1. 單列集合框架結構  
   *|----Collection接口：单列集合，用来存储一个一个的对象  
   \* |----List接口：存储有序的、可重复的数据。 -->“动态”数组  
   \* |----ArrayList、LinkedList、Vector  
   \*  
   \* |----Set接口：存储无序的、不可重复的数据 -->高中讲的“集合”  
   \* |----HashSet、LinkedHashSet、TreeSet*

對應圖示：



1. Collection接口常用方法：
2. *contains(Object obj):判断当前集合中是否包含obj*

*2. containsAll(Collection coll1):判断形参coll1中的所有元素是否都存在于当前集合中。*

*3. remove(Object obj):从当前集合中移除obj元素。*

*4. removeAll(Collection coll1):差集：从当前集合中移除coll1中所有的元素。*

*5.retainAll(Collection coll1):交集：获取当前集合和coll1集合的交集，并返回给当前集合  
如果內容物都相同,list是有序的,如果順序錯就是false. 如果是hashmap(沒有順序問題)就是true.*

*6. equals(Object obj):要想返回true，需要当前集合和形参集合的元素都相同。*

*7. hashCode():返回当前对象的哈希值*

*8. 集合 --->数组：toArray()*

*9. iterator():返回Iterator接口的实例，用于遍历集合元素。*

*10. add(Object e):将元素e添加到集合coll中*

*11. addAll(int index, Collection eles):从index位置开始将eles中的所有元素添加进来*

*12. clear():清空集合元素 13. isEmpty():判断当前集合是否为空*

*13. size():获取添加的元素的个数*

*14. isEmpty():判断当前集合是否为空*

1. Collection集合與數組間的轉換

*集合 --->数组：toArray()*Object[] arr = coll.toArray();  
for(int i = 0;i < arr.length;i++){  
 System.*out*.println(arr[i]);  
}  
  
*//拓展：数组 --->集合:调用Arrays类的静态方法asList(T…t):因為()內放的是對象,所以如果添加數組是基本數據類型,顯示的會是地址值,要顯示其元素,添加時類型就要用對應的包裝類才行.*List<String> list = Arrays.*asList*({"AA", "BB", "CC"});  
System.*out*.println(list);  
  
List arr1 = Arrays.*asList*(new int[]{123, 456});*//把整個數組當成是一個對象*System.*out*.println(arr1.size());*//1*System.*out*.println(arr1);*//[[I@75bd9247] (地址值)  
(開發中不用再寫 new xxx直接{123,456}*List arr2 = Arrays.*asList*(new Integer[]{123, 456});*//123,456:把他們各看成是一個對象*System.*out*.println(arr2.size());*//2*System.*out*.println(arr2);*//[123, 456]*

1. 使用Collection集合來存儲對象，要求對象所屬的類滿足：

*向Collection接口的实现类的对象中添加数据obj时，要求obj所在类要重写equals().*

1. 本章節對大家的要求：

*層次一：選擇合適的集合類去實現數據的保存，調用其內部的相關方法。*

*層次二：不同的集合類底層的數據結構為何？如何實現數據的操作的：增刪改查等。*