**Characters API output screenshots:**

**Create:**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**Read:**

**A screenshot of a computer

Description automatically generated**

**Update: A blue circle with white text

Description automatically generated**

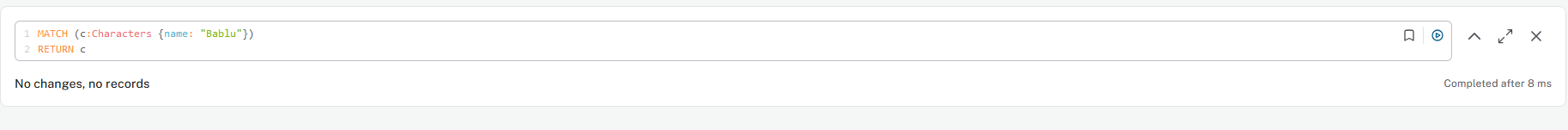
**A screenshot of a computer program

Description automatically generated**

**Delete:**

**A screenshot of a computer

Description automatically generated**

****

**Characters Controller:**

The `CharacterController` class is an ASP.NET Core API controller for managing characters. It defines endpoints for creating, updating, deleting, and retrieving characters using a `CharacterRepository`. The endpoints handle HTTP POST, PUT, DELETE, and GET requests, returning appropriate responses based on the repository's operations.

**A screenshot of a computer program

Description automatically generated**

**Repository :**

The `CharacterRepository` class manages CRUD operations for `Characters` in a Neo4j database using the Neo4j.Driver. The `CreateCharacterAsync` method creates a new character node. The `UpdateCharacterAsync` method updates an existing character's properties based on the character's name. The `DeleteCharacterAsync` method deletes a character node identified by its name. The `GetCharacterAsync` method retrieves a character node by its name and maps it to a `Characters` object. Each method uses `ExecuteWriteAsync` or `ExecuteReadAsync` to perform database operations in a transactional context, ensuring proper session handling and error management with logging for any exceptions.

**A screen shot of a computer program

Description automatically generated**

**A screen shot of a computer program

Description automatically generated**

**A screen shot of a computer program

Description automatically generated**

**Neo4j Database:**

Created and configured the StarWars datases in Neo4j database.

**A diagram of different species

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**Steps to run the project locally:**

* Open solution file using Visual Studio 2019 or above community version.

Add Dependencies:

* Make sure you install Dot Net 8.0
* Right-click on the project in Solution Explorer.
* Select Manage NuGet Packages.
* Install the following packages:
* Neo4j.Driver
* Microsoft.Extensions.Logging
* Swashbuckle.AspNetCore (for Swagger)

Build & Run

* Build and run the visual studio project.
* Swagger will popup automatically and you can see the crud operations.