

C#.NET ASSESSMENT 6

ASSESSMENT 6 - UNIT 8: ENTITY FRAMEWORK

Core concepts: Entity Framework

OVERVIEW

Implement an ASP.NET MVC web application that displays several countries from a database and allows the user to search countries by continent. The application should use the Entity Framework Core Database First approach.

SETUP

Use the ASP.NET MVC Web Application provided in the assessment repository.

BUILD SPECIFICATIONS

The assessment is worth ten points, one for each of the test cases below. **Pay special attention to the spelling and capitalization of the items in bold.**

Use the following SQL Query to make a database and table for you to connect to:

```
CREATE DATABASE CountriesDB;
USE CountriesDb
CREATE TABLE Country(
    Id int IDENTITY(1,1) PRIMARY KEY NOT NULL,
    [Name] nvarchar(40) NOT NULL,
    Continent nvarchar(40) NOT NULL,
);
INSERT INTO Country
VALUES('China', 'Asia'),
('India', 'Asia'),
('USA', 'North America');
```

Within SSMS, create a database named **CountryDb** and a table named **Country** with the following datatypes:

- **Id** (Primary Key) int, auto increment with Identity Specification
- **Name**, nvarchar(40)
- **Continent**, nvarchar(40)

Fill in the following into the table:

Id	Name	Continent
1	China	Asia
2	India	Asia
3	USA	North America

Now build the app:

1. Using Entity Framework, use database first to bring the Countries table into your **Assessment6** project. Call the item **CountriesDb** when prompted. Note: You'll be graded on what's in the **HomeController**, but you are allowed to create another controller to help with populating your table.

Inside your **HomeController**, your web application will have the following views and actions:

2. **Countries** action
 - a. Grab the list of countries from the Database and feed it into the view
3. **Countries** view will display the following:
 - a. The names of all the countries in the database.
 - b. The associated continent of each country displayed next to the name.
 - c. The names and countries must be displayed in an HTML unordered list like the following:
 - **China - Asia**
 - **India - Asia**
 - **USA - North America**
 - d. A form that allows the user to type in a continent and submit the form to the **CountriesByContinent** ActionResult.
4. **CountriesByContinent** action will have the following:
 - a. This action will take a in string for ContinentName
 - b. Also in the action, use that parameter to filter the countries list
5. **CountriesByContinent** view will display the following:
 - a. At the top of the view, display the selected continent
 - b. Next, display the name of each country on the continent

SUBMISSION

When finished, push your changes to the same GitHub repository you cloned your project from.