

第十一章：常用类的概述和使用

11.1 常用的包（熟悉）

11.1.1 包的名称和功能

- java.lang包 - 该包是Java语言的核心包，并且该包中的所有内容由Java虚拟机自动导入。
如：System类、String类、...
- java.util包 - 该包是Java语言的工具包，里面提供了大量工具类以及集合类等。
如：Scanner类、Random类、List集合、...
- java.io包 - 该包是Java语言中的输入输出包，里面提供了大量读写文件相关的类等。
如：FileInputStream类、FileOutputStream类、...
- java.net包 - 该包是Java语言中的网络包，里面提供了大量网络编程相关的类等。
如：ServerSocket类、Socket类、...
- java.sql包 - 该包是Java语言中的数据包，里面提供了大量操作数据库的类和接口等。
如：DriverManager类、Connection接口、...
-
- Java程序员在编程时可以使用大量类库，因此Java编程时需要记的很多，对编程能力本身要求不是特别的高。

11.2 Object类的概述（重点）

11.2.1 基本概念

- java.lang.Object类是Java语言中类层次结构的根类，也就是说任何一个类都是该类的直接或者间接子类。
- 如果定义一个Java类时没有使用extends关键字声明其父类，则其父类为 java.lang.Object 类。
- Object类定义了“对象”的基本行为，被子类默认继承。

11.2.2 常用的方法

方法声明	功能介绍
Object()	使用无参方式构造对象
boolean equals(Object obj)	用于判断调用对象是否与参数对象相等。 该方法默认比较两个对象的地址是否相等，与 == 运算符的结果一致 若希望比较两个对象的内容，则需要重写该方法。 若该方法被重写后，则应该重写hashCode方法来保证结果的一致性。
int hashCode()	用于获取调用对象的哈希码值(内存地址的编号)。 若两个对象调用equals方法相等，则各自调用该方法的结果必须相同 若两个调用对象equals方法不相等，则各自调用该方法的结果应该不相同。 为了使得该方法与equals方法保持一致，需要重写该方法。
String toString()	用于获取调用对象的字符串形式 该方法默认返回的字符串为：包名.类名@哈希码值的十六进制 为了返回更有意义的数据，需要重写该方法 使用print或println打印引用或字符串拼接引用都会自动调用该方法
Class<?> getClass()	用于返回调用对象执行时的Class实例，反射机制使用

案例题目：

编程实现Student类的封装，特征：学号(id)和姓名，要求提供打印所有特征的方法。

编程实现StudentTest类，在main方法中使用有参方式构造两个Student类型的对象并打印特征。

题目扩展:

如何实现以姓名作为基准判断两个对象是否相等？以及以学号和姓名同时作为基准判断两个对象是否相等？

11.3 包装类（熟悉）

11.3.1 包装类的概念

通常情况下基本数据类型的变量不是对象，为了满足万物皆对象的理念就需要对基本数据类型的变量进行打包封装处理变成对象，而负责将这些变量声明为成员变量进行对象化处理的相关类，叫做包装类。

如：

```
Person p = new Person();
```

```
int num = 10;
```

11.3.2 包装类的分类

包装类	对应的基本类型
java.lang.Byte	byte
java.lang.Short	short
java.lang.Integer	int
java.lang.Long	long
java.lang.Float	float
java.lang.Double	double
java.lang.Boolean	boolean
java.lang.Character	char

11.3.3 Integer类的概述

(1) 基本概念

java.lang.Integer类内部包装了一个int类型的变量作为成员变量，主要用于实现对int类型的包装并提供int类型到String类之间的转换等方法。

(2) 常用的常量

常量类型和名称	功能介绍
public static final int MAX_VALUE	表示int类型可以描述的最大值，即 $2^{31}-1$
public static final int MIN_VALUE	表示int类型可以描述的最小值，即 -2^{31}
public static final int SIZE	表示int类型采用二进制补码形式的位数
public static final int BYTES	表示int类型所占的字节个数
public static final Class TYPE	表示int类型的Class实例

(2) 常用的方法

方法声明	功能介绍
Integer(int value)	根据参数指定的整数来构造对象（已过时）
Integer(String s)	根据参数指定的字符串来构造对象（已过时）
int intValue()	获取调用对象中的整数值并返回
static Integer valueOf(int i)	根据参数指定整数值得到Integer类型对象
boolean equals(Object obj)	比较调用对象与参数指定的对象是否相等
String toString()	返回描述调用对象数值的字符串形式
static int parseInt(String s)	将字符串类型转换为int类型并返回
static String toString(int i)	获取参数指定整数的十进制字符串形式
static String toBinaryString(int i)	获取参数指定整数的二进制字符串形式
static String toHexString(int i)	获取参数指定整数的十六进制字符串形式
static String toOctalString(int i)	获取参数指定整数的八进制字符串形式

（3）装箱和拆箱的概念

在Java5发布之前使用包装类对象进行运算时，需要较为繁琐的“拆箱”和“装箱”操作；即运算前先将包装类对象拆分为基本类型数据，运算后再将结果封装成包装类对象。

从Java5开始增加了自动拆箱和自动装箱的功能。

（4）自动装箱池

在Integer类的内部提供了自动装箱池技术，将-128到127之间的整数已经装箱完毕，当程序中使用该范围之间的整数时，无需装箱直接取用自动装箱池中的对象即可，从而提高效率。

11.3.4 Double类的概述

（1）基本概念

java.lang.Double类型内部包装了一个double类型的变量作为成员变量，主要用于实现对double类型的包装并提供double类型到String类之间的转换等方法。

（2）常用的常量

常量类型和名称	功能介绍
public static final int SIZE	表示double类型的二进制位数
public static final int BYTES	表示double类型的字节个数
public static final Class TYPE	表示double类型的Class实例

（3）常用的方法

方法声明	功能介绍
Double(double value)	根据参数指定的浮点数据来构造对象（已过时）
Double(String s)	根据参数指定的字符串来构造对象（已过时）
double doubleValue()	获取调用对象中的浮点数据并返回
static Double valueOf(double d)	根据参数指定浮点数据得到Double类型对象
boolean equals(Object obj)	比较调用对象与参数指定的对象是否相等
String toString()	返回描述调用对象数值的字符串形式
static double parseDouble(String s)	将字符串类型转换为double类型并返回
boolean isNaN()	判断调用对象的数值是否为非数字

- 扩展：

java.lang.Number类是个抽象类，是上述类的父类来描述所有类共有的成员。

11.3.5 Boolean类的概述

（1）基本概念

java.lang.Boolean类型内部包装了一个boolean类型的变量作为成员变量，主要用于实现对boolean类型的包装并提供boolean类型到String类之间的转换等方法。

（2）常用的常量

常量类型和名称	功能介绍
public static final Boolean FALSE	对应基值为false的对象
public static final Boolean TRUE	对应基值为true的对象
public static final Class TYPE	表示boolean类型的Class实例

（3）常用的方法

方法声明	功能介绍
Boolean(boolean value)	根据参数指定的布尔数值来构造对象（已过时）
Boolean(String s)	根据参数指定的字符串来构造对象（已过时）
boolean booleanValue()	获取调用对象中的布尔数值并返回
static Boolean valueOf(boolean b)	根据参数指定布尔数值得到Boolean类型对象
boolean equals(Object obj)	比较调用对象与参数指定的对象是否相等
String toString()	返回描述调用对象数值的字符串形式
static boolean parseBoolean(String s)	将字符串类型转换为boolean类型并返回

11.3.6 Character类的概述

(1) 基本概念

java.lang.Character类型内部包装了一个char类型的变量作为成员变量，主要用于实现对char类型的包装并提供字符类别的判断和转换等方法。

(2) 常用的常量

常量类型和名称	功能介绍
public static final int SIZE	表示char类型的二进制位数
public static final int BYTES	表示char类型的字节个数
public static final Class TYPE	表示char类型的Class实例

(3) 常用的方法

方法声明	功能介绍
Character(char value)	根据参数指定的字符数据来构造对象（已过时）
char charValue()	获取调用对象中的字符数据并返回
static Character valueOf(char c)	根据参数指定字符数据得到Character类型对象
boolean equals(Object obj)	比较调用对象与参数指定的对象是否相等
String toString()	返回描述调用对象数值的字符串形式
static boolean isUpperCase(char ch)	判断参数指定字符是否为大写字符
static boolean isLowerCase(char ch)	判断参数指定字符是否为小写字符
static boolean isDigit(char ch)	判断参数指定字符是否为数字字符
static char toUpperCase(char ch)	将参数指定的字符转换为大写字符
static char toLowerCase(char ch)	将参数指定的字符转换为小写字符

11.3.7 包装类 (Wrapper) 的使用总结

- 基本数据类型转换为对应包装类的方式
调用包装类的构造方法或静态方法即可
- 获取包装类对象中基本数据类型变量数值的方式
调用包装类中的xxxValue方法即可
- 字符串转换为基本数据类型的方式
调用包装类中的parseXxx方法即可

11.4 数学处理类 (熟悉)

11.4.1 Math类的概述

(1) 基本概念

java.lang.Math类主要用于提供执行数学运算的方法，如：对数，平方根。

(2) 常用的方法

方法声明	功能介绍
static int max(int a, int b)	返回两个参数中的最大值
static int min(int a, int b)	返回两个参数中的最小值
static double pow(double a, double b)	返回第一个参数的幂
static int abs(int a)	返回参数指定数值的绝对值
static long round(double a)	返回参数四舍五入的结果
static double sqrt(double a)	返回参数的平方根
static double random()	返回0.0到1.0的随机数

11.4.2 BigDecimal类的概述

(1) 基本概念

由于float类型和double类型在运算时可能会有误差，若希望实现精确运算则借助java.math.BigDecimal类型加以描述。

(2) 常用的方法

方法声明	功能介绍
BigDecimal(String val)	根据参数指定的字符串来构造对象
BigDecimal add(BigDecimal augend)	用于实现加法运算
BigDecimal subtract(BigDecimal subtrahend)	用于实现减法运算
BigDecimal multiply(BigDecimal multiplicand)	用于实现乘法运算
BigDecimal divide(BigDecimal divisor)	用于实现除法运算

11.4.3 BigInteger类的概念

(1) 基本概念

若希望表示比long类型范围还大的整数数据，则需要借助java.math.BigInteger类型描述。

(2) 常用的方法

方法声明	功能介绍
BigInteger(String val)	根据参数指定的字符串来构造对象
BigInteger add(BigInteger val)	用于实现加法运算
BigInteger subtract(BigInteger val)	用于实现减法运算
BigInteger multiply(BigInteger val)	用于实现乘法运算
BigInteger divide(BigInteger val)	用于实现除法运算
BigInteger remainder(BigInteger val)	用于实现取余运算
BigInteger[] divideAndRemainder(BigInteger val)	用于实现取商和余数的运算