Code Documentation
Alexandros Chatziandreou
Viva Assessment

Overview

The `VivaProjectRepository` class is a repository responsible for interacting with both a cache and a database. It handles storing and retrieving data from a memory cache and performing CRUD operations on a SQL Server database using Dapper for database interactions.

Namespaces Used

- `Dapper`: An ORM (Object-Relational Mapper) for database interactions, simplifying data queries.
- `Microsoft.Extensions.Caching.Memory`: Used for in-memory caching to improve performance by reducing database calls.
- `System.Data.SqlClient`: SQL client to manage SQL Server database connections.
- `System.Linq`: LINQ for data querying and transformation.
- `System.Threading.Tasks`: For asynchronous operations.

Database Operations Region

Insert Data into Database

```
public async void InsertData(List<RestCountriesModel> data)
```

Description: This method inserts data into the `VIVA_COUNTRIES_TABLE` in the SQL Server database. It converts lists of capitals and borders into comma-separated strings for storage.
Parameters:

`data`: A list of `RestCountriesModel` to be inserted into the database.

Operation: The method executes an `INSERT` SQL query for each row in the provided list using Dapper's `ExecuteAsync` method.

Get Data from Database

public async Task<List<RestCountriesModel>> GetDataAsync()

Description This method retrieves all data from the `VIVA_COUNTRIES_TABLE` in the SQL Server database.
Returns: A list of `RestCountriesModel` objects mapped from the `CountryDbModel` retrieved from the database.
Mapping:
- The `Cca2`, `Name`, `Borders`, and `Capitals` fields from the database are mapped to the `RestCountriesModel`.

`Borders` and `Capitals` are split into lists of strings.

Database Connection

The class uses the connection string:
    private readonly string _connectionString =
"Server=localhost;Database=master;Trusted_Connection=True;";
This connection string is used for connecting to the local SQL Server instance, specifically the `master` database. It should be updated for production environments with appropriate security measures.

Cache Memory Region

Store Data in Cache

public void StoreDataInCache(string key, List<RestCountriesModel> data)

Description: This method stores a list of `RestCountriesModel` in memory cache.
Parameters:
- `key`: A unique identifier for the cache entry.
- `data`: A list of `RestCountriesModel` to be cached.

Cache Expiration: The cache is set with a sliding expiration of 10 minutes.

Get Data from Cache

public List<RestCountriesModel> GetDataFromCache(string key)

Description: This method retrieves data from the memory cache by its key.
Parameters:
 `key`: The key associated with the cache entry.
Returns: Returns the list of `RestCountriesModel` from the cache if found, or `null` if the cache entry does not exist.

Data Models

RestCountriesModel

Description: A model representing the structure of a country, including its name, abbreviation, borders, and capital cities.
Properties:
 - `Name`: An object representing the country's name.
 - `Cca2`: The country code (e.g., "US").
 - `Capital`: A list of capital cities (e.g., `[\"Washington, D.C.\"]`).
 - `Borders`: A list of countries that share borders with the country (e.g., `[\"CAN\", \"MEX\"]`).

CountryDbModel

Description: A database model representing the structure of country data in the `VIVA_COUNTRIES_TABLE`.
Properties:
 - `Cca2`: The country code.
 - `Name`: The country's name.

 - `Borders`: Comma-separated string of border countries.
 - `Capitals`: Comma-separated string of capital cities.