

Nama : Salman Alfarisi

Kelas : S1SE-07-01

NIM: 2311104036

```
const express = require('express');
const bodyParser = require('body-parser');
const movies = require('./movies');
const Movie = require('./Movieclass');

const app = express();
app.use(bodyParser.json());

app.get('/api/Movies', (req, res) => {
  res.json(movies);
});

app.get('/api/Movies/:id', (req, res) => {
  const id = parseInt(req.params.id);
  if (id >= 0 && id < movies.length) {
    res.json(movies[id]);
  } else {
    res.status(404).send("Movie not found");
  }
});

app.post('/api/Movies', (req, res) => {
  const { title, director, stars, description } = req.body;
  const newMovie = new Movie(title, director, stars, description);
  movies.push(newMovie);
  res.status(201).send("Movie added");
});

app.delete('/api/Movies/:id', (req, res) => {
  const id = parseInt(req.params.id);
  if (id >= 0 && id < movies.length) {
    movies.splice(id, 1);
    res.send("Movie deleted");
  } else {
    res.status(404).send("Movie not found");
  }
});

const PORT = 3000;
app.listen(PORT, () => {
  console.log(`Server is running at http://localhost:${PORT}`);
});

app.get('/', (req, res) => {
  res.send('Web API Movie sedang berjalan. Gunakan endpoint /api/Movies');
});
```

```

class Movie {
  constructor(title, director, stars, description) {
    this.title = title;
    this.director = director;
    this.stars = stars; // Array of strings
    this.description = description;
  }
}

module.exports = Movie;

```

```

const Movie = require('./Movieclass');

let movies = [
  new Movie("The Shawshank Redemption", "Frank Darabont", ["Tim Robbins", "Morgan Freeman"], "Two imprisoned men bond over a number of years."),
  new Movie("The Godfather", "Francis Ford Coppola", ["Marlon Brando", "Al Pacino"], "The aging patriarch of an organized crime dynasty transfers control to his reluctant son."),
  new Movie("The Dark Knight", "Christopher Nolan", ["Christian Bale", "Heath Ledger"], "Batman battles the Joker to save Gotham.")
];

module.exports = movies;

```

Kode ini merupakan bagian dari implementasi backend untuk aplikasi Web API yang mengelola data film. File ini menyimpan data awal berupa daftar film dalam bentuk array `movies`. Setiap elemen dalam array adalah instance dari kelas `Movie`, yang diimpor dari file `Movieclass`. Kelas `Movie` sendiri memiliki atribut `title` (judul film), `director` (sutradara), `stars` (daftar pemeran utama dalam bentuk array string), dan `description` (deskripsi singkat film).

Dalam array `movies`, terdapat tiga film populer yang telah diinisialisasi: "The Shawshank Redemption", "The Godfather", dan "The Dark Knight", masing-masing dengan nama sutradara, daftar aktor, dan deskripsinya. Setelah data ini dibuat, array `movies` diekspor dengan `module.exports`, sehingga dapat digunakan oleh file lain, seperti file utama Express server (`server.js` atau yang serupa), untuk memberikan data film kepada client melalui endpoint API. Dengan pendekatan ini, daftar film disiapkan secara statis di awal dan dapat dimodifikasi melalui endpoint API yang sudah tersedia di server.

A. GET Api/Movies

The screenshot shows the Postman interface with a GET request to `http://localhost:3000/api/Movies`. The request is saved, and the response is a 200 OK status with a 49 ms response time and 770 B of data. The response body is a JSON array of movie objects.

```
1 {
2   {
3     "title": "The Shawshank Redemption",
4     "director": "Frank Darabont",
5     "stars": [
6       "Tim Robbins",
7       "Morgan Freeman"
8     ],
9     "description": "Two imprisoned men bond over a number of years."
10  },
11  {
12    "title": "The Godfather",
13    "director": "Francis Ford Coppola",
14    "stars": [
15      "Marlon Brando",
16      "Al Pacino"
17    ],
18    "description": "The aging patriarch of an organized crime dynasty transfers control to his reluctant son."
19  },
20  {
21    "title": "The Dark Knight",
22    "description": "Batman battles the Joker to save Gotham."
23  }
24 }
```

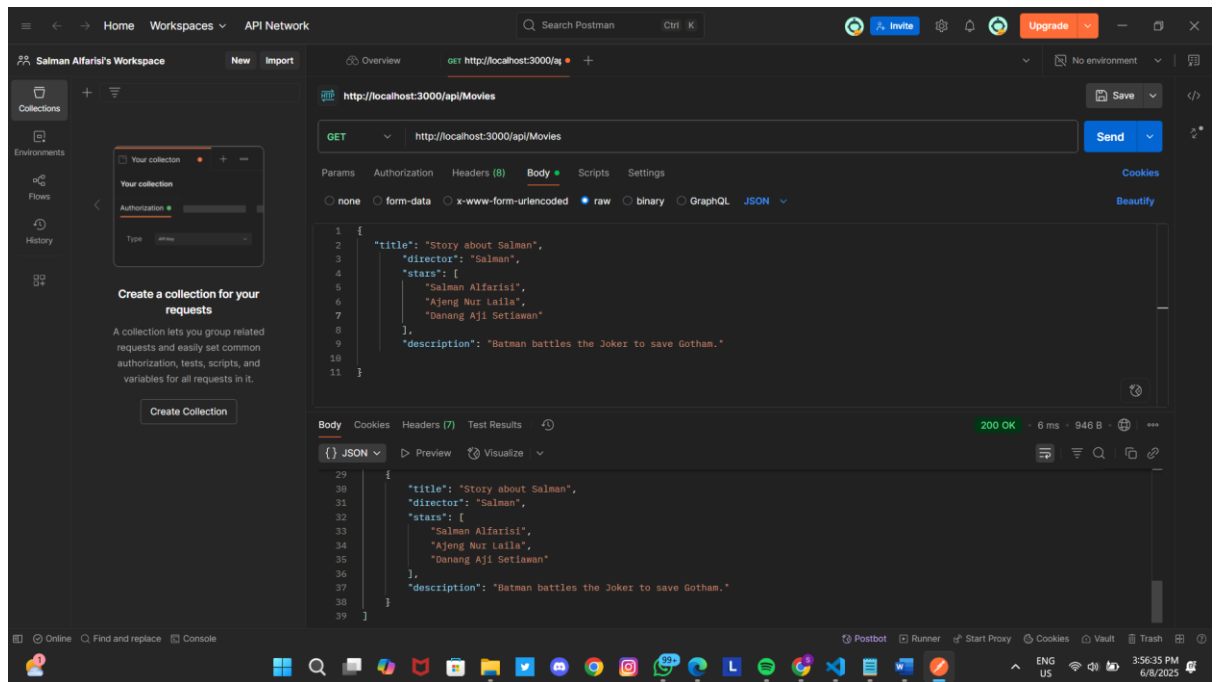
B. POST|

The screenshot shows the Postman interface with a POST request to `http://localhost:3000/api/Movies`. The request is saved, and the response is a 201 Created status with a 10 ms response time and 243 B of data. The response body is an HTML message indicating a movie was added.

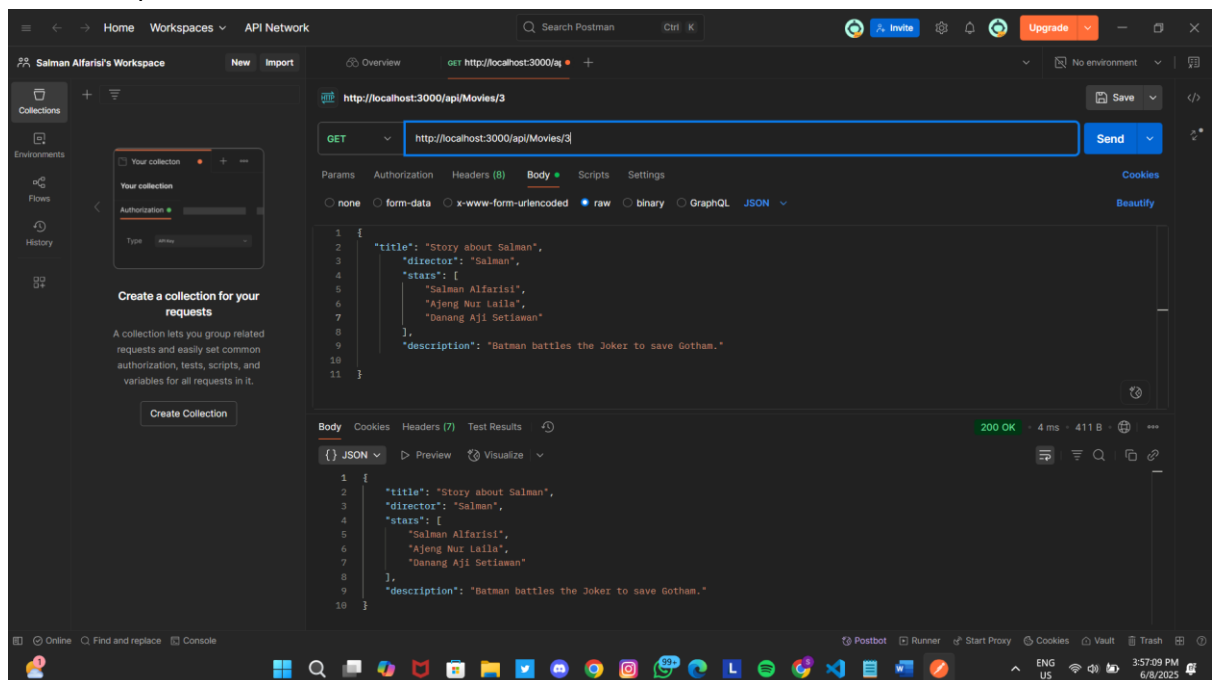
```
1 {
2   "title": "Story about Salman",
3   "director": "Salman",
4   "stars": [
5     "Salman Alfarisi",
6     "Ajeng Nur Laila",
7     "Danang Aji Setiawan"
8   ],
9   "description": "Batman battles the Joker to save Gotham."
10 }
```

1 Movie added

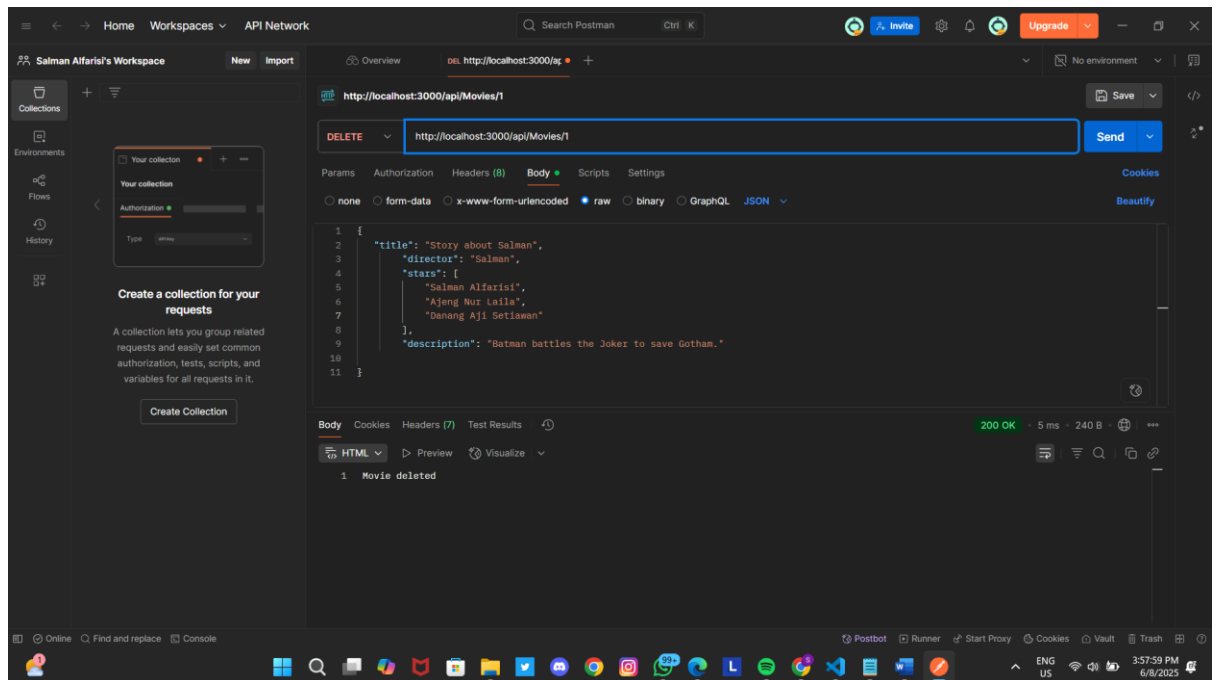
C. Cek



D. Cek Get/api/movies/3



E. Delete/api/api/movies/1



F. Cek

