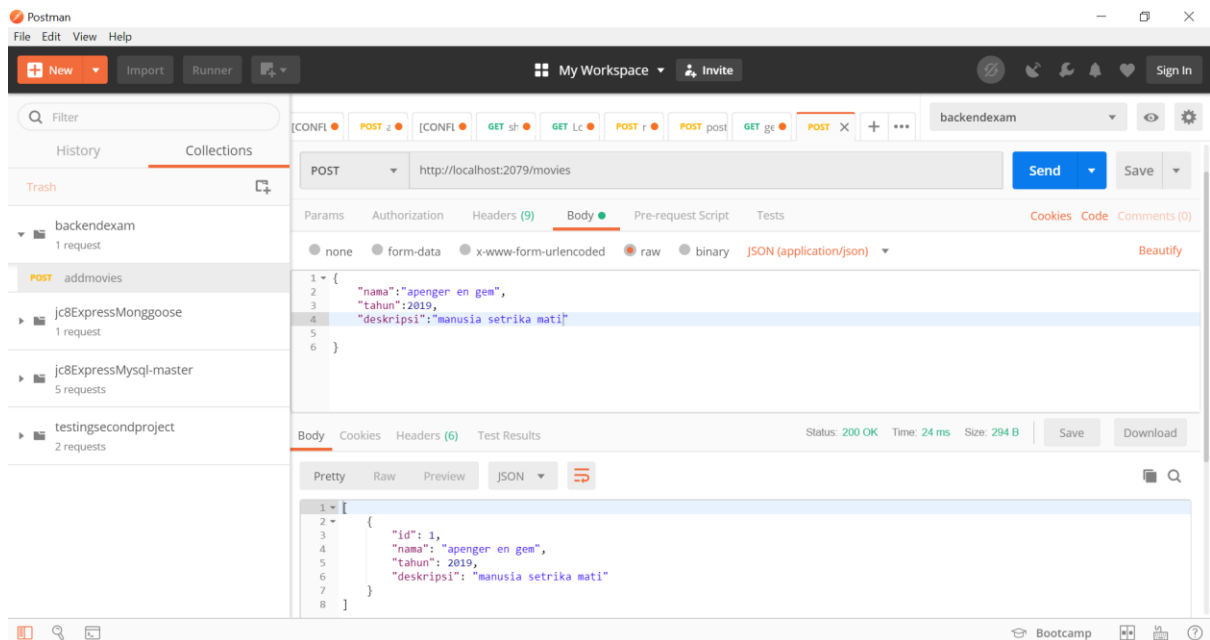
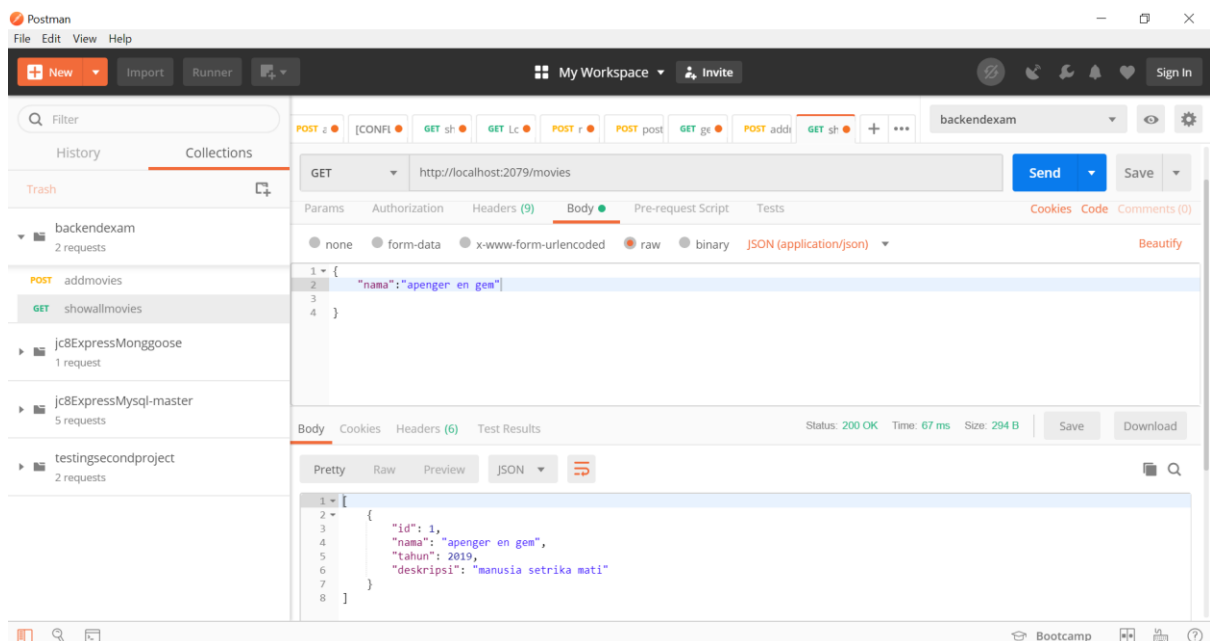


Movies :

1. Post movies



4. show all movies



2. Edit movies

The screenshot shows the Postman application interface. On the left, the 'Collections' tab is active, showing a collection named 'backendexam' with 3 requests. The 'PUT editmovies' request is selected. The main panel displays the details of this request. The URL is 'http://localhost:2079/movies/id'. The method is 'PUT'. The body is a JSON object:

```
{  "id": 2,  "nama": "testing edit",  "tahun": 2019,  "deskripsi": "mencari kitabsuci"}
```

. The status bar at the bottom indicates 'Status: 200 OK', 'Time: 71 ms', and 'Size: 306 B'.

The screenshot shows the Postman application interface after the request has been executed. The 'PUT editmovies' request is still selected. The main panel displays the response. The status bar at the bottom indicates 'Status: 200 OK', 'Time: 11 ms', and 'Size: 381 B'. The response body is a JSON object:

```
{  "fieldCount": 0,  "affectedRows": 1,  "insertId": 0,  "serverStatus": 2,  "warningCount": 0}
```

. The 'Headers' tab is active, showing 6 headers.

3. Delete movies

The screenshot shows the Postman interface with a DELETE request configured. The URL is `http://localhost:2079/movies/id`. The body is set to JSON (application/json) with the following content:

```
1 {
2   "name": "testing edit"
3 }
```

The status bar indicates a successful response: Status: 200 OK, Time: 80 ms, Size: 340 B. The response body is displayed in the bottom panel:

```
1 {
2   "fieldCount": 0,
3   "affectedRows": 1,
4   "insertId": 0,
5   "serverStatus": 2,
6   "warningCount": 0,
7   "message": "",
8   "protocol41": true
9 }
```

The screenshot shows the Postman interface with a PUT request configured. The URL is `http://localhost:2079/movies/id`. The body is set to JSON (application/json) with the following content:

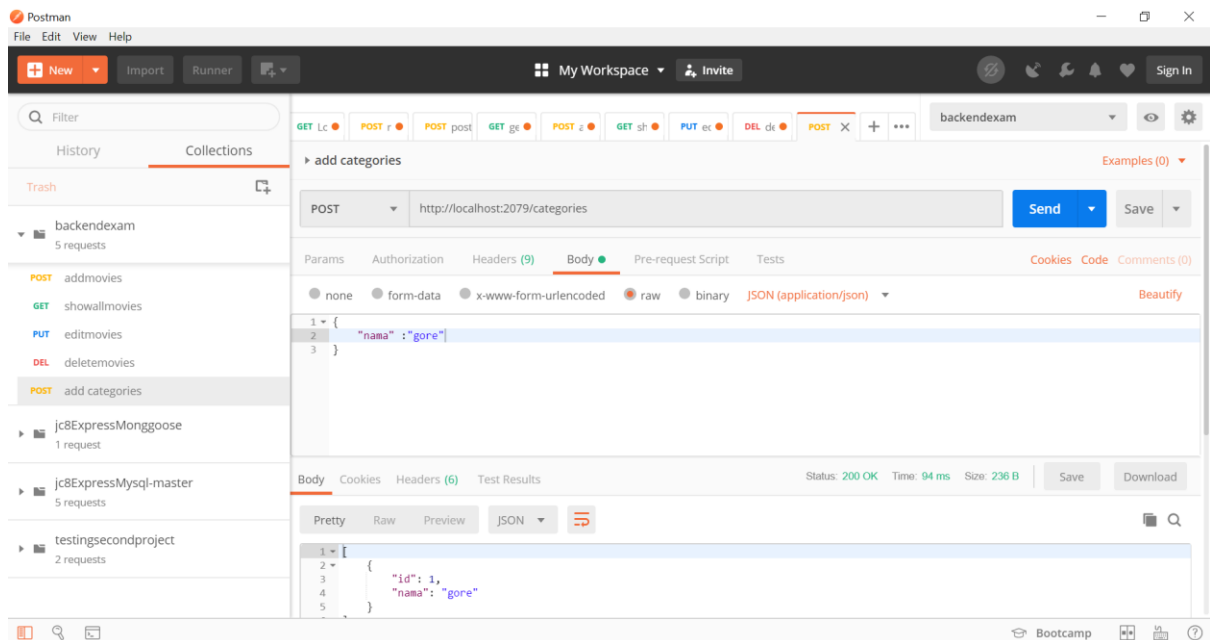
```
1 {
2   "fieldCount": 0,
3   "affectedRows": 1,
4   "insertId": 0,
5   "serverStatus": 2,
6   "warningCount": 0
7 }
```

The status bar indicates a successful response: Status: 200 OK, Time: 11 ms, Size: 381 B. The response body is displayed in the bottom panel:

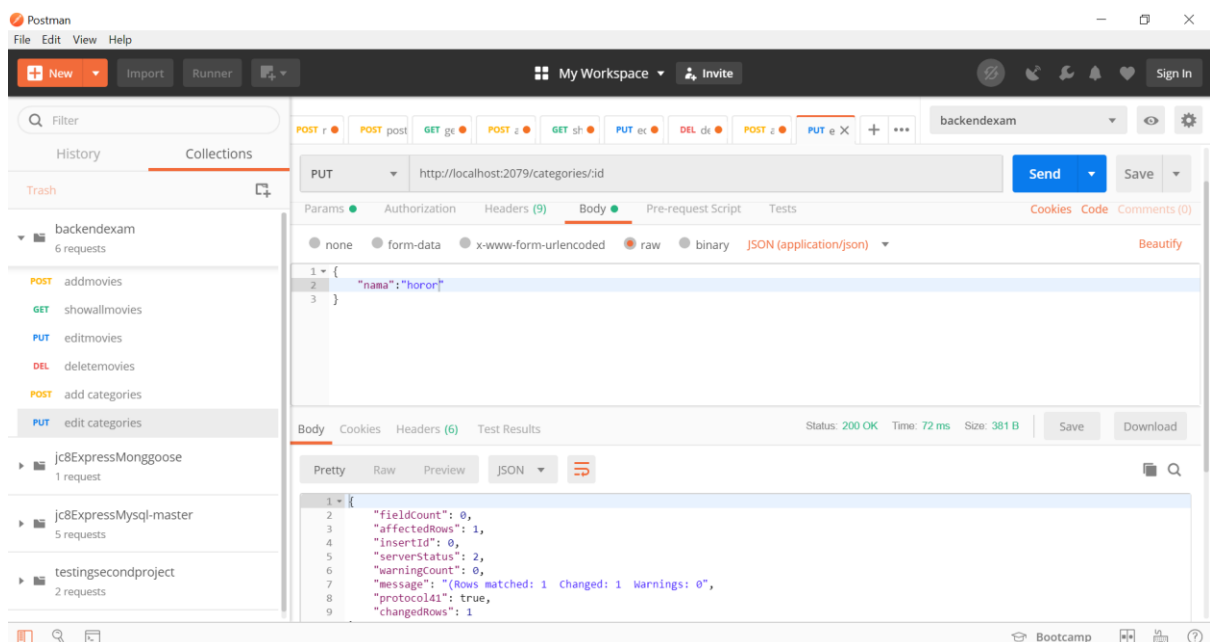
```
1 {
2   "fieldCount": 0,
3   "affectedRows": 1,
4   "insertId": 0,
5   "serverStatus": 2,
6   "warningCount": 0
7 }
```

Categories

1. Post categories



2. Edit categories



Postman

File Edit View Help

New Import Runner My Workspace Invite

Filter

History Collections

Trash

backendexam

9 requests

- POST addmovies
- GET showwallmovies
- PUT editmovies
- DEL deletemovies
- POST add categories
- PUT edit categories
- DEL delete categories
- GET show all categories
- POST get movie id

jc8ExpressMongoose 1 request

jc8ExpressMysql-master 5 requests

PUT http://localhost:2079/categories/id

Send Save

Params Authorization Headers (9) Body Pre-request Script Tests Cookies Code Comments (0)

Query Params

KEY	VALUE	DESCRIPTION	Bulk Edit
Key	Value	Description	

Path Variables

KEY	VALUE	DESCRIPTION	Bulk Edit
id	2	Description	

Body Cookies Headers (6) Test Results Status: 200 OK Time: 72 ms Size: 381 B Save Download

Pretty Raw Preview JSON

```
1 {
2   "fieldCount": 0,
3   "affectedRows": 1,
4   "insertId": 0,
5   "serverStatus": 2,
6   "warningCount": 0,
7   "message": "(Rows matched: 1 Changed: 1 Warnings: 0",
8   "protocol41": true,
9   "changedRows": 1
10 }
```

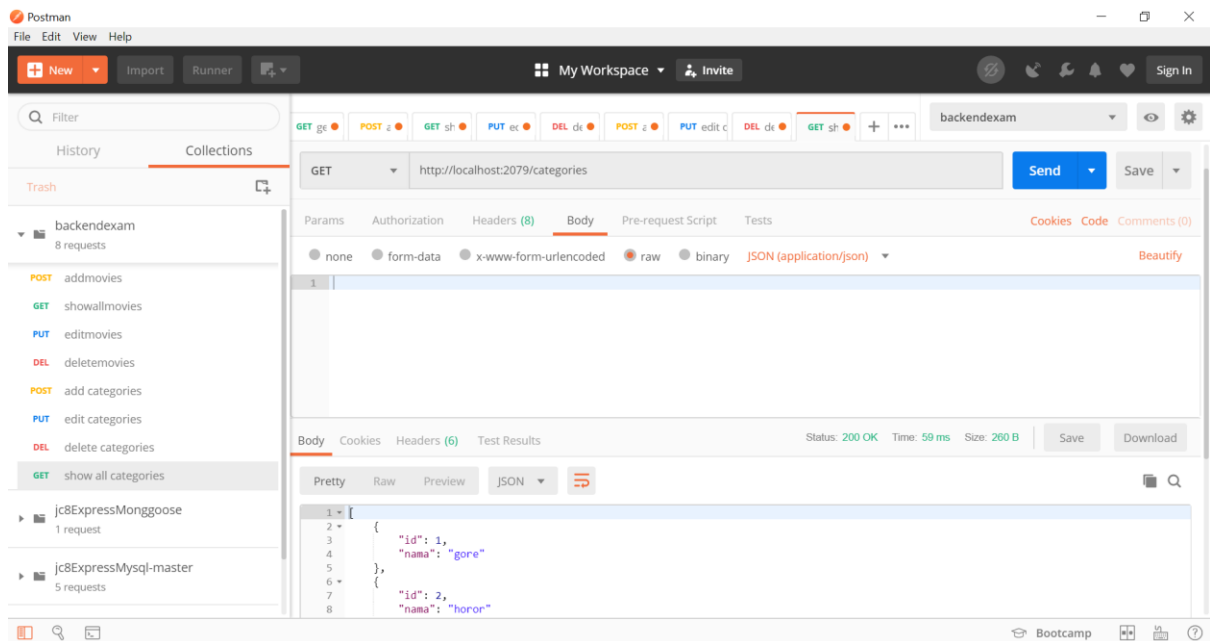
Bootcamp

3. Edit categories

Postman interface showing a DELETE request to `http://localhost:2079/categories/id`. The request body is a JSON object: `{ "name": "comedy" }`. The response status is 200 OK, and the response body is a JSON object: `{ "fieldCount": 0, "affectedRows": 1, "insertId": 0, "serverStatus": 2, "warningCount": 0, "message": "", "protocol41": true }`.

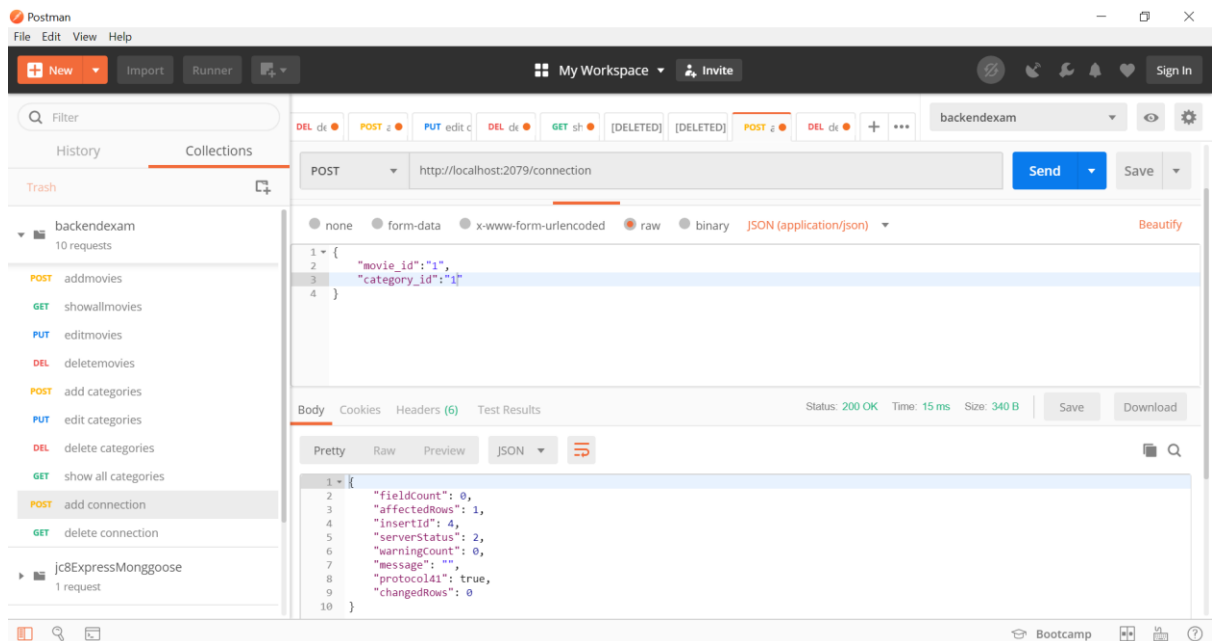
Postman interface showing a PUT request to `http://localhost:2079/categories/id`. The request body is a JSON object: `{ "name": "comedy" }`. The response status is 200 OK, and the response body is a JSON object: `{ "fieldCount": 0, "affectedRows": 1, "insertId": 0, "serverStatus": 2, "warningCount": 0, "message": "(Rows matched: 1 Changed: 1 Warnings: 0)", "protocol41": true, "changedRows": 1 }`.

4. delete categories



Connection

1. Add connection



2. Delete connection

