**泛型知识点**

## 泛型的定义

**<T>或者<T extends class>用来声明类或方法是泛型的，以下是示例**

* 定义类Public class Test<T>{}
* 定义方法public <T> T test(){};
* 定义方法public <T> T test(T t){};
* 定义方法public <T> void test(T t){};
* 定义静态方法public static <T> void test(){};
* 定义可变长参数方法public static <T> T test(T… args){};

**在泛型类用作参数时，如果还是不确定具体类型，用通配符?代替**

* Public void test(Test<?> t){};
* Public void test(Test<? Extends String> t){};

## 类型擦除

在生成的Java字节代码中是不包含泛型中的类型信息的。使用泛型的时候加上的类型参数，会被编译器在编译的时候去掉。这个过程就称为类型擦除。 如在代码中定义的List<Object>和List<String>等类型，在编译之后都会变成List。

**类型擦除的过程：**首先是找到用来替换类型参数的具体类，把代码中的类型参数都替换成具体的类。同时去掉出现的类型声明，即去掉<>的内容。如T get()方法声明就变成了Object get()

## 泛型特性

* 因为类型擦除，不同参数泛型类编译后为同个类类型

