CUSTOM CACHE

Ultimate C# Masterclass Assignment

Overview

The purpose of this assignment is to create a Cache class that will provide a caching mechanism for data of any type, identified by a key of any type.	Library

Problem outline

```csharp
public class SlowDataDownloader : IDataDownloader
{
    8 references
    public string DownloadData(string resourceId)
    {
        //let's imagine this method downloads real data,
        //and it does it slowly
        Thread.Sleep(1000);
        return $"Some data for {resourceId}";
    }
}
```

This class is used to fetch some data identified by a string ID. Unfortunately, each execution of the DownloadData method takes around 1 s. If the program often needs to fetch the data identified by the same IDs...

```csharp
Console.WriteLine(dataDownloader.DownloadData("id1"));
Console.WriteLine(dataDownloader.DownloadData("id2"));
Console.WriteLine(dataDownloader.DownloadData("id3"));
Console.WriteLine(dataDownloader.DownloadData("id1"));
Console.WriteLine(dataDownloader.DownloadData("id3"));
Console.WriteLine(dataDownloader.DownloadData("id1"));
Console.WriteLine(dataDownloader.DownloadData("id2"));
```

... it will work slowly because each operation will take 1 s. To make it work faster, we could cache the data. So, for example, after the data identified by ID "id1" is fetched for the first time, the next time, it could be served from the cache.

Acceptance criteria

1.   A generic Cache class is defined and used. It should be able to work for any type of key and data, not only strings like here.
2.   The first time we fetch the data identified by a certain key, it still works slowly. But the next time this data is fetched it should be served from the cache, so it should happen almost immediately.