

Day 6: collection api

List :ArrayList, LinkedList

Set: HashSet, LinkedHashSet

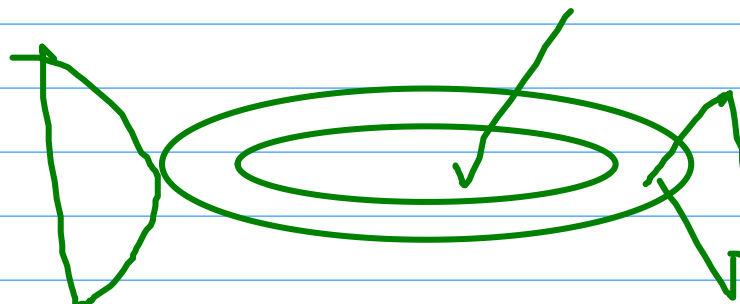
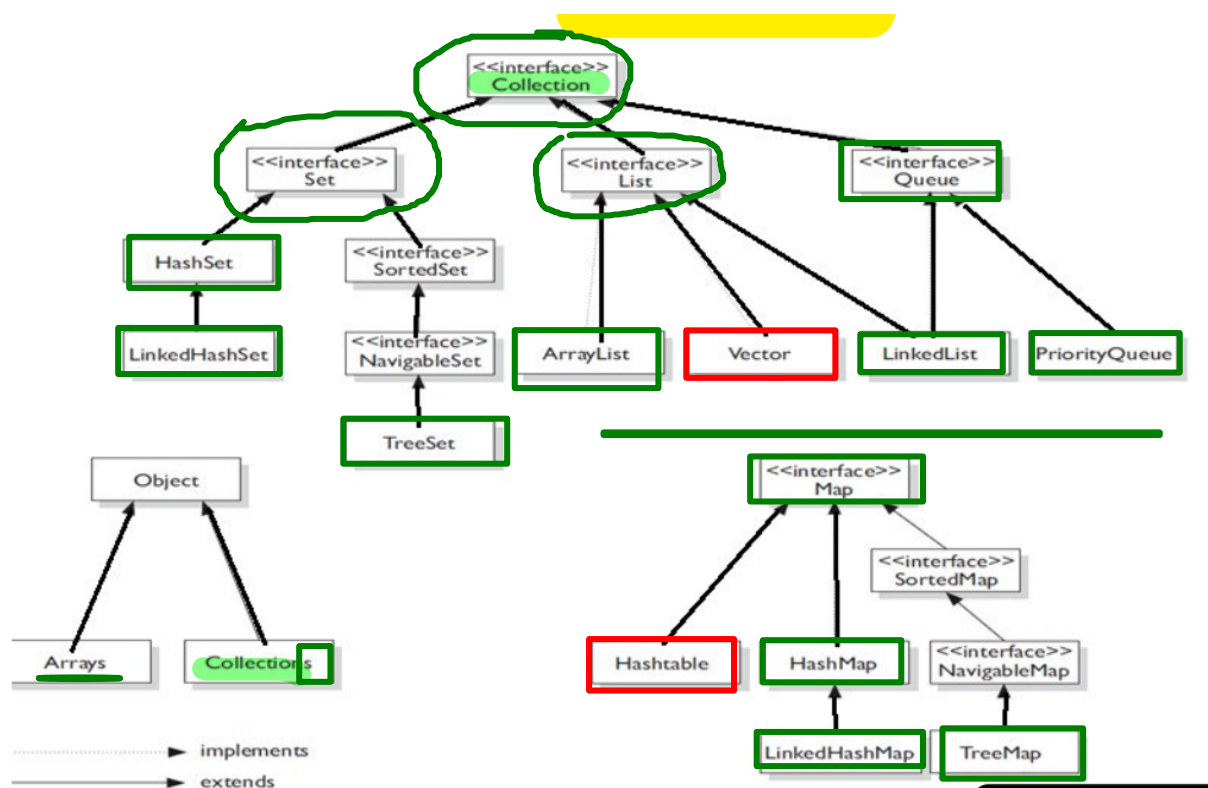
Map: HashMap, TreeMap, LinkedHashMap

Iterator

Arrays
Collections

Comparable , Comparable
Performance

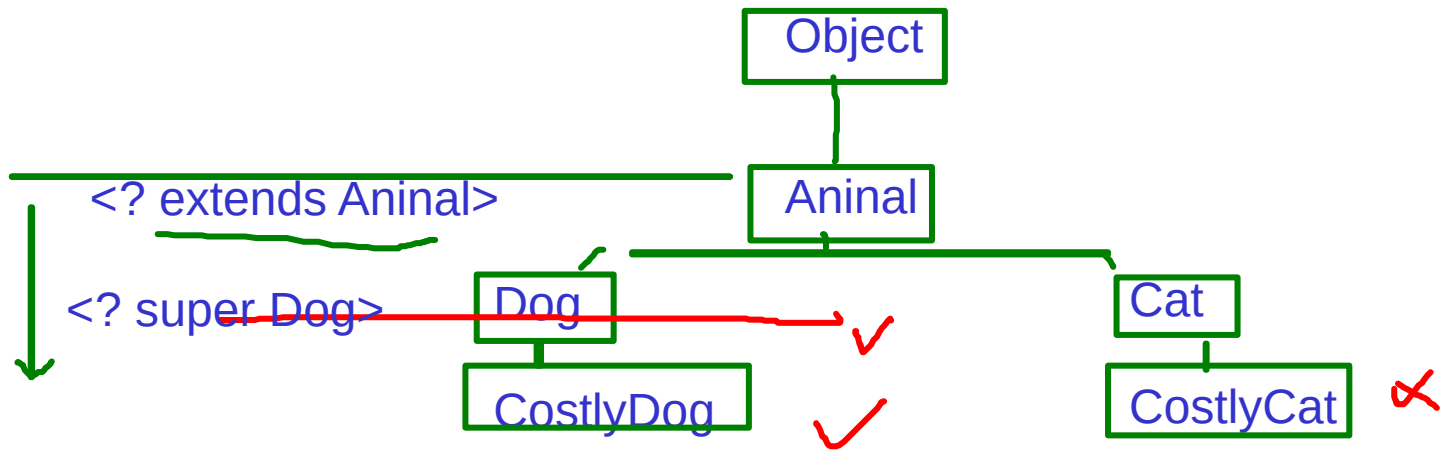
Generic, generics class generic method , wildcard



wildcard in generics

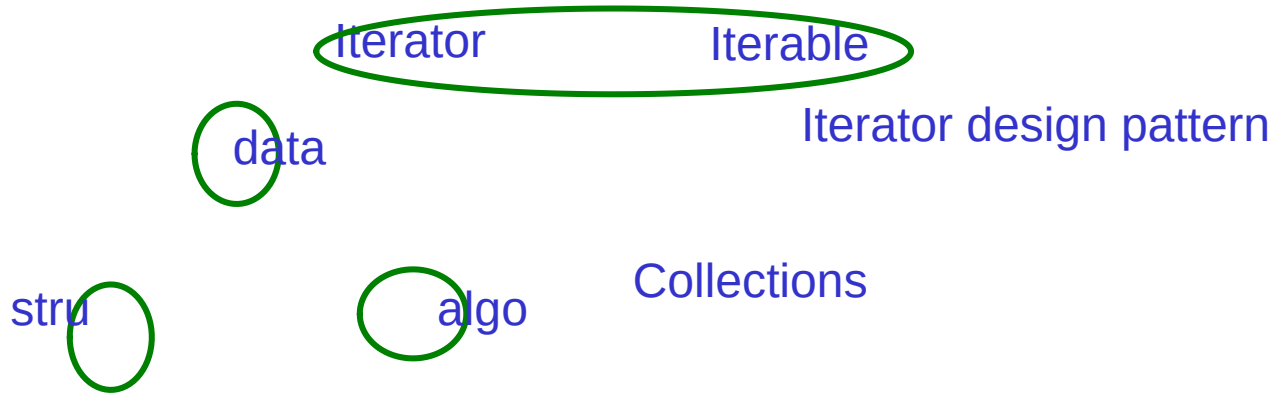
? extends XXXX

? super XXXX



✓ `List<? super Dog> = new AL<Animal>();`

✗ `List<? super Dog> l = new AL<CostlyDog>();`



```
public interface java.util.Collection<E> extends java.lang.Iterable<E> {  
    public abstract int size();  
    public abstract boolean isEmpty();  
    public abstract boolean contains(java.lang.Object);  
    public abstract java.util.Iterator<E> iterator();  
    public abstract java.lang.Object[] toArray();  
    public abstract <T> T[] toArray(T[]);  
    public abstract boolean add(E);  
    public abstract boolean remove(java.lang.Object);  
    public abstract boolean containsAll(java.util.Collection<?>);  
    public abstract boolean addAll(java.util.Collection<? extends E>);  
    public abstract boolean removeAll(java.util.Collection<?>);  
    public boolean removeAll(java.util.function.Predicate<? super E>);  
    public abstract boolean retainAll(java.util.Collection<?>);  
    public abstract void clear();  
    public abstract boolean equals(java.lang.Object);  
    public abstract int hashCode();  
    public java.util.Spliterator<E> spliterator();  
    public java.util.stream.Stream<E> stream();  
    public java.util.stream.Stream<E> parallelStream();  
}
```