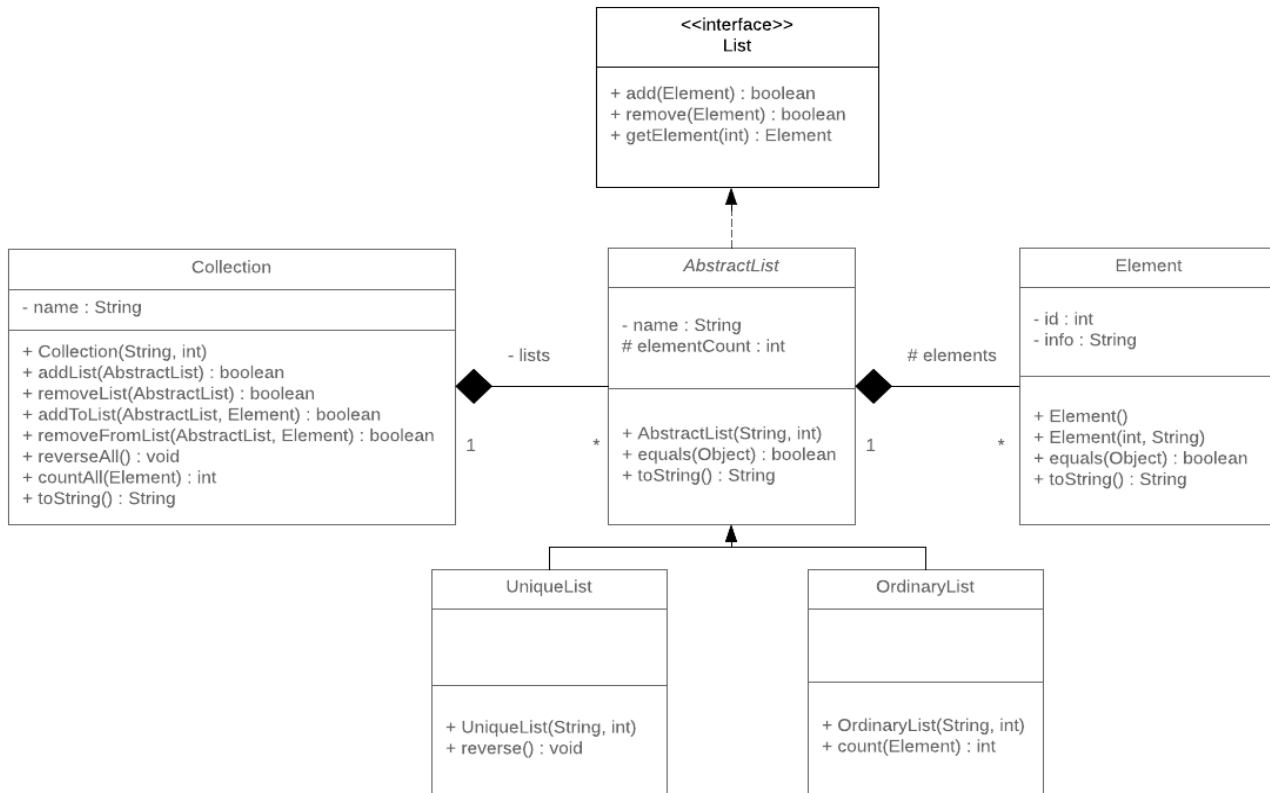


King Saud University
 College of Computer and Information Sciences
 Department of Computer Science
 CSC113 – Computer Programming II –Interface Lab – Fall 2019



Element class:

- **Attributes:**
 - **id**: the id of the element
 - **info**: the info of the element
- **Methods:**
 - **Element()**: default constructor
 - **Element(int id, String info)**: constructor
 - **equals(Object other)**: compares two objects of type **Element** based on their **id** and returns the result of the equality
 - **toString()**: returns a string representing the element

List interface:

- **Methods:**
 - **add(Element e)**: adds **e** to the list in the following ways:
 - **UniqueList**: adds **e** if there's space and it isn't already on the list
 - **OrdinaryList**: adds **e** if there's space
 - **remove(Element e)**: removes **e** from the list in the following ways:
 - **UniqueList**: removes **e** if it's in the list while maintaining the order
 - **OrdinaryList**: removes **e** if it exists in the list and replaces it with the last element if possible
 - **getElement(int index)**: returns the element at **index** if possible

AbstractList class:

- **Attributes:**
 - ***name***: the name of the list
- **Methods:**
 - ***AbstractList(String name, int size)***: constructor
 - ***equals(Object other)***: compares two objects of type ***AbstractList*** based on their ***name*** and returns the result of the equality
 - ***toString()***: returns a string representing the list and its elements

UniqueList class:

- **Methods:**
 - ***UniqueList(String name, int size)***: constructor
 - ***reverse()***: reverses the order of the elements in the list. Example: (1st, 2nd, 3rd, ..., ith → ith, i-1th, i-2th, ..., 1st) do this **in-place**

OrdinaryList class:

- **Methods:**
 - ***OrdinaryList(String name, int size)***: constructor
 - ***count(Element e)***: returns the number of occurrences of the element ***e*** in the list

Collection class:

- **Attributes:**
 - ***name***: the name of the collection
- **Methods:**
 - ***Collection(String name, int size)***: constructor
 - ***addList(AbstractList l)***: adds the list to the collection if there's space and it's not already there
 - ***removeList(AbstractList l)***: removes the list ***l*** from the collection if it is in the collection and replaces it with the last list
 - ***addToList(AbstractList l, Element e)***: adds ***e*** to list ***l***
 - ***removeFromList(AbstractList l, Element e)***: removes ***e*** from the list if possible
 - ***reverseAll()***: reverses all the unique lists in the collection
 - ***countAll(Element e)***: returns the number of occurrences of the element ***e*** in all the ordinary lists in the collection
 - ***toString()***: returns a string representing the collection and its lists

Exercise 1: Translate into Java code all the previous classes & interfaces.

Exercise 2: Write a main method and test all the previous functionalities.