

## 2XX 응답 코드

The screenshot shows the Postman interface with a workspace named '박세운's Workspace'. The left sidebar shows a collection of requests, including 'POST POST\_3' which is selected. The main panel displays the details of the POST request to 'http://localhost:3000/items'. The request body is a JSON object: 

```
{ 1: { 2:   "name": "test" 3: }}
```

. The response is a 201 Created status, with a response body of 

```
{ 1: { 2:   "status": "success", 3:   "data": { 4:     "id": 0, 5:     "name": "test" 6:   } 7: }}
```

. The status bar indicates '201 Created' with a response time of 99 ms and a body size of 290 B.

The screenshot shows the Postman interface with the same workspace. The left sidebar shows a collection of requests, including 'POST POST\_2' which is selected. The main panel displays the details of the POST request to 'http://localhost:3000/users'. The request body is a JSON object: 

```
{ 1: { 2:   "username": "Park" 3: }}
```

. The response is a 201 Created status, with a response body of 

```
{ 1: { 2:   "status": "success", 3:   "data": { 4:     "id": 1, 5:     "username": "Park" 6:   } 7: }}
```

. The status bar indicates '201 Created' with a response time of 12 ms and a body size of 294 B.

Home Workspaces API Network Search Postman Ctrl K

Workspace 박세윤's Workspace New Import

3분반\_박세윤\_202112016\_실습1 / GET / GET\_1

GET http://localhost:3000/items Send

Don't repeat yourself 🙌  
Create a variable for your base URL and reuse it across your requests like this: {{baseUrl}}

Set as new variable

Name baseUrl

Value http://localhost:3000

Scope Collection

Set Variable No, thanks

200 OK · 9 ms · 287 B · Save Response

```
1 {
2   "status": "success",
3   "data": [
4     {
5       "id": 0,
6       "name": "test"
7     }
8   ]
9 }
```

Home Workspaces API Network Search Postman Ctrl K

Workspace 박세윤's Workspace New Import

3분반\_박세윤\_202112016\_실습1 / GET / GET\_2

GET http://localhost:3000/users/1 Send

Params Authorization Headers (6) Body Scripts Settings Cookies

Query Params

Key	Value	Description
Key	Value	Description

Body Cookies Headers (7) Test Results

200 OK · 7 ms · 291 B · Save Response

```
1 {
2   "status": "success",
3   "data": [
4     {
5       "id": 1,
6       "username": "Park"
7     }
8   ]
9 }
```

Postman interface showing a PUT request to `http://localhost:3000/items/0`. The request body is a JSON object: `{ "name": "testupdate" }`. The response is a 200 OK status with a response time of 9 ms and a body of 291 B. The response body is a JSON object: `{ "status": "success", "data": { "id": 0, "name": "testupdate" } }`.

Postman interface showing a PUT request to `http://localhost:3000/users/1`. The request body is a JSON object: `{ "username": "Lee" }`. The response is a 200 OK status with a response time of 10 ms and a body of 288 B. The response body is a JSON object: `{ "status": "success", "data": { "id": 1, "username": "Lee" } }`.

Postman interface showing a DELETE request to `http://localhost:3000/items/0`. The response is a 200 OK status with a JSON body:

```
{
  "status": "success",
  "data": {
    "id": 0,
    "name": "testupdate"
  }
}
```

The interface includes a sidebar with collections and environments, and a main panel for editing requests and viewing responses.

Postman interface showing a DELETE request to `http://localhost:3000/users/1`. The response is a 200 OK status with a JSON body:

```
{
  "status": "success",
  "data": {
    "id": 1,
    "username": "Lee"
  }
}
```

The interface includes a sidebar with collections and environments, and a main panel for editing requests and viewing responses.

## 4XX 응답 코드

The screenshot shows the Postman interface with a workspace named '박세운's Workspace'. The left sidebar displays a collection of API requests, including DELETE, GET, PATCH, POST, and PUT methods. The main panel shows a POST request to 'http://localhost:3000/items' with a raw JSON body: 

```
1 {  
2  
3 }
```

. The response is a 400 Bad Request, with a status bar indicating '400 Bad Request' and a message: 

```
1 {  
2   "status": "error",  
3   "message": "Name is required",  
4   "data": null  
5 }
```

.

The screenshot shows the Postman interface with a workspace named '박세운's Workspace'. The left sidebar displays a collection of API requests, including DELETE, GET, PATCH, POST, and PUT methods. The main panel shows a POST request to 'http://localhost:3000/users' with a raw JSON body: 

```
1 {  
2  
3 }
```

. The response is a 400 Bad Request, with a status bar indicating '400 Bad Request' and a message: 

```
1 {  
2   "status": "error",  
3   "message": "Username required",  
4   "data": null  
5 }
```

.

Home Workspaces API Network Search Postman Ctrl K

박세운's Workspace New Import

3분반\_박세운\_202112016\_실습1 / GET / GET\_2

GET http://localhost:3000/users/2 Send

Params

Query Params

Key	Value	Description
Key	Value	Description

Body Cookies Headers (7) Test Results 404 Not Found 6 ms 299 B Save Response

JSON Preview Debug with AI

```
1 {
2   "status": "error",
3   "message": "User not found",
4   "data": null
5 }
```

Home Workspaces API Network Search Postman Ctrl K

박세운's Workspace New Import

3분반\_박세운\_202112016\_실습1 / PUT / PUT\_1

PUT http://localhost:3000/items/1 Send

Body raw

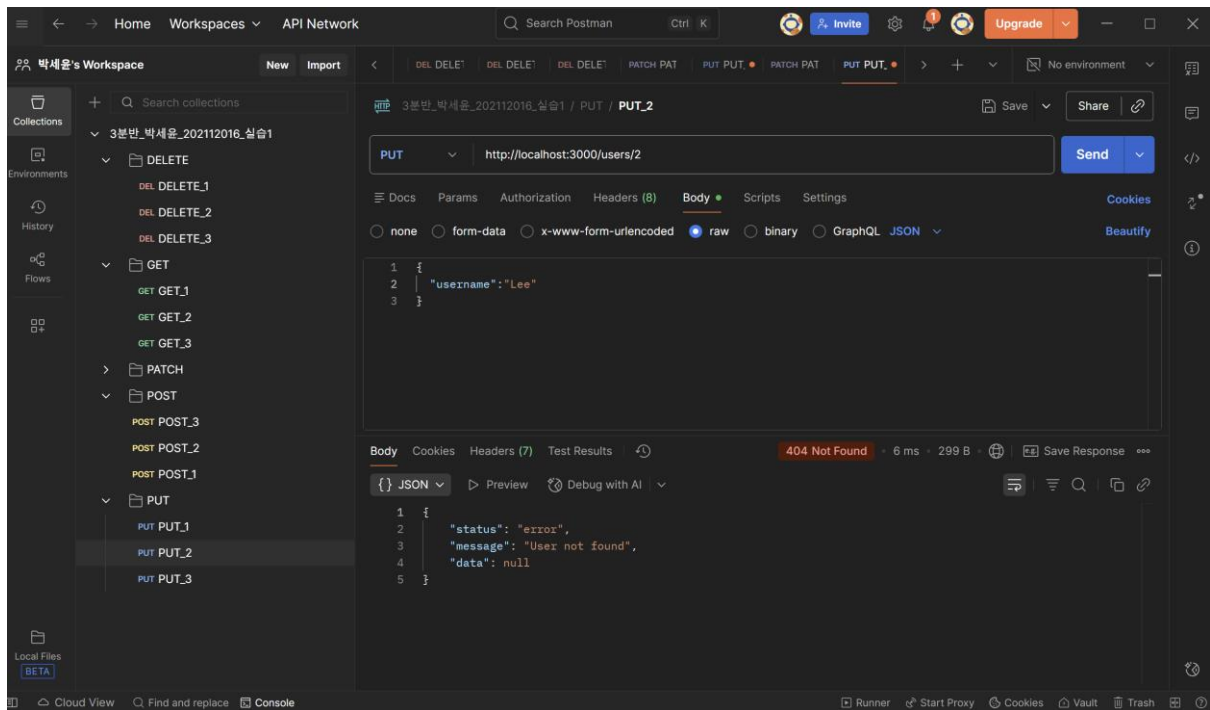
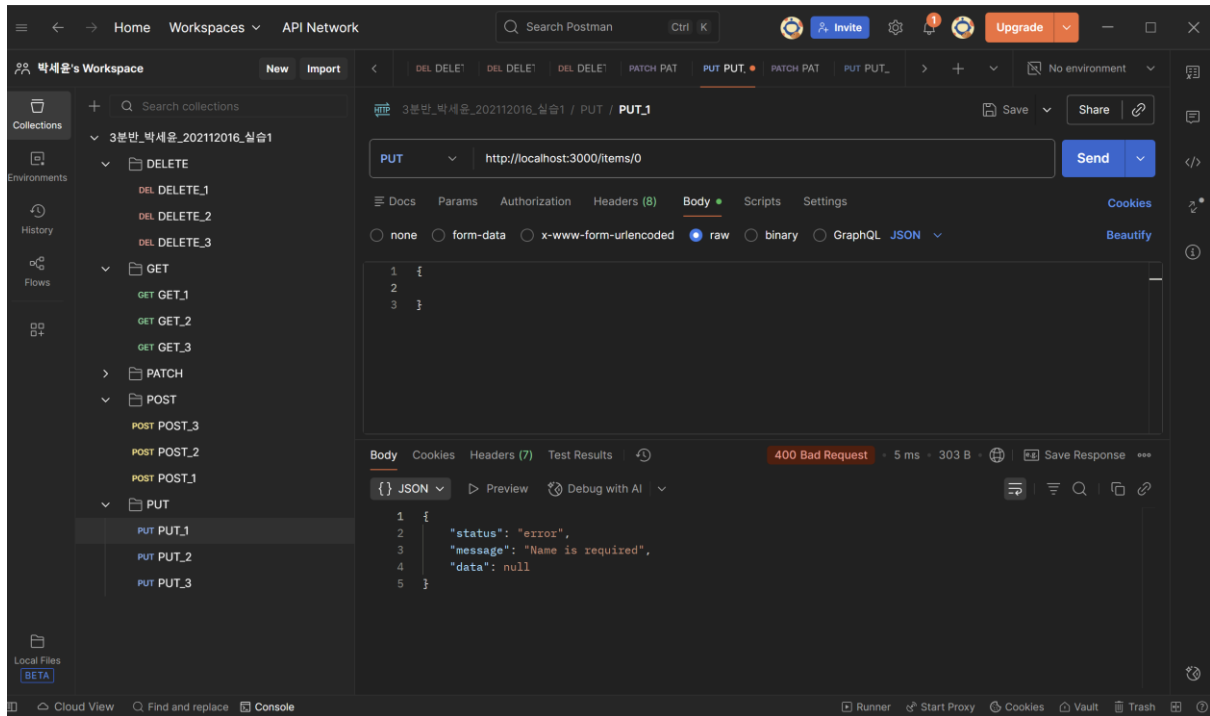
```
1 {
2   "name": "testupdate"
3 }
```

Body Cookies Headers (7) Test Results 404 Not Found 8 ms 299 B Save Response

JSON Preview Debug with AI

```
1 {
2   "status": "error",
3   "message": "Item not found",
4   "data": null
5 }
```

Cloud View Find and replace Console Runner Start Proxy Cookies Vault Trash



Postman interface showing a PUT request to `http://localhost:3000/users/1`. The request body is a JSON object:

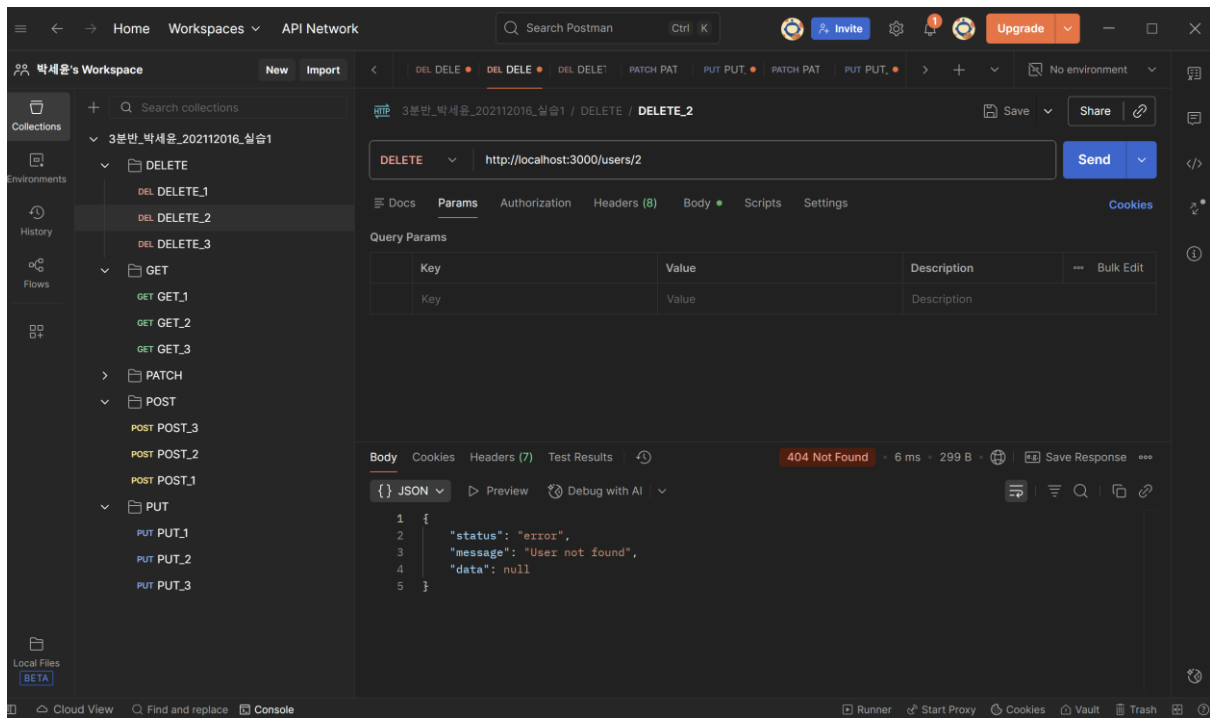
```
1 {
2   }
3 }
```

The response is a 400 Bad Request error:

```
1 {
2   "status": "error",
3   "message": "Username required",
4   "data": null
5 }
```

Postman interface showing a DELETE request to `http://localhost:3000/items/1`. The response is a 404 Not Found error:

```
1 {
2   "status": "error",
3   "message": "Item not found",
4   "data": null
5 }
```



## 5XX 응답 코드

The screenshot shows the Postman interface with a workspace named '박세운's Workspace'. The left sidebar shows a collection of API requests under the folder '3분반\_박세운\_202112016\_실습1'. The selected request is 'POST POST\_3'. The main panel shows the request details for a POST request to 'http://localhost:3000/items'. The response is a 500 Internal Server Error, with a status bar indicating '500 Internal Server Error' and a response time of 57 ms and a body size of 309 B. The response body is shown in JSON format:

```
1 {
2   "name": "error"
3 }
4 }
```

The screenshot shows the Postman interface with the same workspace. The selected request is 'GET GET\_3'. The main panel shows the request details for a GET request to 'http://localhost:3000/maintenance'. The response is a 503 Service Unavailable, with a status bar indicating '503 Service Unavailable' and a response time of 7 ms and a body size of 326 B. The response body is shown in JSON format:

```
1 {
2   "status": "error",
3   "message": "Service temporarily unavailable",
4   "data": null
5 }
```

미들웨어 구현(메소드 호출 시 출력)

```
mid Time 1763183229399  
request logging: GET /maintenance  
app Time: 1763183229401
```