

Nama : Kafka Putra Riyadi  
Kelas : SE 07-02  
NIM : 2311104041

## JURNAL MODUL 9

1. A. Mencoba “GET /api/Movies” saat baru dijalankan yang mengeluarkan list film dari TOP 3 IMDB seperti pada tampilan berikut pada saat dicoba dengan menekan tombol “Try it out” dan tombol “Execute”

The screenshot displays a REST client interface with the following components:

- Execute** button (highlighted in blue) and **Clear** button.
- Responses** section.
- Curl** section showing the command: 

```
curl -X 'GET' \
'https://localhost:7027/api/Movies' \
-H 'accept: text/plain'
```
- Request URL** section showing: `https://localhost:7027/api/Movies`
- Server response** section with a **Code** tab selected, showing a **200** status.
- Response body** section showing a JSON array of movie data:

```
[
  {
    "title": "The Shawshank Redemption",
    "director": "Frank Darabont",
    "stars": [
      "Jim Robbins",
      "Morgan Freeman"
    ],
    "description": "Two imprisoned men bond over years."
  },
  {
    "title": "The Godfather",
    "director": "Francis Ford Coppola",
    "stars": [
      "Marlon Brando",
      "Al Pacino"
    ],
    "description": "The aging patriarch of an organized crime dynasty..."
  },
  {
    "title": "The Dark Knight",
    "director": "Christopher Nolan",
    "stars": [
      "Christian Bale",
      "Heath Ledger"
    ],
    "description": "Batman battles the Joker."
  }
]
```

At the bottom, there is a **Response headers** section and a **Download** button.

B. Menambahkan Movie baru yaitu urutan ke-4 pada TOP IMDB list dengan memanggil API pada bagian “POST /api/Movies”

The screenshot shows a REST client interface with the following sections:

- POST /api/Movies**: The endpoint and method.
- Parameters**: A section with "No parameters" and buttons for "Cancel" and "Reset".
- Request body**: A text area containing a JSON object: 

```
{  "title": "12 Angry Men",  "director": "Sidney Lumet",  "stars": [    "Henry Fonda",    "Lee J. Cobb"  ],  "description": "A jury holdout attempts to prevent a miscarriage of justice."}
```

 The content type is set to "application/json".
- Execute** and **Clear** buttons.
- Responses**: A section for the response, currently empty.
- Request URL**: A section for the request URL, currently empty.

C. Cek list/array dari semua Movie lagi dengan “GET /api/Movies”, pastikan Movie yang baru ditambahkan sebelumnya sudah ada:

The screenshot shows a REST client interface with the following sections:

- Execute** and **Clear** buttons.
- Responses**: A section for the response, currently empty.
- Request**: A section for the request, containing a curl command: 

```
curl -X 'GET' \  'https://localhost:7027/api/Movies' \  -H 'accept: text/plain'
```
- Request URL**: A section for the request URL, containing `https://localhost:7027/api/Movies`.
- Server response**: A section for the server response, showing a status code of 200.
- Response body**: A text area containing a JSON array of movie objects: 

```
[  {    "title": "The Godfather",    "director": "Francis Ford Coppola",    "stars": [      "Marlon Brando",      "Al Pacino"    ],    "description": "The aging patriarch of an organized crime dynasty..."  },  {    "title": "The Dark Knight",    "director": "Christopher Nolan",    "stars": [      "Christian Bale",      "Heath Ledger"    ],    "description": "Batman battles the Joker."  },  {    "title": "12 Angry Men",    "director": "Sidney Lumet",    "stars": [      "Henry Fonda",      "Lee J. Cobb"    ],    "description": "A jury holdout attempts to prevent a miscarriage of justice."  }]
```

 A "Download" button is visible at the bottom right of the response body.
- Response headers**: A section for the response headers, currently empty.

D. Mencoba meminta Movie dengan index 3, “GET /api/Movies/3” yang seharusnya mengembalikan Movie yang baru saja ditambah:

GET

/api/Movies/{id}

Parameters

Cancel

Name	Description
id * required	
integer(\$int32)	3
(path)	

ExecuteClear

Responses

Curl

```
curl -X 'GET' \
'https://localhost:7027/api/Movies/3' \
-H 'accept: text/plain'
```

Request URL

```
https://localhost:7027/api/Movies/3
```

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{   "title": "12 Angry Men",   "director": "Sidney Lumet",   "stars": [     "Henry Fonda",     "Lee J. Cobb"   ],   "description": "A jury holdout attempts to prevent a miscarriage of justice." }</pre></div><div>Download</div></div>

Response headers

E. Menghapus objek Movie dengan index ke-1 dengan “DELETE /api/Movies/1”

DELETE

/api/Movies/{id}

Parameters

Cancel

Name	Description
id * required	
integer(\$int32)	1
(path)	

ExecuteClear

Responses

Curl

```
curl -X 'DELETE' \
'https://localhost:7027/api/Movies/1' \
-H 'accept: */*'
```

Request URL

```
https://localhost:7027/api/Movies/1
```

Server response

Code	Details
200	<div><div>Response headers</div><div><pre>content-length: 0 date: Wed, 30 Apr 2025 13:33:20 GMT server: Kestrel</pre></div></div>

Responses

Code	Description	Links
200	OK	No links

F. Cek list/array dari semua Movie sekali lagi dengan “GET /api/Movies”, film dengan ranking kedua “Godfather” sudah tidak ada di list:

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'https://localhost:7027/api/Movies' \
  -H 'accept: text/plain'
```

Request URL

```
https://localhost:7027/api/Movies
```

Server response

Code

Details

200

Response body

```
{
  "title": "The Shawshank Redemption",
  "director": "Frank Darabont",
  "stars": [
    "Tim Robbins",
    "Morgan Freeman"
  ],
  "description": "Two imprisoned men bond over years."
},
{
  "title": "The Dark Knight",
  "director": "Christopher Nolan",
  "stars": [
    "Christian Bale",
    "Heath Ledger"
  ],
  "description": "Batman battles the Joker."
},
{
  "title": "12 Angry Men",
  "director": "Sidney Lumet",
  "stars": [
    "Henry Fonda",
    "Lee J. Cobb"
  ],
  "description": "A jury holdout attempts to prevent a miscarriage of justice."
}
```

Response headers

Download