Programming in Java

Coursework: Data Structures

See the Moodle site for the due date

Aims

The aims of this coursework are to gain practice with:

- working against an interface; and
- dynamic data structures by implementing some of their most commonly used methods; and
- arrays, linked lists, and stacks by writing examples of higher level methods

Requirements

Your `github` repo must be named "`cw-ds`".

All interfaces referred to in this document are provided. Every class that implements an interface must follow the convention *InterfaceImpl* except when noted otherwise in the text. For example, the implementation of interface `ReturnObject` must be called `ReturnObjectImpl`. Do not make any change to the provided Java files: they will be overwritten when the coursework is graded anyway.

You must not use any complex type to provide a solution to this coursework with the following exceptions: boxed types, String, arrays, and any other complex type that you define yourself are allowed. You cannot use third-party libraries either, or any class from the Java library with the exceptions noted above. Do not use generics either, you do not need them.

In order to complete this coursework, you have to:

1. Write a class that implements interface `ReturnObject`.
2. Write an implementation of interface `List` based on arrays called `ArrayList`.
3. Write an implementation of interface `List` based on pointers called `LinkedList`.
4. Write an implementation of interface `FunctionalList` that extends `ArrayList` called `FunctionalArrayList`.
5. Write an implementation of interface `FunctionalList` that extends `LinkedList` called `FunctionalLinkedList`.

6. Write an implementation of interface `SampleableList`.

7. Write a class `StackImpl` that extends class `AbstractStack`.

8. Write an implementation of interface `ImprovedStack`. This class cannot extend either `AbstractStack` or `StackImpl`.

Submission

We will clone your `github` repo at the due date and time.