

Java 中的 ArrayList

ArrayList 是一个基于数组，支持自动扩容的数据结构（即这个类是对动态数组的一个封装），非常常用。

● 定义 ArrayList

定义 ArrayList 类型的变量必须指定元素的类型（这个类型不能是简单类型，必须是 Class）。

即： `ArrayList<int> iArray;` //~~错误~~ 错误，元素类型不能是简单类型

`ArrayList<String> sArray;` //正确

`ArrayList<Integer> iArray;` //正确， Integer 是一个 Class （是对 int 的包装类）

● 创建 ArrayList 对象

使用 new 来创建一个 ArrayList 对象（这个过程通过 new 给对象分配内存空间）

```
ArrayList<String> sArray;
sArray = new ArrayList<String>(); //注意格式， ArrayList<String> 是类型描述， 后面跟一对()
//括号里放类构造函数的参数列表
通常定义对象和创建对象会写成一行，显得比较简洁。
ArrayList<String> sArray = new ArrayList<String>();
```

● 常用方法

创建了 ArrayList 对象后，就可以对这个创建好的对象调用其方法来完成要做的功能。

◆ 增加一个元素

- `add(E)`, 将元素增加到集合末尾。
- `add(int index, E)`, 将元素增加到集合中指定的下标位置。下标从 0 开始

```
ArrayList<String> sArray = new ArrayList<String>();
String s1 = "hello";
String s2 = "world";
sArray.add(s1);
sArray.add(0, s2); //将 s2 插入到下标 0 的位置，即列表中的第一个元素
```

◆ 删除所有元素

- `clear()`, 清空列表。
- `sArray.clear();`

◆ 获取一个元素

- `get(int index)`, 获取指定下标的元素。
- `String s = sArray.get(1); //获取下标 1 的元素，即列表中第二个元素`

◆ 删除一个元素

- `remove(int index)`, 删除指定下标的元素。
- `remove(E)`, 删除列表中与 E 的值相等的第一个匹配的元素

```
ArrayList<String> sArray = new ArrayList<String>();
String s1 = "hello";
String s2 = "world";
sArray.add(s1);
sArray.add(0, s2);
String s3 = sArray.remove(0); //删除第 1 个元素，并将删除的元素的值返回，所以 s3="world" 第 1 个元素值。
boolean b = sArray.remove("hello"); //删除列表中值等于"hello"的第一个匹配的元素，成功返回 true，否则 false
```

◆ 判断列表是否为空

- `isEmpty()`, 如果为空返回 true，否则返回 false。
- `boolean noElem = sArray.isEmpty();`

◆ 获取列表中元素个数

- `size()`， 返回元素个数，空列表返回 0

```
int cnt= sArray.size();
```

◆ 判断列表中是否包含某个值的元素

- `contains(E)`，如果包含返回 true，否则 false

```
String s = "hello";
```

```
boolean b = sArray.contains(s);
```

◆ 遍历一个 ArrayList

通常用两种方式遍历一个 ArrayList

- **for 循环通过下标遍历所有元素，通过 get(index)得到元素值**

```
for (int i=0; i<sArray.size(); i++){  
    String s = sArray.get(i);  
    System.out.println(s);  
}
```

- **foreach**

```
for (String s: sArray) {  
    System.out.println(s);  
}
```