

Midterm - 203

หน้าเว็บ

LIST CREATE MOVIE

Create Movies

Title

Genre

Director

Release

Create

Movies Created

1

Get all Movies

Movies

Title Movie

Search

Movie Id	Title	Genre	Director	Release
1	Joker	psychological thriller	Todd Phillips	2019

เพิ่มหนัง

LIST CREATE MOVIE

Create Movies

Title

Genre

Director

Release

Create

Movies Created

3

Get all Movies

Movies

Title Movie

Search

Movie Id	Title	Genre	Director	Release
1	Joker	psychological thriller	Todd Phillips	2019
2	The Grinch	comedy	Scott Mosier	2018
3	The Wind Rises	Drama style	Hayao Miyazaki	2013

ฐานข้อมูล

title	genre	director	release_year
Joker	psychological thriller	Todd Phillips	2019
The Grinch	comedy	Scott Mosier	2018
The Wind Rises	Drama style	Hayao Miyazaki	2013
NULL	NULL	NULL	NULL

Code

```
const onChangeForm = (e) => {  
  if (e.target.name === "title") {  
    movie.title = e.target.value;  
  } else if (e.target.name === "genre") {  
    movie.genre = e.target.value;  
  } else if (e.target.name === "director") {  
    movie.director = e.target.value;  
  } else if (e.target.name === "release_year") {  
    movie.release_year = e.target.value;  
  }  
  setMovie(movie);  
};
```

เริ่มตั้งแต่ปรับปรุง App.js

แก้ไขตัวแปร title , genre , director , release_year ให้ตรงกับ column ใน db

MovieService.js

```
export async function getAllMovies() {  
  try {  
    const response = await fetch("http://localhost:4000/api/movie/all");  
    return await response.json();  
  } catch (error) {  
    return [];  
  }  
}  
  
export async function createMovie(data) {  
  const response = await fetch("http://localhost:4000/api/movie/insert", {  
    method: "POST",  
    headers: { "Content-Type": "application/json" },  
    body: JSON.stringify({ ...data }),  
  });  
  return await response.json();  
}
```

แก้ url จาก /api/movie เป็น url localhost port 4000 ตามด้วยหน้าต่างๆ ในวงเล็บจาก movie , data เป็น ... data เพราะต้องการส่งแค่ data ที่สร้างใหม่จาก frontend ไปยังเซิร์ฟเวอร์ ผ่าน api

Movies.js

```
export const Movies = ({ movies }) => {  
  const [searchText, setSearchText] = useState('');  
  const [searchResults, setSearchResults] = useState([]);  
  const [isSearching, setIsSearching] = useState(false);  
  
  const handleSearch = async () => {  
    try {  
      const response = await fetch(`http://localhost:3001/api/movie/search?search_text=${searchText}`);  
      const data = await response.json();  
  
      if (data.returnCode === 1) {  
        setSearchResults(data.data);  
        setIsSearching(true);  
      } else {  
        setSearchResults([]);  
        setIsSearching(false);  
      }  
    } catch (error) {  
      console.error('Error searching for movies:', error);  
    }  
  };  
};
```

สร้างตัวแปร searchText มาเก็บข้อความที่ใช้ค้นหาหนัง

สร้าง searchResults เก็บผลลัพธ์ที่ได้จากการค้นหา

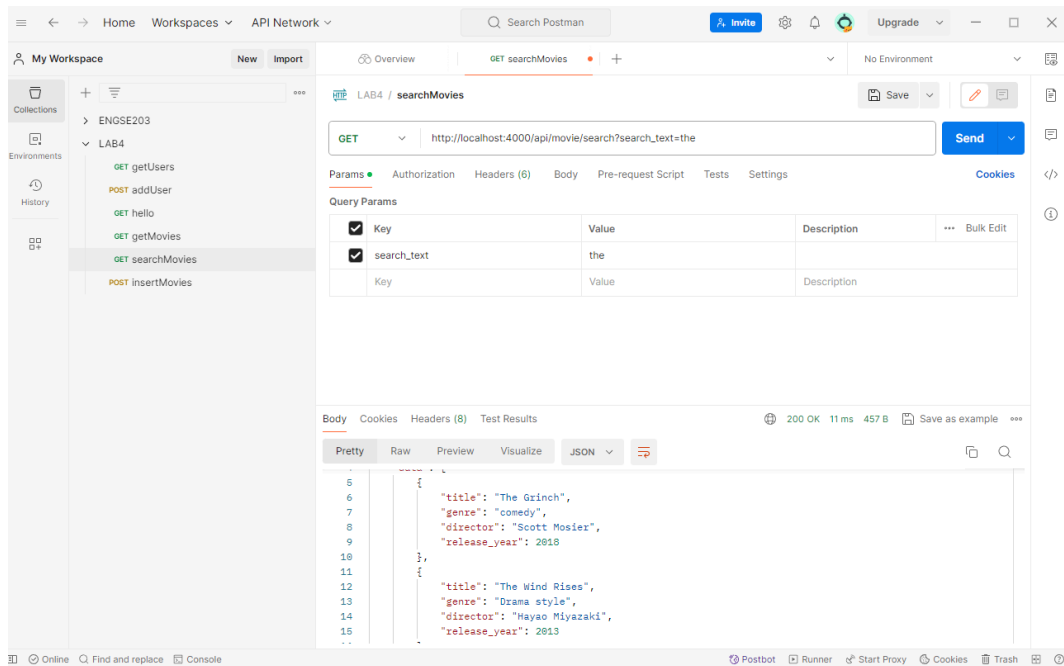
สร้าง isSearching เพื่อเป็นสถานะบ่งชี้ว่า functions กำลังทำงานอยู่หรือไม่

handleSearch

```
return (  
  <div className="container">  
    <div style={{ display: "flex", justifyContent: "flex-start", alignItems: "center" }}>  
      <div>  
        <h2>Movies</h2>  
        <div style={{ display: "flex", alignItems: "center" }}>  
          <input  
            type="text"  
            className="Search_input"  
            style={{  
              padding: "10px",  
              fontSize: "16px",  
              border: "1px solid #ccc",  
              borderRadius: "5px",  
              width: "300px",  
              marginRight: "10px",  
            }}  
            placeholder="Title Movie"  
            value={searchText}  
            onChange={(e) => setSearchText(e.target.value)}  
          />  
          <button  
            className="btn btn-info"  
            style={{ padding: "10px", borderRadius: "5px" }}  
            onClick={handleSearch}  
          >  
            Search  
          </button>  
        </div>  
      </div>  
    </div>  
    <hr />  
  
    <table className="table table-dark table-striped">  
      <thead className="table-dark">  
        <tr>  
          <th>Movie Id</th>  
          <th>Title</th>  
          <th>Genre</th>  
          <th>Director</th>  
          <th>Release</th>  
        </tr>  
      </thead>  
      <tbody>  
        {movieTable}  
        {isSearching && searchResults.length === 0 && <tr><td colspan="5">No movies found</td></tr>}  
      </tbody>  
    </table>  
  </div>  
)
```

กดปุ่ม search ฟังก์ชันนี้จะ fetch ข้อมูลจาก api โดยใช้ข้อมูลจากผู้ใช้ใส่เข้าไปใน input ถ้าหาเจอ สถานะจะเป็น true ถ้าไม่ใช่จะเป็น false เพิ่ม input และ ปุ่ม search

ทดสอบการทำงาน



ทดสอบใน api

