# 回顾

1. 异常Exception :运行时异常(空指针异常/数组越界异常/算术异常/类转换异常/数字格式转换异常)与检查性异常(类找不到异常)
2. 处理异常:①try-catch-finally ②throws ③throw ④自定义异常

# 集合框架

概念:特指java工具包(java.util)提供的对集合的一系列操作类

作用:1. 对静态数组的增强(集合长度可变. 支持链表,无序集以及映射集迭代器)

2.支持泛型(jdk1.5)

顶层接口:

List接口:有序集合

Collection -> Set接口:无序集合

Map接口:映射集合

总体说明:主要的是学习集合框架中常见的接口,接口的实现类,以及对应的方法和应用场景.

## List接口有序集合实现类

### ArrayList

先从最简单的一个类入手:ArrayList

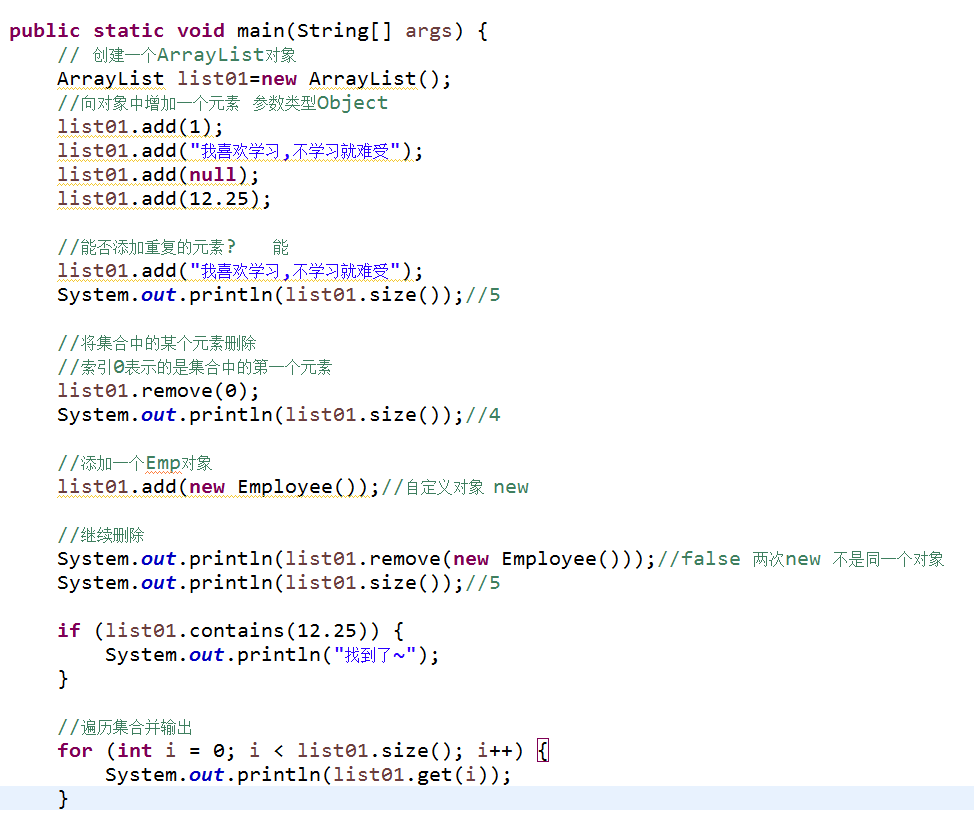


|  |  |
| --- | --- |
| **构造方法摘要** | |
| **[ArrayList](mk:@MSITStore:E:\\标准化教案\\帮助文档\\API帮助文档\\jdk1.6\\JDK_API_1_6.CHM::/java/util/../../java/util/ArrayList.html" \l "ArrayList())**()  构造一个初始容量为 10 的空列表。 |  |
| **[ArrayList](mk:@MSITStore:E:\\标准化教案\\帮助文档\\API帮助文档\\jdk1.6\\JDK_API_1_6.CHM::/java/util/../../java/util/ArrayList.html" \l "ArrayList(java.util.Collection))**([Collection](mk:@MSITStore:E:\\标准化教案\\帮助文档\\API帮助文档\\jdk1.6\\JDK_API_1_6.CHM::/java/util/../../java/util/Collection.html" \o "java.util 中的接口)<? extends [E](mk:@MSITStore:E:\\标准化教案\\帮助文档\\API帮助文档\\jdk1.6\\JDK_API_1_6.CHM::/java/util/../../java/util/ArrayList.html" \o "ArrayList 中的类型参数)> c)  构造一个包含指定 collection 的元素的列表，这些元素是按照该 collection 的迭代器返回它们的顺序排列的。 |  |
| **[ArrayList](mk:@MSITStore:E:\\标准化教案\\帮助文档\\API帮助文档\\jdk1.6\\JDK_API_1_6.CHM::/java/util/../../java/util/ArrayList.html" \l "ArrayList(int))**(int initialCapacity)  构造一个具有指定初始容量的空列表。 |  |

重要的方法:

|  |  |
| --- | --- |
| boolean | **[add](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/util/ArrayList.html" \l "add(E))**([E](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/util/ArrayList.html" \o "ArrayList 中的类型参数) e)  将指定的元素添加到此列表的尾部。 |
| void | **[add](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/util/ArrayList.html" \l "add(int, E))**(int index, [E](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/util/ArrayList.html" \o "ArrayList 中的类型参数) element)  将指定的元素插入此列表中的指定位置。 |
| void | **[clear](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/util/ArrayList.html" \l "clear())()  移除此列表中的所有元素。** |
| boolean | **[contains](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/util/ArrayList.html" \l "contains(java.lang.Object))([Object](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/lang/Object.html" \o "java.lang 中的类) o)  如果此列表中包含指定的元素，则返回 true。** |
| [E](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/util/ArrayList.html" \o "ArrayList 中的类型参数) | **[get](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/util/ArrayList.html" \l "get(int))(int index)  返回此列表中指定位置上的元素。** |
| [E](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/util/ArrayList.html" \o "ArrayList 中的类型参数) | **[remove](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/util/ArrayList.html" \l "remove(int))(int index)  移除此列表中指定位置上的元素。** |
| int | **[size](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/util/ArrayList.html" \l "size())()  返回此列表中的元素数。长度** |
| [Object](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/lang/Object.html" \o "java.lang 中的类)[] | **[toArray](mk:@MSITStore:E:\\标准化教案\\API帮助文档\\JDK_API_1_6.CHM::/java/util/ArrayList.html" \l "toArray())()  按适当顺序（从第一个到最后一个元素）返回包含此列表中所有元素的数组。** |

常见的方法:



我们发现,刚刚的代码有效的补充了我们之前案例中数组的不足:

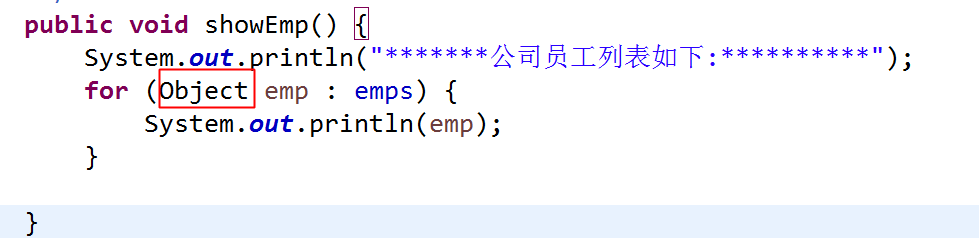
1. 长度可变
2. 数组定义,数组类型是固定,Arraylist中类型是Object,随意添加各种类型
3. ArrayList中提供了大量已经实现的方法例如添加add,remove等等,让我们更多的时间关注业务,而不是算法

一个问题:

add(任何类型)->当一个对象存放到集合中的时候,这个对象就自动被装箱(Object)

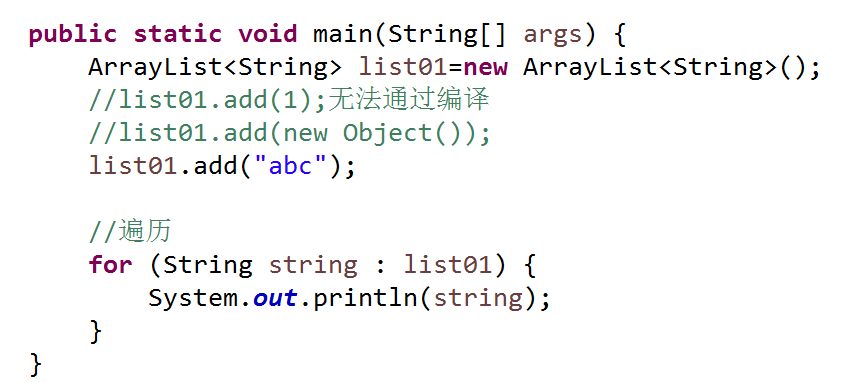
把刚才用数组做的员工管理改成用ArrayList



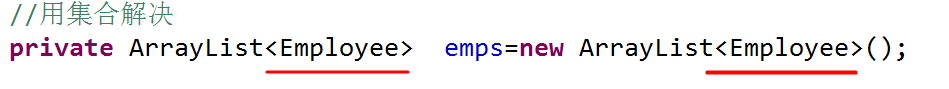


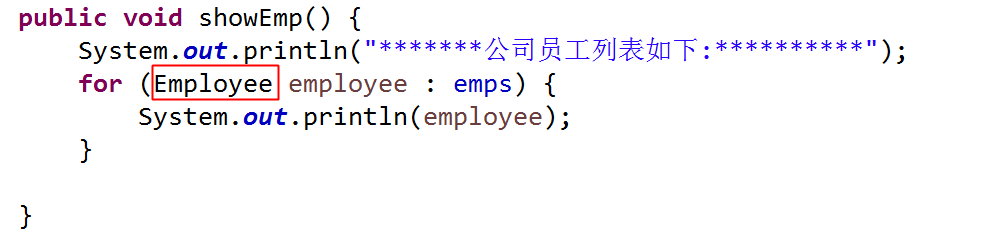
### 泛型

泛型 本质上认为是一种类型检查的机制,存放的元素到集合要进行类型的检查,取出来不需要强制转型了.

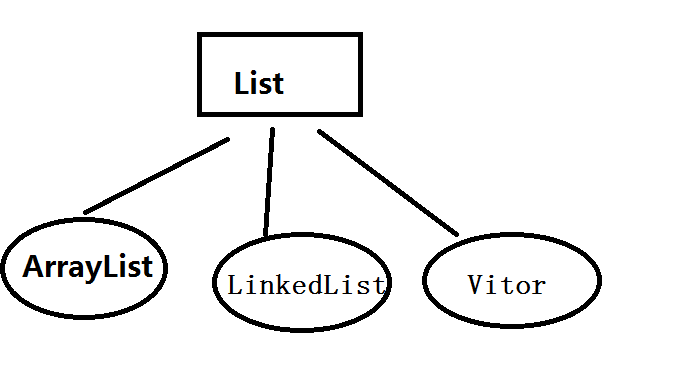


刚才的查询员工加上泛型

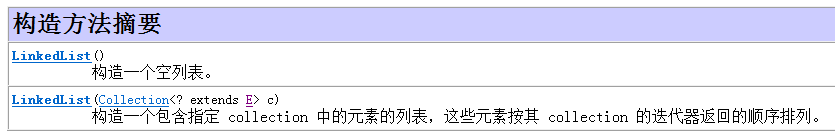




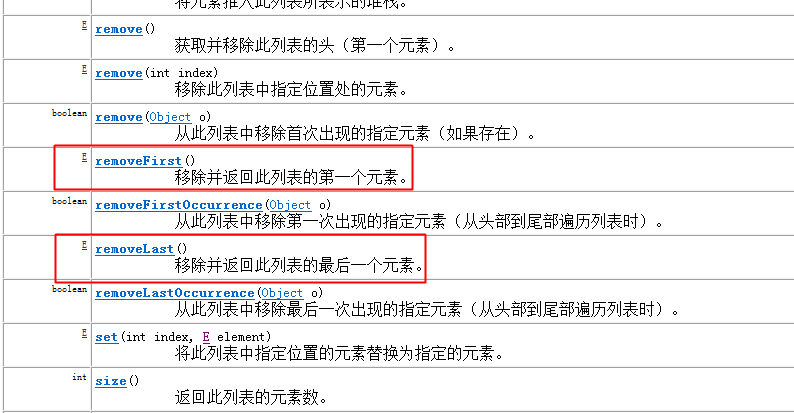
## LinkedList



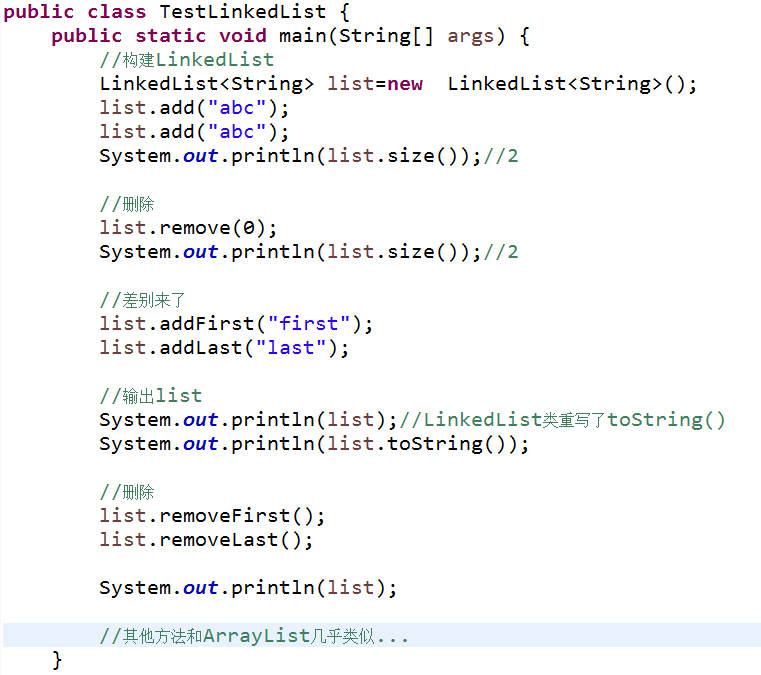




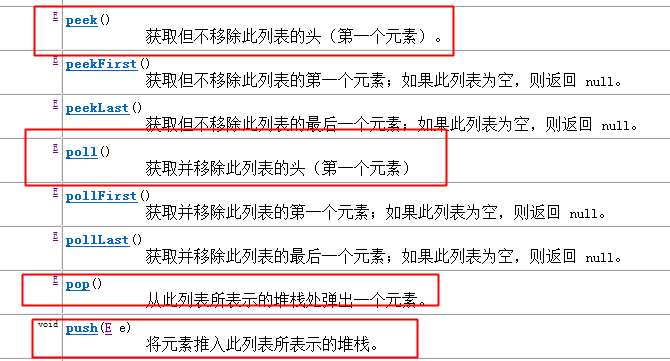


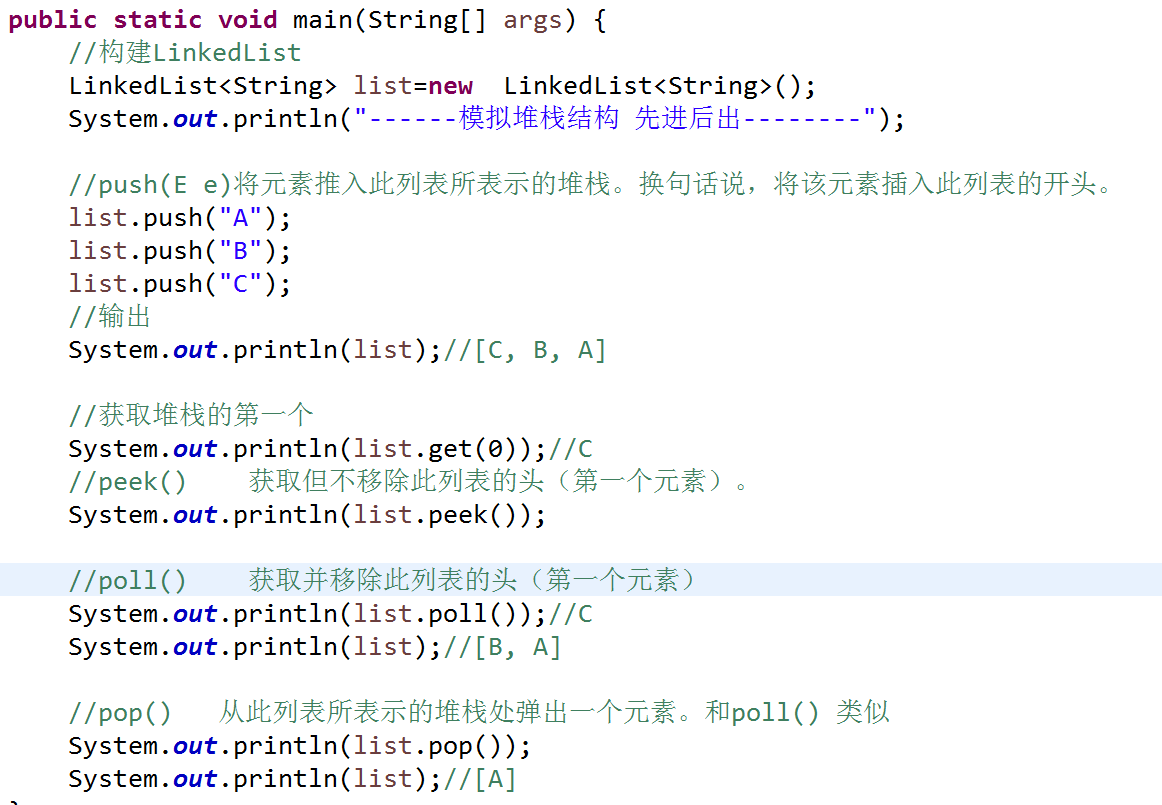


### LinkedList新增的方法



### 栈操作

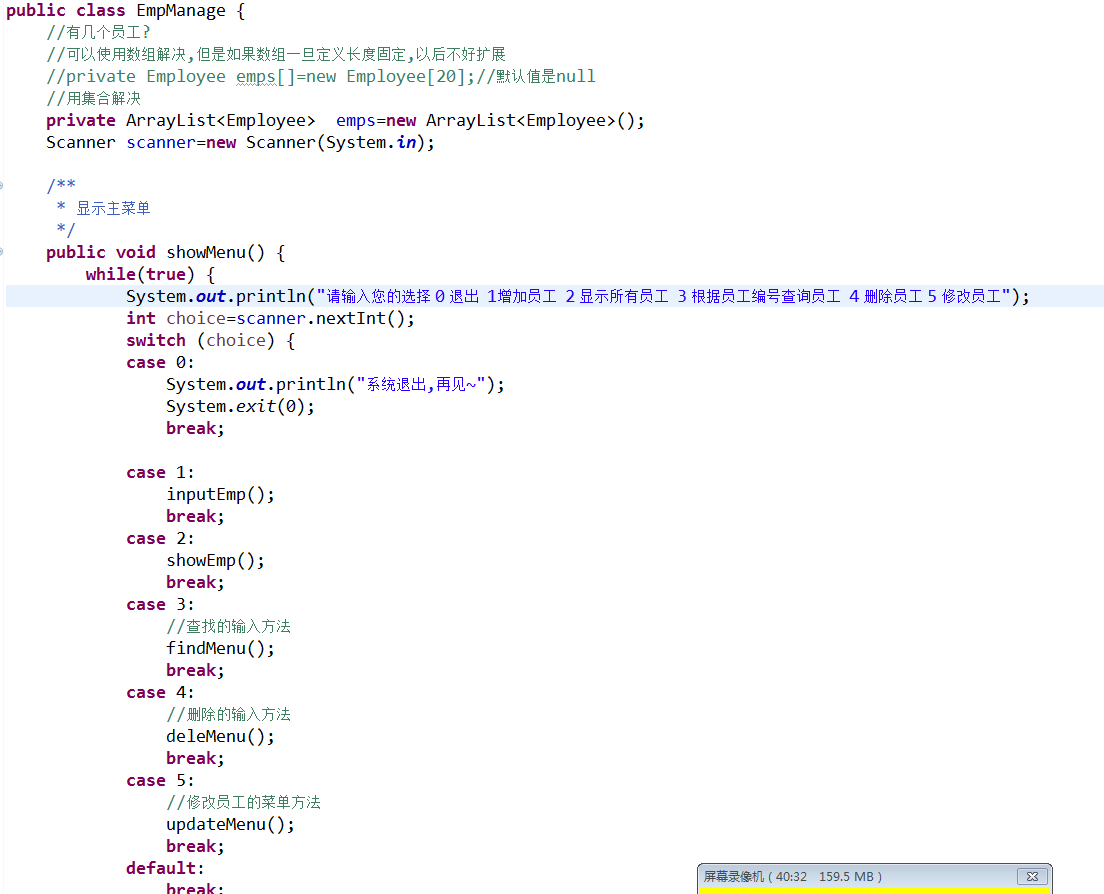




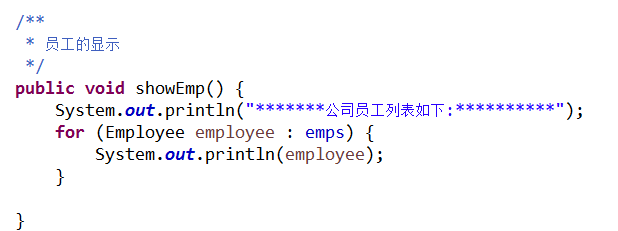
## 作业

使用ArrayList实现:查询所有员工,增加,查找单个员工,删除,修改和之前的菜单功能

0 退出 1增加员工 2 显示所有员工 3 根据员工编号查询员工 4 删除员工 5 修改员工





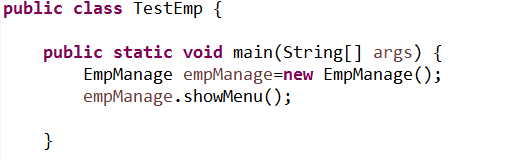












# 总结