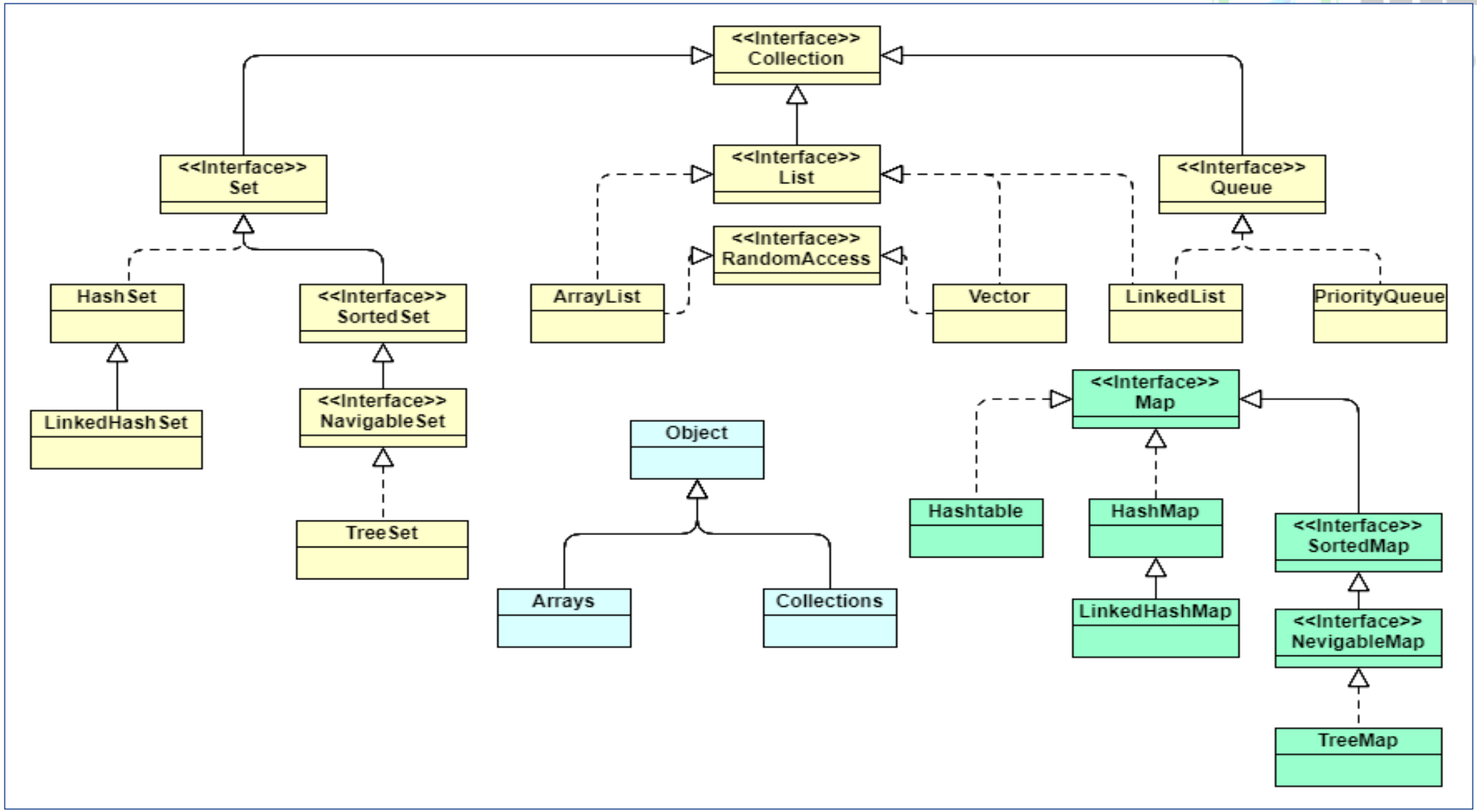
Short Introduction to Collections

Sawan Rai


<<Interface>>
Collection
<<Interface>>
Set
<<Interface>>
List
<<Interface>>
Queue
<<Interface>>
RandomAccess
HashSet
<<Interface>>
SortedSet
ArrayList
Vector
LinkedList
PriorityQueue
LinkedHashSet
<<Interface>>
NavigableSet
Object
<<Interface>>
Map
TreeSet
Arrays
Collections
Hashtable
HashMap
<<Interface>>
SortedMap
LinkedHashMap
<<Interface>>
NevigableMap
TreeMap

List: Ordering

int indexOf(Object o);

PDPM Jabalpur

ArrayList

- Random Acces
- Best for frequent insertion
- Not thread safe

Vector

- Thread safe version of ArrayList

LinkedList

- Doubly Linked
- Not good for iteration
- Good for insertion/deletion

```
public int indexOf(Object o) {
      if (o == null) {
         for (int i = 0; i < size; i++)
            if (elementData[i]==null)
               return i;
      } else {
         for (int i = 0; i < size; i++)
            if (o.equals(elementData[i]))
               return i;
      }
      return -1;
   }
```

```
public synchronized int indexOf(Object o, int
index) {
      if (o == null) {
         for (int i = index ; i < elementCount ; i++)
            if (elementData[i]==null)
               return i;
      } else {
         for (int i = index ; i < elementCount ; i++)
            if (o.equals(elementData[i]))
               return i;
      }
      return -1;
   }
```

```
public int indexOf(Object o) {
      int index = 0;
      if (o == null) {
         for (Node<E> x = first; x != null; x = x.next)
{
            if (x.item == null)
               return index;
            index++;
         }
      } else {
         for (Node<E> x = first; x != null; x = x.next)
{
            if (o.equals(x.item))
               return index;
            index++;
         }
      }
      return -1;
   }
```

Set: No duplicates

PDPM IIITDM

```
boolean add(E e);
```

HashSet	TreeSet	LinkedHashSet
• Not sorted • Not ordered	• Sorted • Ordered	• Not sorted • Ordered • Doubly linked

```
public boolean add(E e) {
        return map.put(e, PRESENT)==null;
    }
```

```
public boolean add(E e) {
        return m.put(e, PRESENT)==null;
    }
```

LinkedHashSet uses HashSet class “add”

method

Map: [Key: Value] pairing

HashMap	TreeMap	LinkedHashMap	Hashtable
• Not sorted • Not ordered • Efficient access	• Sorted • Ordered	• Not sorted • Ordered • Doubly linked	• Thread safe version of HashMap

Collection vs Collections

- Collection
  - Base Interface of List, Set and Queue.
  - add(), remove(), size(), isEmpty() etc…….
  - Is there any method with body inside the Collection ?? Check it
- Collections
  - Utility Class: provides some predefined methods for collection classes.
  - Methods: swap(), shuffle(), binarySearch() etc…….

Arrays

- Utility class having array ([]) related manipulation methods.