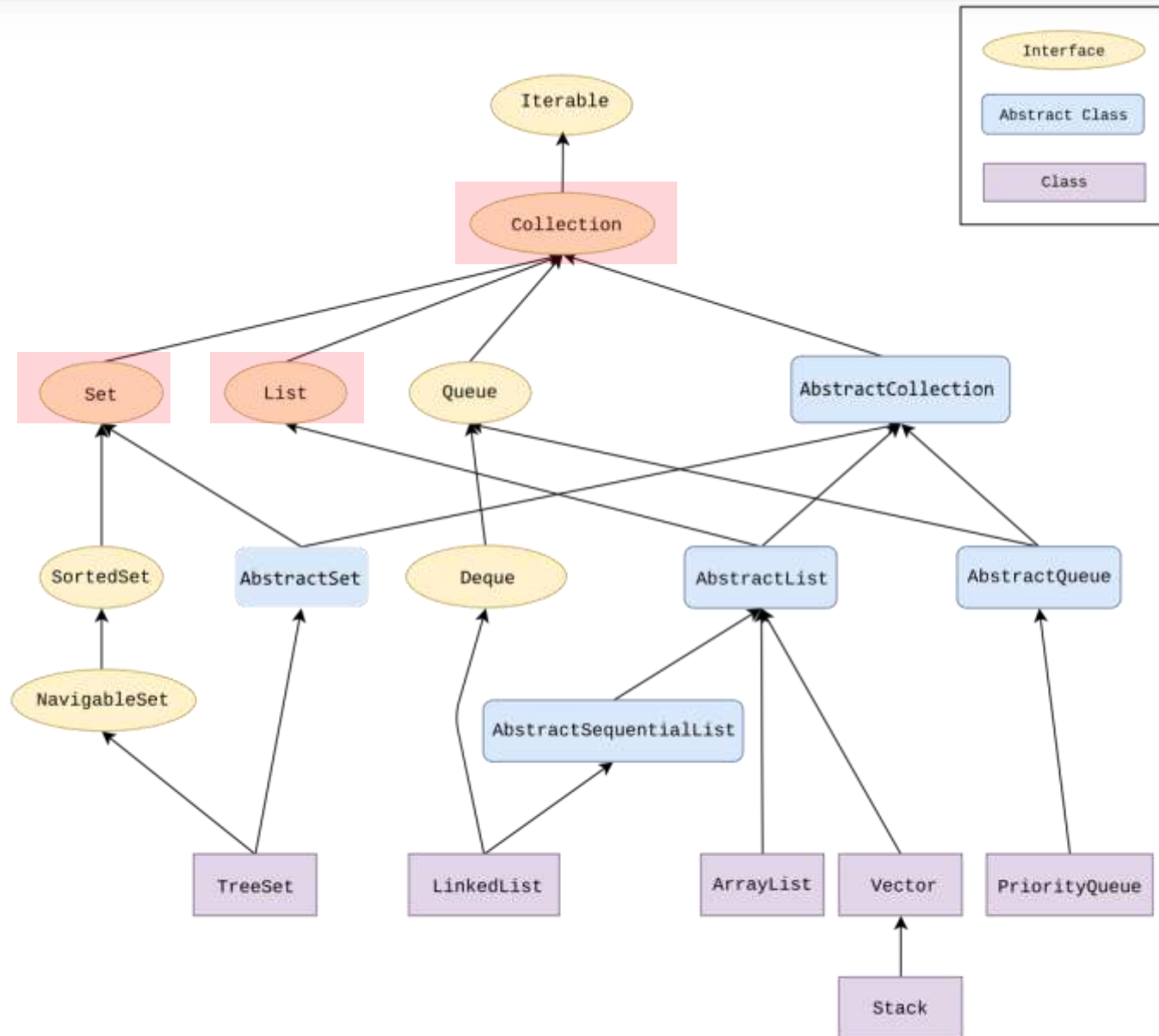# Selenium WebDriver Training

List

Array Vs Collection

- An array is a collection of similar types of data.
- Limitation: Size is fixed.

- What if I do not know size of the data array upfront?

- What if I like to increase or decrease the size at runtime?

- That can be achieved using Collection (Dynamic Array)

Testleaf
Always Ahead

Collection

List

- Interface (Sub Interface of Collection)
- Since List is an interface, we cannot create objects from it

Testleaf
Always Ahead

List

- Interface (Sub Interface of Collection)
- Since List is an interface, we cannot create objects from it
- In order to use functionalities of the List interface, we can use any of the implementation classes like ArrayList

Testleaf
Always Ahead

List Properties

- List interface is an ordered collection that allows us to store and access elements sequentially
- List allows duplicate values inside the collection

Syntax

```
List<String> mentors = new ArrayList<String>();
```

Syntax

```
List<String> mentors = new ArrayList<String>();
```

Interface

Syntax

```
List<String> mentors = new ArrayList<String>();
```

Generic (Type)

Syntax

```
List<String> mentors = new ArrayList<String>();
```

List Name

Syntax

```
List<String> mentors = new ArrayList<String>();
```

Cannot create object for List

Syntax

```
List<String> mentors = new ArrayList<String>();
```

The implementation class of List

Syntax

```
List<String> mentors = new ArrayList<String>();
```

What happens here?

All the methods of ArrayList is not exposed in the mentors. **Only the methods of List is exposed (only showing the required implementation)**

Testleaf
Always Ahead

List Methods

- add() - adds an element to a list

List Methods

- add() - adds an element to a list
- addAll() - adds all elements of one list to another

List Methods

- add() - adds an element to a list
- addAll() - adds all elements of one list to another
- get() - get particular element based on index

List Methods

- add() - adds an element to a list
- addAll() - adds all elements of one list to another
- get() - get particular element based on index
- remove() - removes an element from the list

List Methods

- add() - adds an element to a list
- addAll() - adds all elements of one list to another
- get() - get particular element based on index
- remove() - removes an element from the list
- removeAll() - removes all the elements from the list

List Methods

- add() - adds an element to a list
- addAll() - adds all elements of one list to another
- get() - get particular element based on index
- remove() - removes an element from the list
- removeAll() - removes all the elements from the list
- clear() - removes all the elements from the list

Testleaf
Always Ahead

List Methods

- add() - adds an element to a list
- addAll() - adds all elements of one list to another
- get() - get particular element based on index
- remove() - removes an element from the list
- removeAll() - removes all the elements from the list
- clear() - removes all the elements from the list
- size() - returns the length of lists

List Methods

- add() - adds an element to a list
- addAll() - adds all elements of one list to another
- get() - get particular element based on index
- remove() - removes an element from the list
- removeAll() - removes all the elements from the list
- clear() - removes all the elements from the list
- size() - returns the length of lists
- toArray() - converts a list into an array

Testleaf
A l w a y s   A h e a d

List Methods

- add() - adds an element to a list
- addAll() - adds all elements of one list to another
- get() - get particular element based on index
- remove() - removes a particular element from the list based on index
- removeAll() - removes all the elements from the list
- clear() - removes all the elements from the list
- size() - returns the length of lists
- toArray() - converts a list into an array
- contains() - returns true if list contains specified element

Testleaf
Always Ahead

- Demonstrate the index (order) and duplicates ..
- Also demonstrate to sort List !!

Testleaf
Always Ahead

List Problem Solving in Selenium

- Get Link Count (Leafground)
- Get Dropdown values (Leafground)

Summary

- List – **interface** and implemented by **ArrayList**

- List can have **duplicated** and **ordered** collection

- Selenium Example: **findElements**

Testleaf
Always Ahead

Classroom Exercise (Breakout)

- Go to https://amazon.in
- Type phones -> id: twotabsearchtextbox
- Print price of every phone -> class: a-price-whole
- Print the lowest price !

Hints:

- Look what is common locator for all price info of every phone.
- Use Collection sort to get the lowest price
- (make sure the stored values are in Integer to support sorting)
- Use Integer.parseInt to convert String to integer

Testleaf
Always Ahead