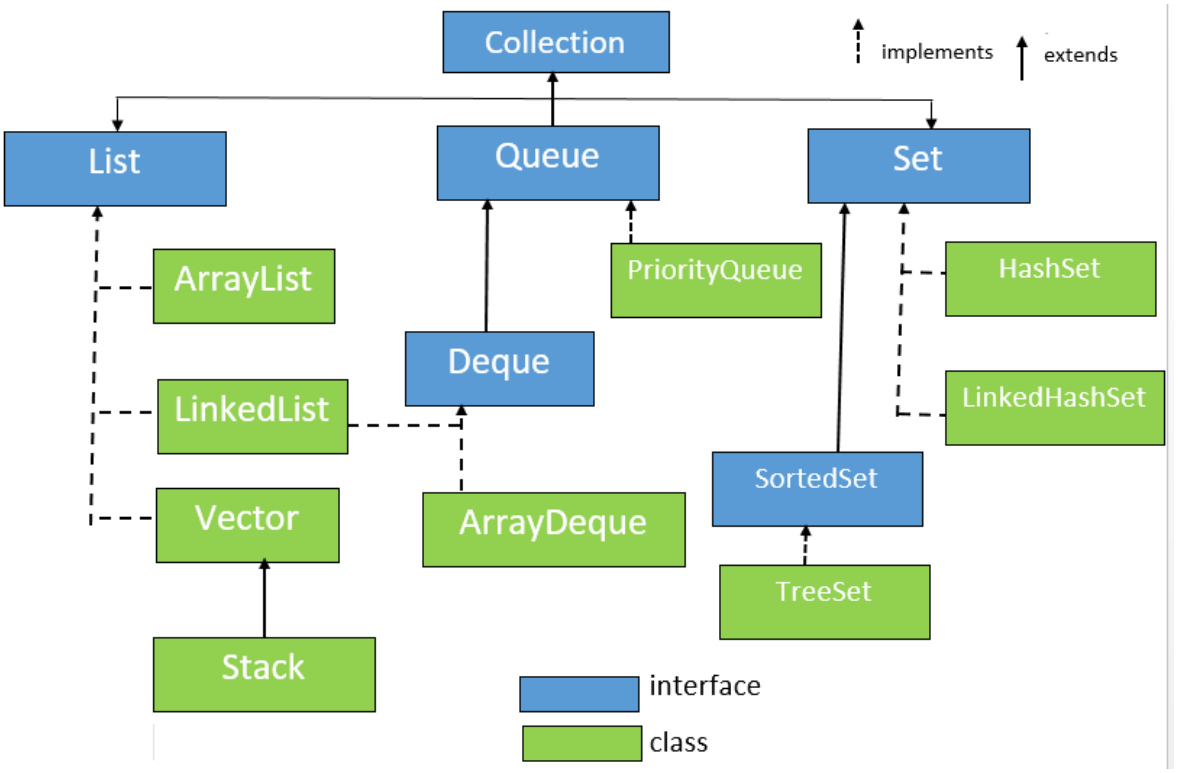
Collection Framework
==========

- It is the set of predefined classes and interface which is used to store the multiple data.

- **Collection**
  it is a single entity object where we can store multiple data.
- **Framework**
  It represents the library.

- Collection contains the following interfaces and classes.



- Collection is an interface in Java which is extended by three interfaces: List, Queue and Set.
- List is an interface which is implemented by three classes : ArrayList, LinkedList and Vector.
- Queue is an interface which is implemented by PriorityQueue class and also extends by Deque interface.
- Set is an interface which is extended by classes : HashSet and LinkedHashSet.
- Set is an interface which is also extended by Sortedset interface which is implemented by TreeSet.

Difference between List and Set
==========

List	Set
List is an index based data structure.	Set is not an index based data structure, it stores the data according to the hashcode.
List can store duplicate data.	Set does not allow storing duplicate elements.
List can store any number of null values.	Set can store only one null value.
List follows the insertion/sequence order.	Set does not follow insertion/ sequence order.
We can get the list elements by using Iterator.	We can get the set elements by using Iterator.

E.x List example

```
package testJava;

import java.util.ArrayList;
import java.util.Iterator;
import java.util.List;

class sclass
{
	public static void main(String[] args)
	{
		ArrayList l = new ArrayList();		// up casting

		l.add(null);
		l.add(null);
		l.add(30);
		l.add("class");

		Iterator it = l.iterator();

		while (it.hasNext()) {
			System.out.println(it.next());
		}
	}
}
```

```
Output :	null
		null
		30
		class
```

—----------

E.x Set example

```
package testJava;

import java.util.HashSet;
import java.util.Iterator;
import java.util.Set;

public class pclass
{
	public static void main(String[] args)
	{
		HashSet s = new HashSet();

		s.add(null);
		s.add(null);
		s.add(10);
		s.add("java");

		Iterator it = s.iterator();
		while(it.hasNext())
		{
			System.out.println(it.next());
		}
	}
}
```

Output : null
java
10

==========

Difference between Array and Collection
==========

Array	**Collection**
Arrays are fixed in size, once we create an array we cannot increase or decrease the size based on our requirement.	Collections are growable in nature i.e, based on our requirement we can increase or decrease the size.
With respect to memory array are not recommended.because of Memory wastage	With respect to memory collection are recommended.because of no memory wastage
Performance low	Performance high
Arrays can store only homogeneous/similar data types.	Collections can store homogeneous/similar and heterogeneous/different elements.
There are no ready-made methods	There are ready-made methods supported

supported by array.	by collections.
Arrays can hold both primitive and object types.	Collection can hold only object elements.