# Outcome 3 (Part)

## Describe, develop and use abstract data types.

**Knowledge and/or Skills**

♦ Concept of an Abstract Data Type (ADT)

♦ Concept of a range of ADTs including Stack, Queue, Set, **List** and Map

♦ Develop Interfaces (method signatures) for a range of ADTs including Stack, Queue, Set, **List** and Map

♦ Implement a Stack, Queue and **List** using a given array data structure

For this assignment, you are required to implement a **Linked-list** data structure that performs all of the expected Stack operations. You should refer back to your original array-based working Stack, to remind you of the basic operations involved. You may use any data type as your sample elements.

Your Top Level should consist of a menu-driven program that will offer the following options –

1. Add an element to the top of the stack
2. Remove the top element from the stack
3. Report whether the stack is empty
4. View the entire contents of the stack
5. Exit the program

A stack should automatically be created and initialised when your main program is run. It should not be possible to remove an item from an empty stack. Note that there is no max size for a list-based stack. Your main program should call functions to perform the various tasks.

**Please ensure that standard stack nomenclature is used** and that the stack is based on a linked-list structure**, i.e elements are added as singly-linked nodes.** A file of starter code has been supplied, to help you get started.

The program must be thoroughly tested, including screenshots. You must include tests to add to a full stack or take from an empty one, and include a test to verify that your stack is empty when it is first created.

You are required to submit –

* Your completed LinkedStack class
* Your completed Top Level (main) program
* Your test plan and log, including screenshots, to verify that all operations work correctly

Good luck!