**Coding Test Guidelines**

There are 4 simple challenges to complete.

In the visual studio solution there are three projects.

1. The SkeletonCode project contains the classes you are required to edit to complete the C# challenges.
2. The UnitTest project contains some simple unit tests to help you complete the C# tasks. The NUnit test runner is included as an alternative to the visual studio test runner.
3. The JasmineTests website contains an index.html file that will execute all the JavaScript unit tests. You will only need to update the content in the “scripts” folder to complete the JavaScript challenges.

The challenges require the .NET Framework 4.5 to compile and the JavaScript challenges should support IE9.

**StringUtilites**

This class has one failing test.  
Edit the code to make it pass.  
You may wish to improve the performance of the algorithm with large strings.

**Currency Converter**

The currency converter implementation is incomplete.  
It should take in two ISO codes that represent the currencies it should convert to and from.  
The implementation should be able to convert between USD and GBP at a rate of 1 GBP = 1.25 USD.  
An exception should be thrown if an unrecognised code is passed in.  
There are some basic unit tests that should pass once your implementation is complete.

**CardGame**

Please finish the implementation of PackOfCardsCreator and create implementations of IPackOfCards and ICard.

The PackOfCardsCreator should create a standard pack of cards. This should be made up of 52 cards, with 4 different suits (Clubs, Hearts, Spades and Diamonds) and numbered 2, 3, 4, 5, 6, 7, 8, 9, 10, Jack, Queen, King, and Ace.

You are free to do this however you like.

IPackOfCards.Shuffle() should rearrange the cards in a random order. Repeated shuffles should not all return the cards in the same order.

IPackOfCards.TakeCardFromTopOfPack() should return and remove the first card from cards in the pack.

There are some unit tests that should pass after you have finished writing your classes. Please be aware that they do not cover all the expected behaviour of the classes.

**JasmineTests**

The final test is a set of string extensions written in JavaScript. Open “JasmineTests/index.html” to run the unit tests. Then complete the implementation of all the extension functions in “JasmineTests/scripts/string-extensions.js” so that all the unit tests pass.

Spend some time to improve the quality of the JavaScript, ensuring industry standard best practices.