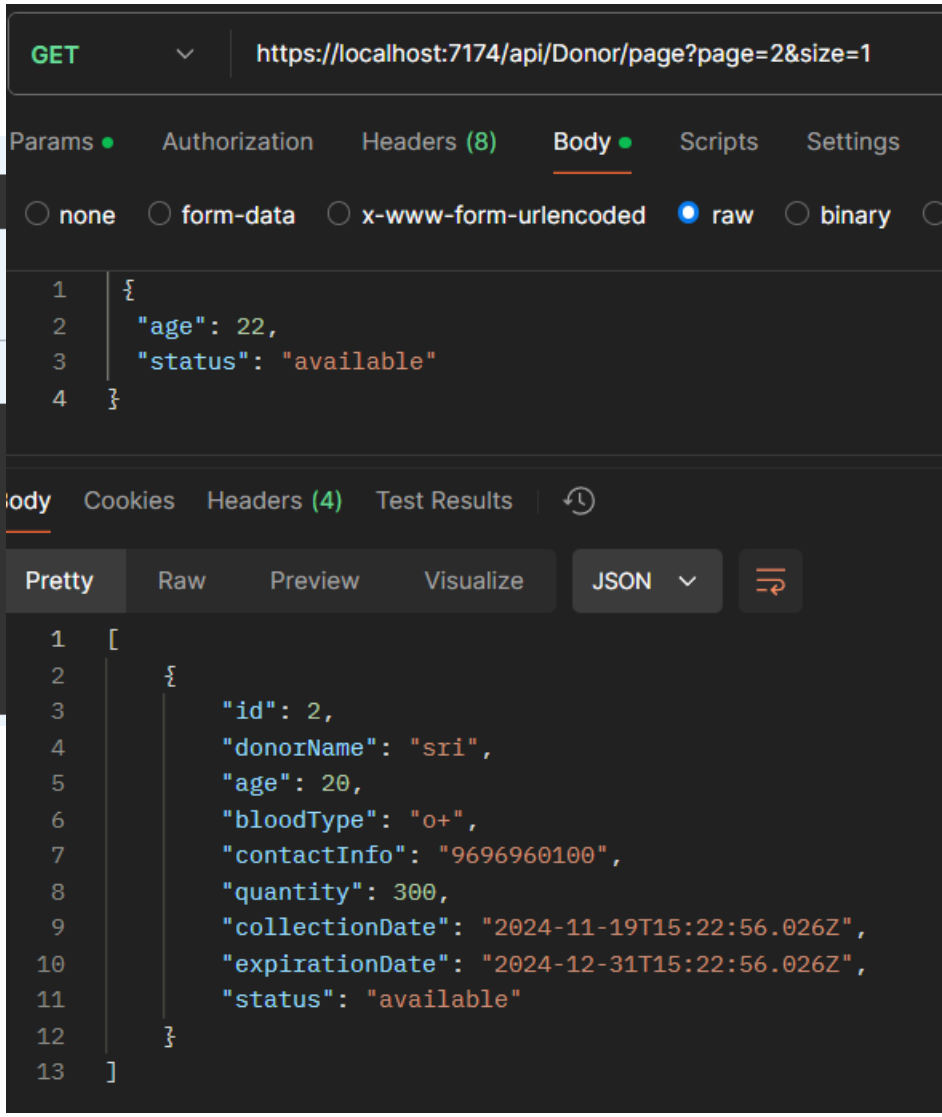



**GET /api/Donor/page**

Retrieves a paginated list of donors.

- **Validation:**
  - Ensures valid pagination parameters are provided (e.g., page, size).
  - Defaults to a specific range if parameters are missing.



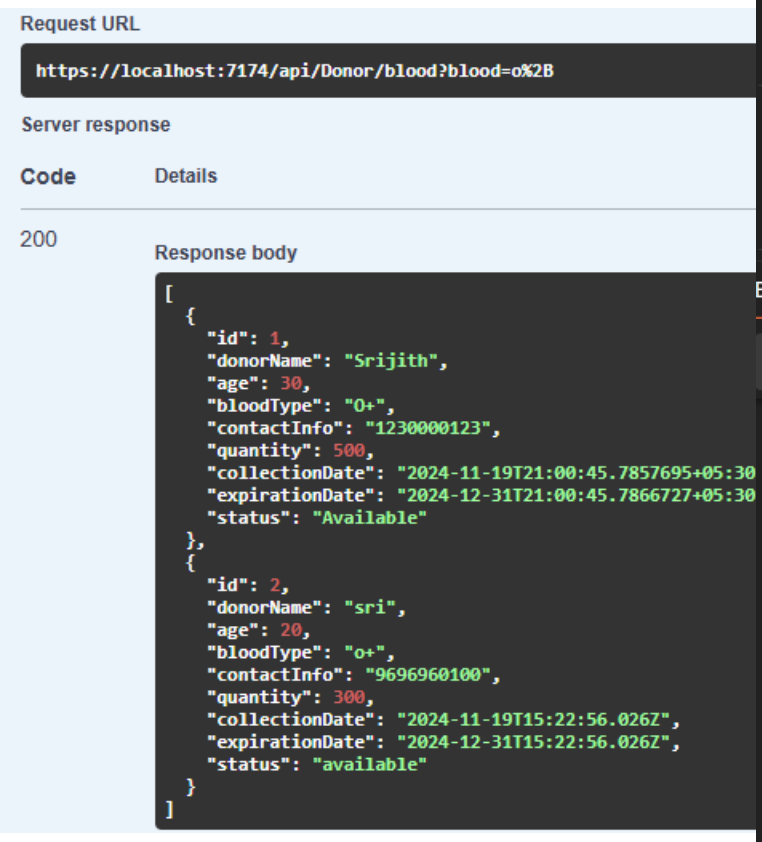



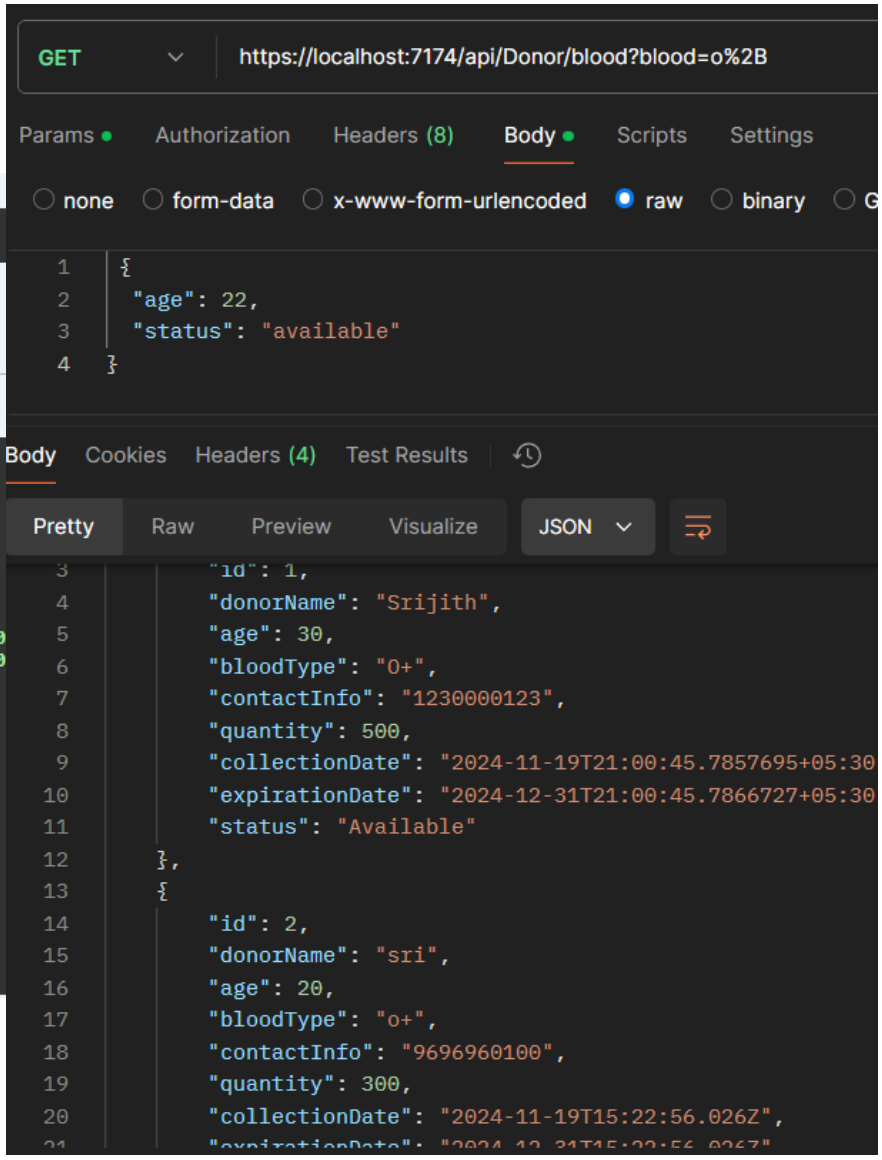
**GET /api/Donor/blood**

Filters donors by blood type.

- **Validation:**
  - Ensures the provided blood type is valid (e.g., A+, B-, O-, etc.).
  - Returns an empty list if no donors match the filter.

Example :- blood type = “O+”



## GET /api/Donor/status

Filters donors by their status (e.g., Available, Requested, Expired).

- **Validation:**
  - Ensures the provided status is one of the allowed values.

Example input :- Wanted  (wrong input)

Request URL

```
https://localhost:7174/api/Donor/status?status=wanted
```

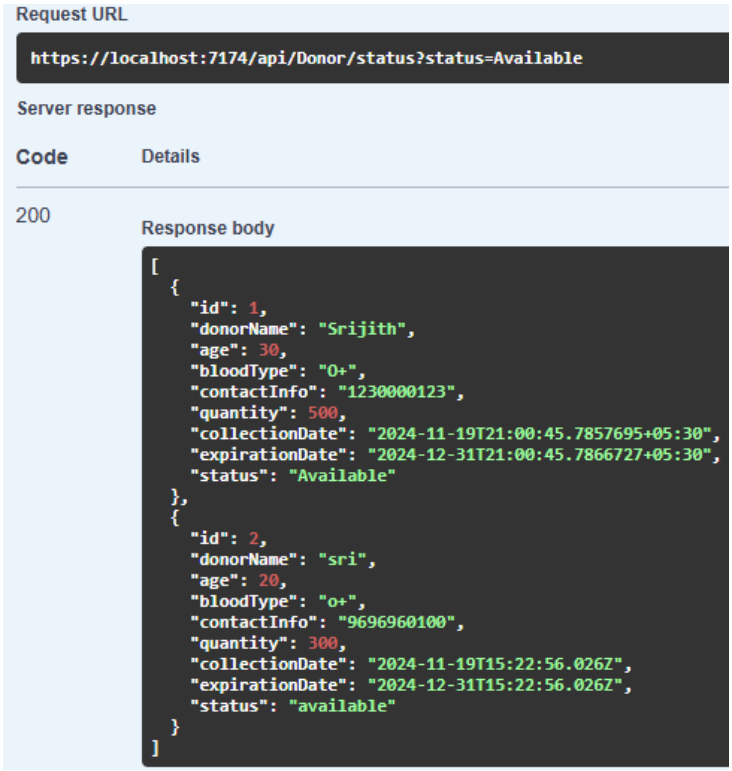
Server response

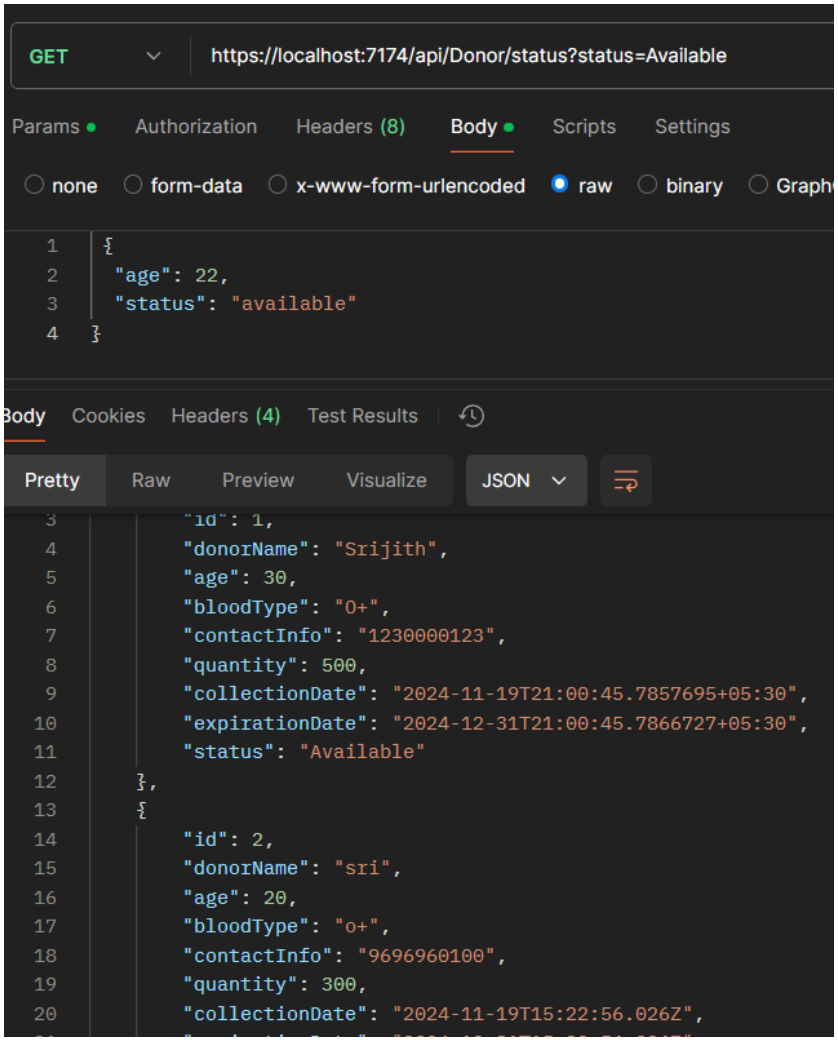
| Code | Details |
|---|---|
| 400 *Undocumented* | Error: response status is 400 |

Response body

```
Invalid status. Please provide a valid status (Available, Requested, or Expired).
```

Input :- Available

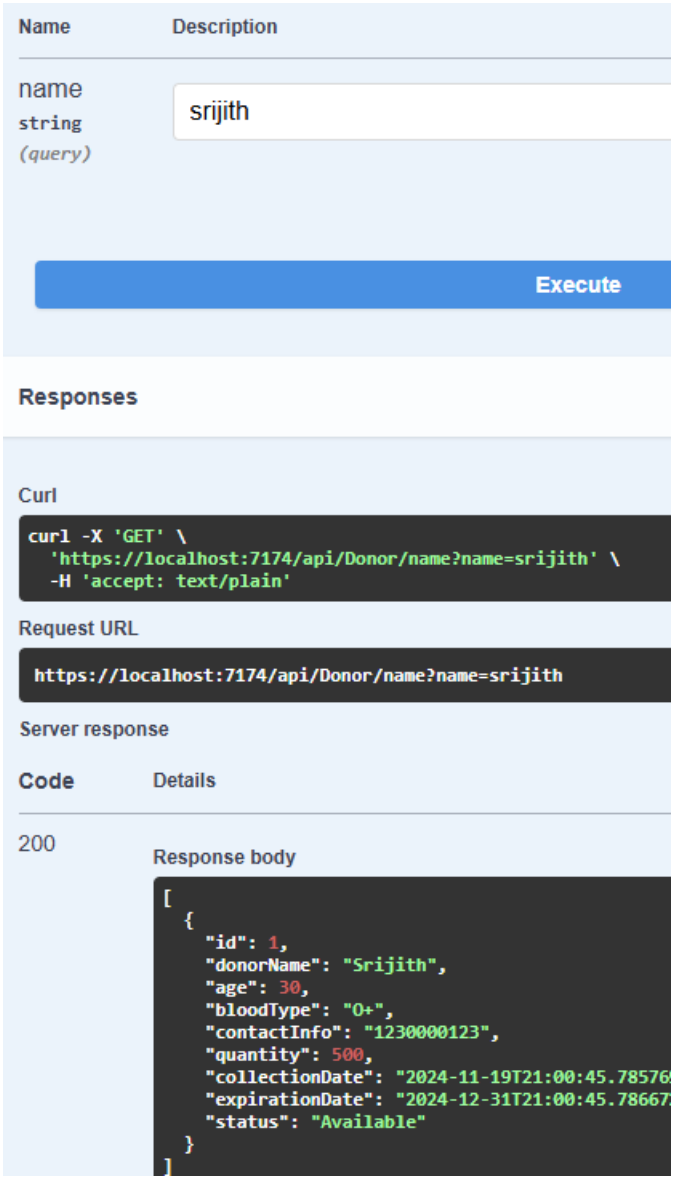


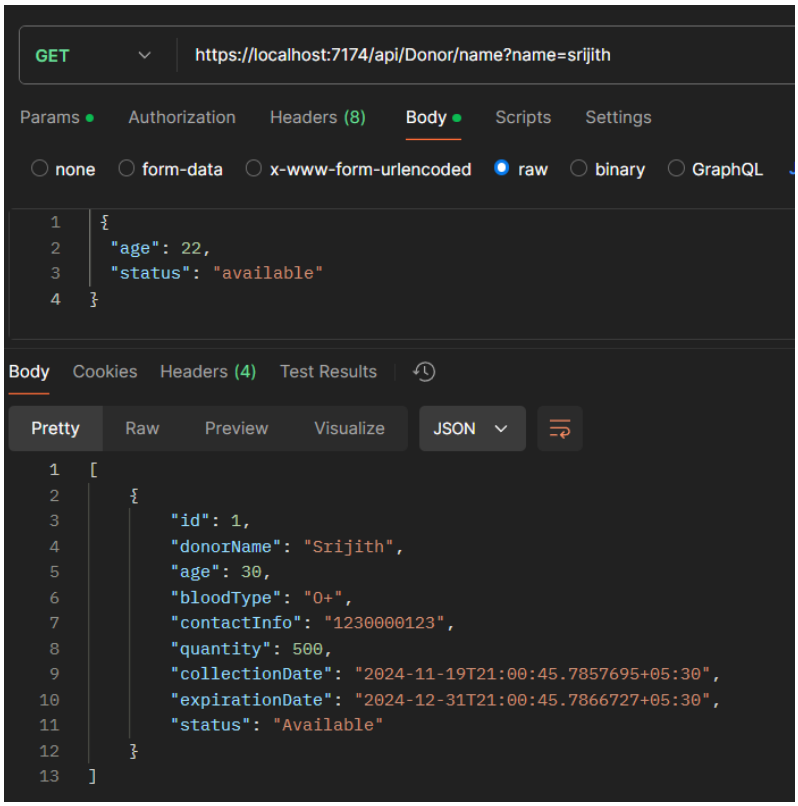



## GET /api/Donor/name

Retrieves donors filtered by their name or part of their name.

- **Validation:**
  - Ensures the name parameter is non-empty.

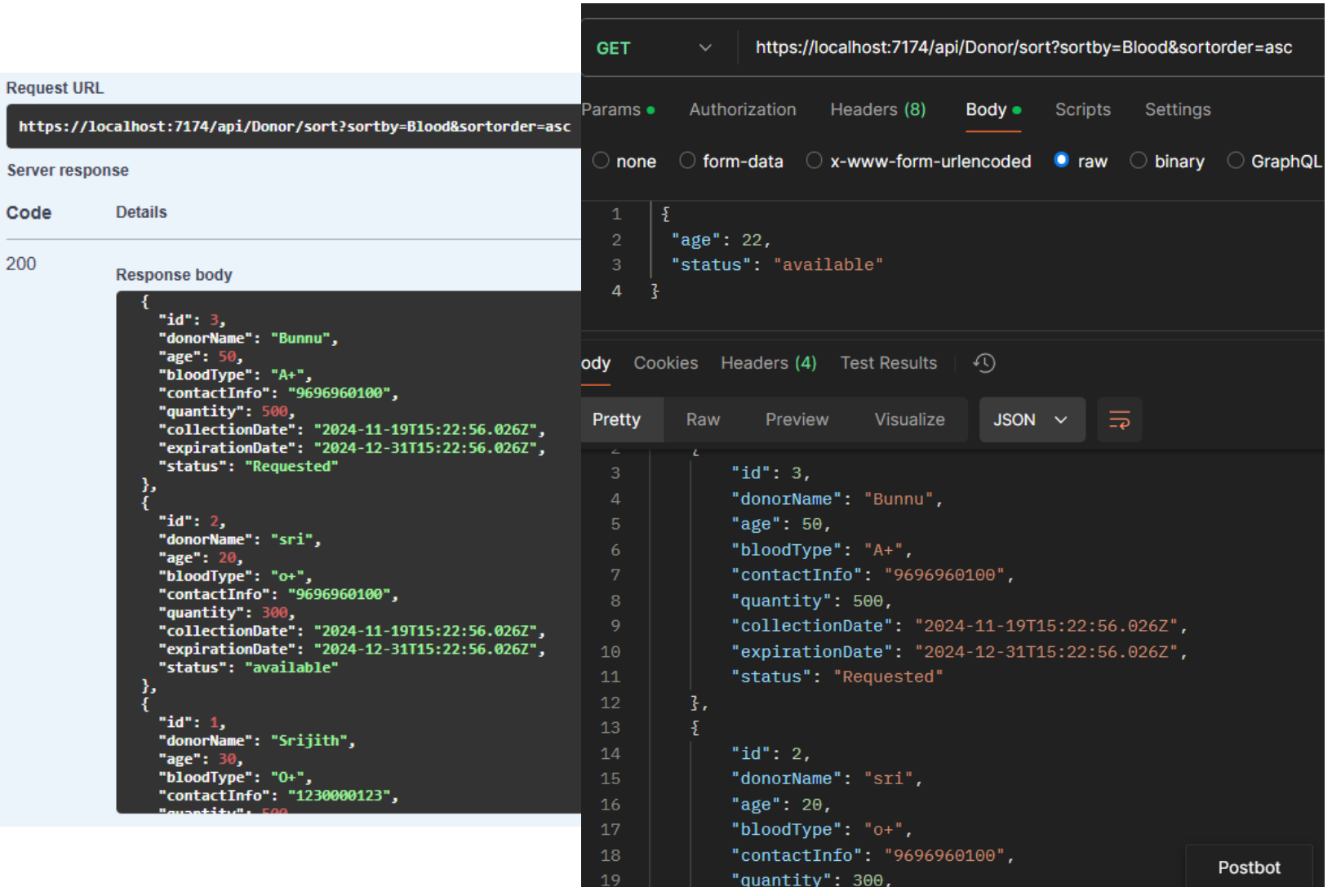


## GET /api/Donor/sort

Returns a sorted list of donors based on a specific attribute (e.g., name, age).

- **Validation:**
  - Ensures the sort parameter matches a valid attribute.
  - Defaults to ascending order if no direction is specified.

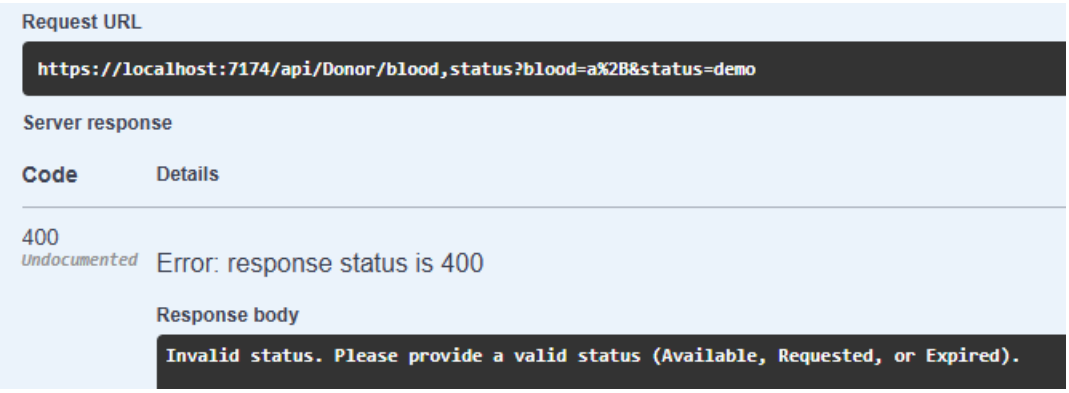


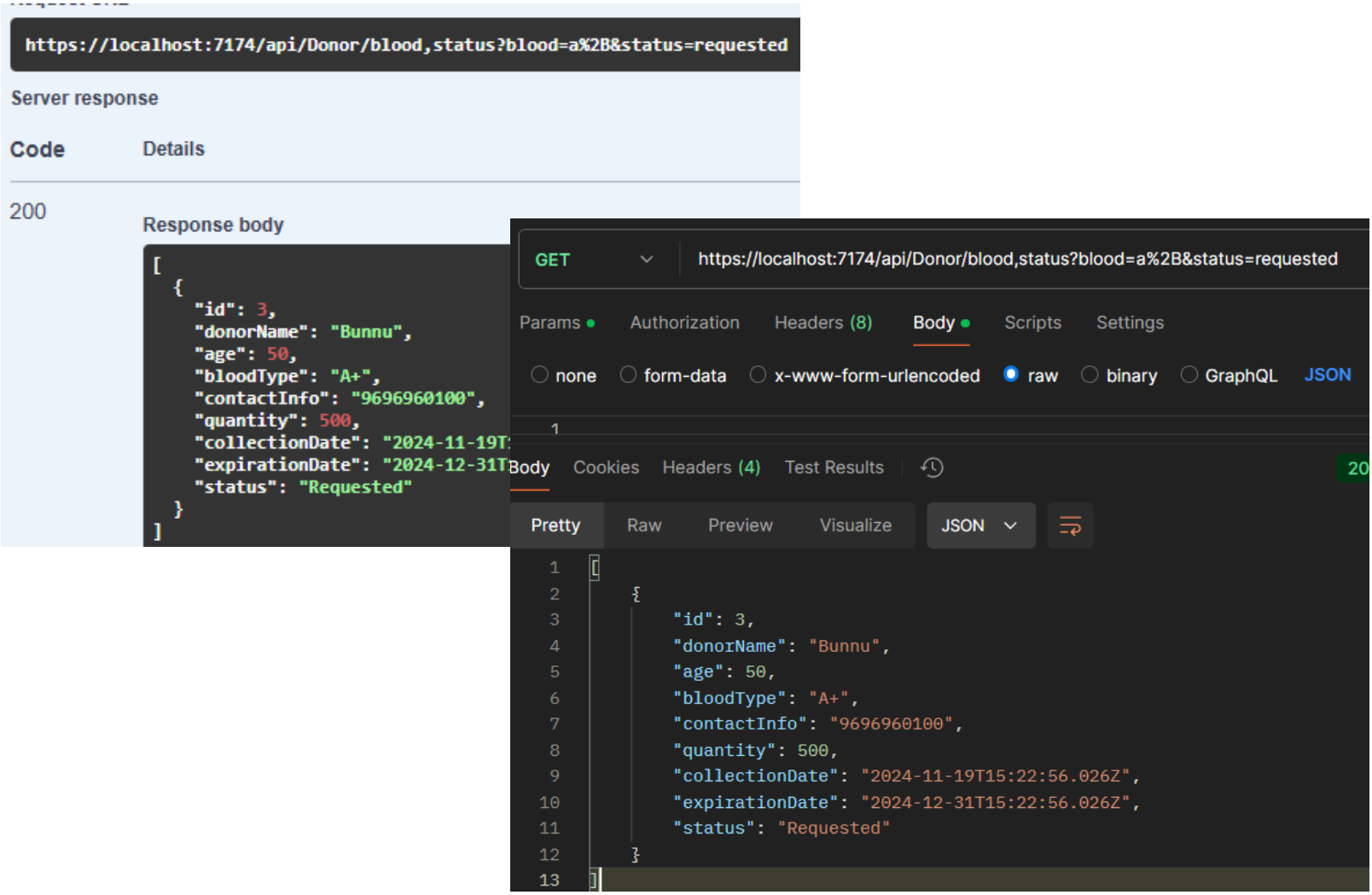
## GET /api/Donor/blood_status

Filters donors by a combination of blood type and status.

- **Validation:**
  - Verifies both blood type and status are valid.

Example :- Blood = a+ , Status = “demo” (invalid input)




Example :- Blood = A+ , Status = “requested”