

编程入门基础

——包装类型

主讲教师：耿宇航



包装类

- 对应每个基本的类型，都有一个对象类型与之对应，为之服务
 - int Integer
 - double Double
 - char Character
 - ...

包装类的作用

包装类是基本类型与对象类型间的桥梁

把基本类型封装为对象，
或把对象转为基本类型

封装为对象

- Vector a = new Vector();
- a.add(...);

```
Vector a = new Vector();  
a.add(new Integer(10));  
a.add(new Double(3.14));
```

此处需要的是一个对象，而**int, double**不是对象，不能加入

把普通类型“包”起来，就可以当作对象用了。

还原为普通类型

- `Double d = new Double(3.14);`
- `double x = d.doubleValue();`
- 整数相应为：`intValue()`

Double类的其它方法

- `parseDouble(String s)`
 - 把串转化为浮点数
- `MAX_VALUE`
 - 属性。最大能表示的数字
- `toHexString()`
 - 在内存中的表示形式

Integer的其它方法

- parseInt(String s)
 - 把串转化为整数
- toBinaryString(int i)
 - 转化为二进制的格式
- toHexString(int i)
 - 转化为十六进制的形式

自动装箱和拆箱

- JDK1.5以后，提供了一个很有趣的特性：自动进行“装箱”和“拆箱”
- 就是在 int --- Integer 之间，double ---- Double 之间等，进行相互转换的时候，可以不显式地写出代码。java会自动替我们完成转换。

装箱

- `double d = 3.14;`
- `Double dd;`
- `dd = d;` // ??? jdk1.4中会出错

拆箱

- `Integer ii = new Integer(15);`
- `int n = ii;`
- 在低版本的jdk中会出错

谢谢！