

# HOW TO TEST ON POSTMAN

URL : localhost:2020

## 1. ADD MOVIES

The screenshot shows the Postman interface for a POST request. The URL is `{{url}}/movies`. The request body is a JSON object with the following fields: `"nama": "Avenger Endgame", "tahun": "2019", "deskripsi": "Epic Comeback Lah"`. The request is set to use raw JSON with the content type `application/json`.

```
1 {
2   "nama": "Avenger Endgame",
3   "tahun": "2019",
4   "deskripsi": "Epic Comeback Lah"
5 }
```

## 2. EDIT MOVIES

The screenshot shows the Postman interface for a PATCH request. The URL is `{{url}}/movies/:movieId`. The request body is a JSON object with the following fields: `"nama": "tralalatrili", "tahun": "2020", "deskripsi": "mantap punya"`. The request is set to use raw JSON with the content type `application/json`.

Query Params

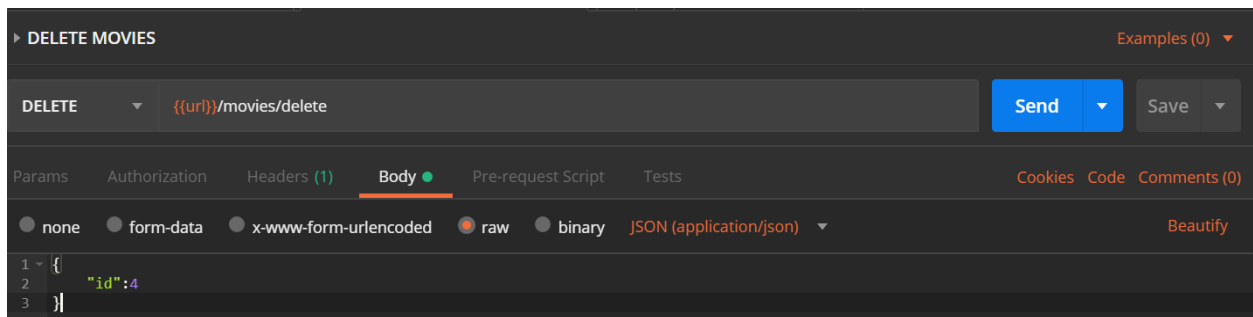
KEY	VALUE	DESCRIPTION
Key	Value	Description

Path Variables

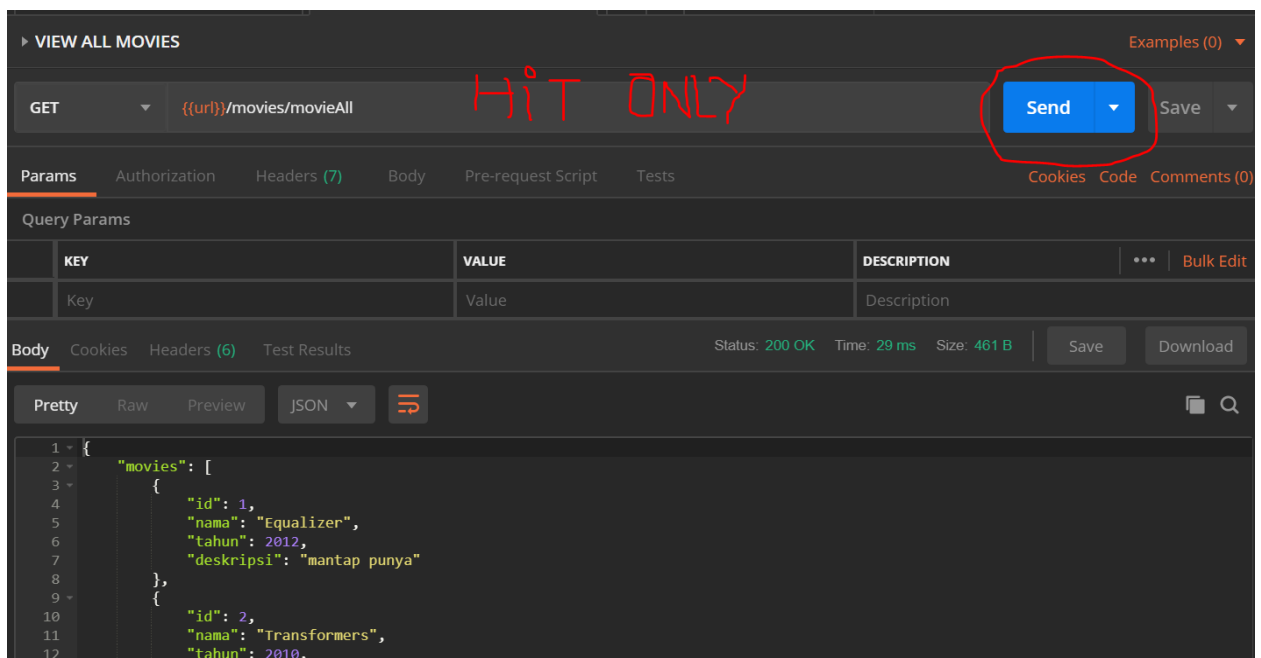
KEY	VALUE	DESCRIPTION
movieId	1	Description

```
1 {
2   "nama": "tralalatrili",
3   "tahun": "2020",
4   "deskripsi": "mantap punya"
5 }
```

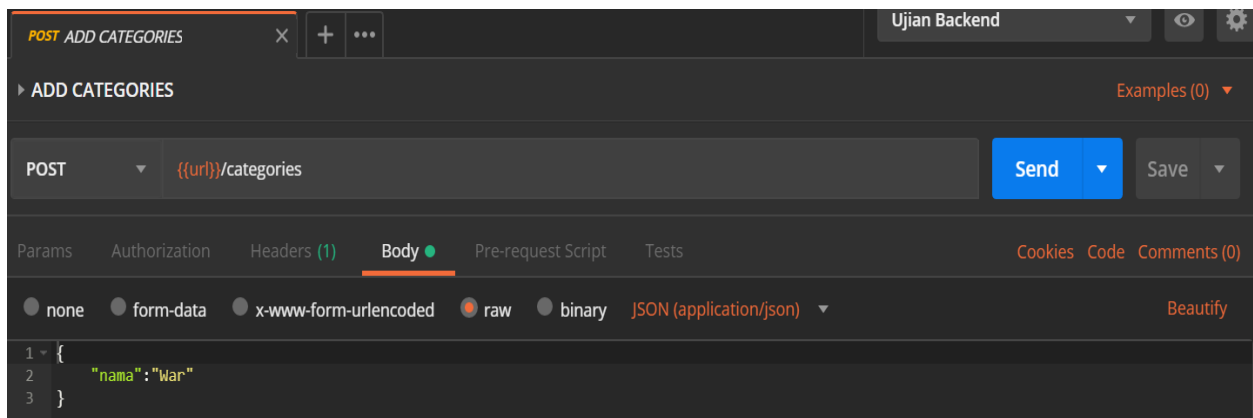
## 3. DELETE MOVIES



#### 4. SHOW ALL MOVIES



#### 5. ADD CATEGORIES



## 6. EDIT CATEGORIES

EDIT CATEGORIES Examples (0) ▾

PATCH `{{url}}/categories/catgId` Send ▾ Save ▾

Params Authorization Headers (1) 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
catgId	5	Description		

EDIT CATEGORIES Examples (0) ▾

PATCH `{{url}}/categories/catgId` Send ▾ Save ▾

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code Comments (0)

none form-data x-www-form-urlencoded raw binary JSON (application/json) Beautify

```
1 {  
2   "nama": "lalalalaland"  
3 }
```

## 7. DELETE CATEGORIES

DELETE CATEGORIES Examples (0) ▾

DELETE `{{url}}/categories/delete` Send ▾ Save ▾

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code Comments (0)

none form-data x-www-form-urlencoded raw binary JSON (application/json) Beautify

```
1 {  
2   "id": 5  
3 }
```

## 8. VIEW ALL CATEGORIES

VIEW ALL CATEGORIES Examples (0) ▾

GET `{{url}}/categories/catgAll` HIT ONLY Send ▾ Save ▾

Params Authorization Headers (7) Body Pre-request Script Tests Cookies Code Comments (0)

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (6) Test Results Status: 200 OK Time: 17 ms Size: 328 B Save Download

Pretty Raw Preview JSON ↺

```
1 {  
2   "categories": [  
3     {  
4       "id": 1,  
5       "nama": "Action"  
6     },  
7     {  
8       "id": 2,  
9       "nama": "Drama"  
10    },  
11    {  
12      "id": 3,
```

## 9. ADD CONNECTION

The screenshot shows a REST client interface with a POST request configured. The URL is `{{url}}/movcat/movieId/catId`. The request body is a JSON object: `{ "movie_id": 3, "category_id": 2 }`. The response status is 200 OK, with a time of 221 ms and a size of 545 B. The response body is displayed in a pretty-printed JSON format: `[ { "nama_movie": "Equalizer", "nama_category": "Action" }, { "nama_movie": "Equalizer", "nama_category": "Drama" } ]`.

```
POST {{url}}/movcat/movieId/catId
```

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

none form-data x-www-form-urlencoded **raw** binary JSON (application/json) Beautify

```
1 {
2   "movie_id": 3,
3   "category_id": 2
4 }
```

Body Cookies Headers (6) Test Results Status: 200 OK Time: 221 ms Size: 545 B Save Download

Pretty Raw Preview JSON

```
1 [
2   {
3     "nama_movie": "Equalizer",
4     "nama_category": "Action"
5   },
6   {
7     "nama_movie": "Equalizer",
8     "nama_category": "Drama"
9   },
10 ]
```

## 10. DELETE CONNECTION

The screenshot shows a REST client interface with a DELETE request configured. The URL is `{{url}}/movcat/delete`. The request body is a JSON object: `{ "id": 5 }`.

```
DELETE {{url}}/movcat/delete
```

Params Authorization Headers (1) **Body** Pre-request Script Tests Cookies Code Comments (0)

none form-data x-www-form-urlencoded **raw** binary JSON (application/json) Beautify

```
1 {
2   "id": 5
3 }
```