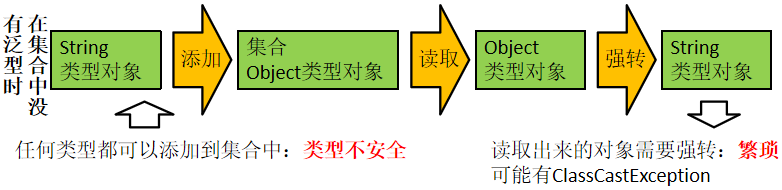
* **核心思想：**

**把一个集合中的内容限制为一个特定的数据类型，这就是generics背后的核心思想**。

* **为什么要有泛型（Generic）？**
* 解决元素存储的安全性问题。
* 解决获取数据元素时，需要类型强转的问题。





* 泛型，JDK1.5新加入的，解决数据类型的安全性问题，其**主要原理**是**在类声明时通过一个标识表示类中某个属性的类型或者是某个方法的返回值及参数类型**。这样在类声明或实例化时只要指定好需要的具体的类型即可。
* Java泛型可以保证如果程序在编译时没有发出警告，运行时就不会产生ClassCastException异常。同时，代码更加简洁、健壮。
* **使用泛型**
* **泛型的声明：**
  + interface List<T>和class TestGen<K,V>
  + 其中，T,K,V不代表值，而是表示类型。这里使用任意字母都可以。常用T表示，是Type的缩写。
* **泛型的实例化：**
  + 一定要在类名后面指定类型参数的值（类型）。如：

List<String> strList = new ArrayList<String>();

Iterator<Customer> iterator = customers.iterator();

* + **T只能是类，不能用基本数据类型填充**。
* **泛型的几个重要使用**
* 在集合中使用泛型
* 自定义泛型类
* 泛型方法
* 泛型接口