Collections

Session 7

Contents

- Need of collections
- Collection interfaces – List & Set

Why collections?

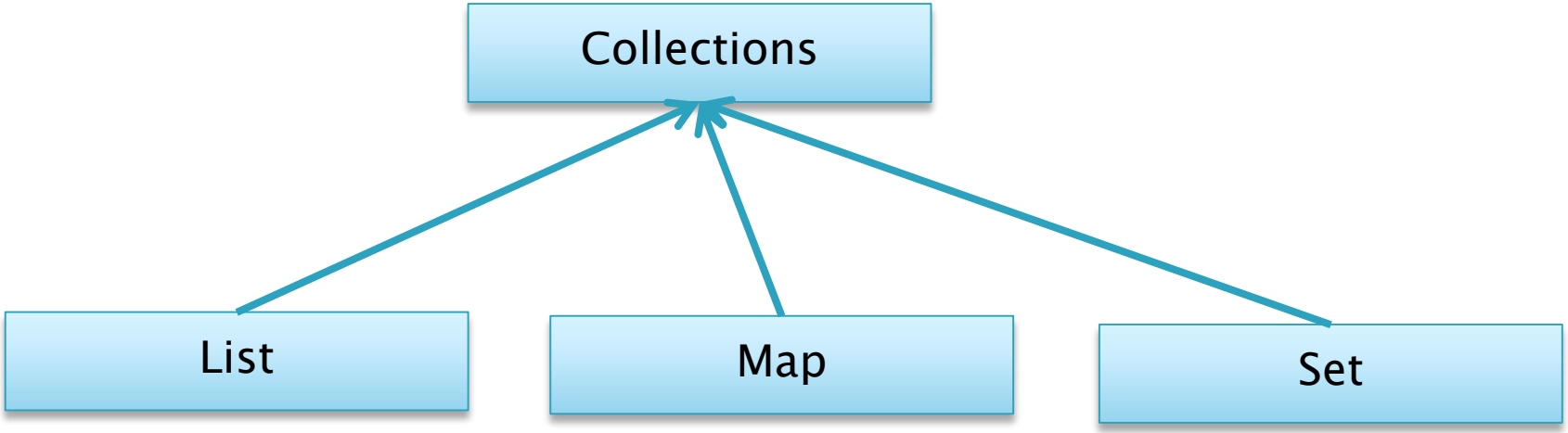
- Array size is fixed. Size can't be increased or decreased dynamically.
  - In actual development scenario, same type of elements need to be processed dynamically.

Why Collections?

- Data may be obtained from various data sources.
  - Values obtained may be single or multiple.
  - Depending upon number of values we need to change the method.

Collections

Collection Interface Hierarchy



Collection Interfaces

Interface	Descricption
List	Represents a collection of objects that is accessed via index. Ordered collection. Allows duplicates.
Set	Represents a collection that contains no duplicates. An order is not fixed.
Map	Represents a collection of key-value pairs.

List Interface

- Ordered collection.
- May contain duplicates.
- Implemented by classes
  - ArrayList
  - LinkedList
  - Vector

Set Interface

- Collection of unique elements.
- No duplicate elements are allowed.
- Collection is not any particular order.
- Implemented by
  - HashSet
  - LinkedHashSet
  - EnumSet
  - TreeSet

Thank
You!