DanSet<Object>

- Array[Object]
- int size
- int currrentSize

+ add(Object element): boolean
+ addAll(Collection c) : boolean
+ clear() : void
+ contains(Object object) : boolean
+ isEmpty() : boolean
+ remove(Object object): boolean
+ size() : int
+ toArray() : Object[]

DanList<Object>

- Array[Object]
-int currentSize

+ add(Object element): boolean
+ addObject element, Integer index) : boolean
+ remove(Integer index): boolean
+ remove(Object object): boolean
+ get(Integer index): Object
+ set(Integer index, Object object): void
+ clear() : void
+ isEmpty() : boolean
+ contains() : boolean
+ size() : int

DanMap<Object, Object>

- Array[Object] (for keys)
- Array[Object] (for values)

+ clear() : void
+ containsKey(Object key) : boolean
+ containsValue(Object value) : boolean
+ equals(Object map) : boolean
+ get(Object key) : Object (value at key)
+ isEmpty() : boolean
+ keySet() : Object[] (of keys)
+ put(Object key , Object value) : void
+ putAll(Map<Object,Object>) : void
+ remove(Object key) : void
+ size(): int
+ values() : Object[] (of values)