

Nama : Gilang Trisetya Indrawan

Kode Peserta : FSDO002ONL018

## 1. Create Movie (POST)

The screenshot shows the Postman interface with a POST request to `https://localhost:5001/api/movie`. The request body is a JSON object: `{ "name": "Psiko", "genre": "Psyco", "duration": "3 hour", "releaseDate": "2018-10-01" }`. The response status is 201 Created, with a time of 38 ms and a size of 294 B. The response body is a JSON object: `{ "id": 7, "name": "Psiko", "genre": "Psyco", "duration": "3 hour", "releaseDate": "2018-10-01T00:00:00" }`.

```
POST https://localhost:5001/api/movie

{
  "name": "Psiko",
  "genre": "Psyco",
  "duration": "3 hour",
  "releaseDate": "2018-10-01"
}
```

Status: 201 Created Time: 38 ms Size: 294 B

```
{
  "id": 7,
  "name": "Psiko",
  "genre": "Psyco",
  "duration": "3 hour",
  "releaseDate": "2018-10-01T00:00:00"
}
```

## 2. Read All Movie (GET)

The screenshot shows the Postman interface with a GET request to `https://localhost:5001/api/movie`. The response status is 200 OK, with a time of 12 ms and a size of 774 B. The response body is a JSON array of two movie objects: `[ { "id": 6, "name": "AAAAAAAAAAAA", "genre": "Action", "duration": "2 hour", "releaseDate": "2021-11-26T13:29:32" }, { "id": 7, "name": "Psiko", "genre": "Psyco", "duration": "3 hour", "releaseDate": "2018-10-01T00:00:00" } ]`.

```
GET https://localhost:5001/api/movie
```

Status: 200 OK Time: 12 ms Size: 774 B

```
[
  {
    "id": 6,
    "name": "AAAAAAAAAAAA",
    "genre": "Action",
    "duration": "2 hour",
    "releaseDate": "2021-11-26T13:29:32"
  },
  {
    "id": 7,
    "name": "Psiko",
    "genre": "Psyco",
    "duration": "3 hour",
    "releaseDate": "2018-10-01T00:00:00"
  }
]
```

### 3. Read Movie By Id (GET)

The screenshot shows the Postman application interface. On the left sidebar, the 'Environments' tab is active, showing a list of environments: 'Gilang\_Sesi10 With JWT', 'Gilang\_Sesi10 Without JWT', and 'Gilang\_Sesi10\_Assignment3'. The 'GET Movie' environment is selected. The main panel displays a GET request to 'https://localhost:5001/api/movie/7'. The 'Query Params' section is empty. The 'Body' tab is selected, showing a JSON response in 'Pretty' format. The response status is '200 OK' with a time of '10 ms' and a size of '414 B'. The JSON response is as follows:

```
1 {
2   "result": {
3     "id": 7,
4     "name": "Psiko",
5     "genre": "Psyco",
6     "duration": "3 hour",
7     "releaseDate": "2018-10-01T00:00:00"
8   },
9   "id": 2,
10  "exception": null,
11  "status": 5,
12  "isCanceled": false,
13  "isCompleted": true,
14  ...
15 }
```

The bottom of the image shows a Windows taskbar with various application icons and a system clock indicating 20:47 on 26/11/2021.

#### 4. Update Movie By Id (PUT)

The screenshot shows the Postman interface with a PUT request to `https://localhost:5001/api/movie/7`. The request body is a JSON object: `{ "name": "Happy", "genre": "Happy", "duration": "1 hour", "releaseDate": "2019-10-01" }`. The response status is 200 OK, and the body is `1 Update movie success`.

Left sidebar menu:

- APIs
  - Gilang\_Sesi10 With JWT 🔒
  - Gilang\_Sesi10 Without JWT 🔒
  - Gilang\_Sesi10\_Assignment3 🔒
    - GET All Movie
    - PUT Movie
    - DEL Movie
    - POST Movie
- Environments
- Mock Servers
- Monitors
- History

Top bar: Find and Replace, Console, Bootcamp, Runner, Trash

#### Data terupdate

The screenshot shows the Postman interface with a GET request to `https://localhost:5001/api/movie/7`. The response status is 200 OK, and the body is a JSON object: `{ "result": { "id": 7, "name": "Happy", "genre": "Happy", "duration": "1 hour", "releaseDate": "2019-10-01T00:00:00" }, "id": 3, "exception": null, "status": 5, "isCanceled": false, "isCompleted": true }`.

Left sidebar menu:

- APIs
  - Gilang\_Sesi10 With JWT 🔒
  - Gilang\_Sesi10 Without JWT 🔒
  - Gilang\_Sesi10\_Assignment3 🔒
    - GET All Movie
    - GET Movie
    - PUT Movie
    - DEL Movie
    - POST Movie
- Environments
- Mock Servers
- Monitors
- History

Top bar: Find and Replace, Console, Bootcamp, Runner, Trash

## 5. Delete Movie By Id (DELETE)

The screenshot shows the Postman interface with a DELETE request to `https://localhost:5001/api/movie/7`. The request is successful, returning a status of 200 OK. The response body is displayed in the 'Text' view, showing the message: `1 Delete movie success`.

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body: Cookies Headers (4) Test Results

Status: 200 OK Time: 46 ms Size: 162 B Save Response

## Data terhapus

The screenshot shows the Postman interface with a GET request to `https://localhost:5001/api/movie/7`. The request is successful, returning a status of 200 OK. The response body is displayed in the 'JSON' view, showing the following data:

```
1 {
2   "result": null,
3   "id": 4,
4   "exception": null,
5   "status": 5,
6   "isCanceled": false,
7   "isCompleted": true,
8   "isCompletedSuccessfully": true,
9   "creationOptions": 0,
10  "asyncState": null,
11  "isFaulted": false
12 }
```

Body: Cookies Headers (4) Test Results

Status: 200 OK Time: 112 ms Size: 323 B Save Response