

Indhold

Code Test	1
Palindrome Challenge	1
API Usage Challenge.....	1
Threading Challenge	2

Code Test

Complete the following 3 assignments using the following programs and technology

- Visual Studio (free community version can be downloaded here <https://visualstudio.microsoft.com/downloads/>)
- .Net (.Net 6 is latest stable version)
- C#

All three projects must be in one Visual Studio solution and be able to compile and run from the solution.

Upon completion projects can either be uploaded to Github or send the link by mail to a.wais@rn.dk

If uploaded to GitHub the project must be public or you must add the user a.wais@rn.dk / Omega3Vitamin as a collaborator to the project.

General information

For every challenge add comments to explain your logic and design choices.

All challenges must handle errors – this means that if an error occurs there must be error handling and logging. The log can simple be write to console or you can implement a logger class in the solution.

Palindrome Challenge

Goal

Create a Console app that can identify if a word is a palindrome (the same forward and backwards like racecar). Create an overload method to check an int to see if it is a palindrome.

Ignore spacing, casing, and special characters when identifying palindromes. For instance, “stack cats” is a palindrome.

API Usage Challenge

Goal

Create a Console app that will look up people in the SWAPI (<https://swapi.dev/>) based upon their ID number. Handle missing IDs/errors and load all the data for successful entries.

The data should be parsed into a class that represent the data structure from the API, so that it’s possible to access the properties on the class e.g. `PersonClass.Name` (and all other properties on “people” from the API) can be used like this `PersonClass.Name` (and all other properties on “people” from the API)

Create a cache (in memory) so that if you make the same API call again, the request will be fetched from the cache and not the API.

The cache should store the parsed class and not the raw response from the API.

Hint

RestSharp can be used to consume the API <https://json2csharp.com/> can generate classes from raw JSON

Threading Challenge

Goal Create a console app that uses threading to process all items in a list of strings (min 20 items in list). In this challenge you must use SemaphoreSlim class. There is only allowed to run a max of 3 simultaneous threads at one time to simulate limited resources.

The process function can simply write the string to console or possible call the swapi API used in the last challenge and output the data from the API call.

There should be inserted a random delay in the process function between 1-5 seconds to simulate actual processing of data.

Hint SemaphoreSlim Class – info here <https://docs.microsoft.com/en-us/dotnet/api/system.threading.semaphoreslim?view=net-6.0>