# 第六章 常用类

## 6.1 System类

1. System.getProperty(String key)：获取系统某些信息

java.version -- Java运行时环境版本

java.vendor – Java运行时环境供应商

java.vendor.url – Java供应商的URL

java.home –Java安装目录

java.class.path – Java类路径

os.name – 操作系统的名称

os.version –操作系统的版本

os.arch – 操作系统的架构

file.separator – 文件分隔符（在UNIX系统中是“/”）

path.separator – 路径分隔符（在UNIX系统中是“：”）

line.separator – 行分隔符(在UNIX系统中是“/n”)

user.name – 用户的账户名称

user.home -- 用户的主目录

user.dir -- 用户当前的工作目录

1. System.currentTimeMillis()：返回当前的计算机时间，时间的表达格式为当前计算机时间和GMT时间(格林威治时间)1970年1月1号0时0分0秒所差的毫秒数。

## 6.2 Character类

1. Character.isLetter()：方法用于判断指定字符是否为字母
2. Character.isDigit() 方法用于判断指定字符是否为数字。
3. Character.isWhitespace() 方法用于判断指定字符是否为空白字符，空白符包含：空格、tab键、换行符。
4. Character.isUpperCase() 方法用于判断指定字符是否为大写字母。
5. Character.isLowerCase() 方法用于判断指定字符是否为小写字母。
6. Character.toUpperCase() 方法用于将小写字符转换为大写。
7. Character.toLowerCase() 方法用于将大写字符转换为小写。
8. toString() 方法用于返回一个表示指定 char 值的 String 对象。结果是长度为 1 的字符串，仅由指定的 char 组成。
9. Character.isTitleCase()用于判断指定字符是否为标题首字符。
10. Character.getNumericValue()用于获取指定字符的unicode的值。

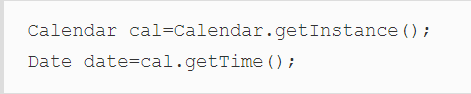
## 6.3 Random类

* 1. Random.nextInt()方法，是生成一个随机的int值，该值介于[0,n)的区间，也就是0到n之间的随机int值，包含0而不包含n。
  2. Random是伪随机，有时候随机数会相同，如果想让生成的随机数更加无法预测，可以用Random.setseed()设置随机数种子，使随机数更加随机。

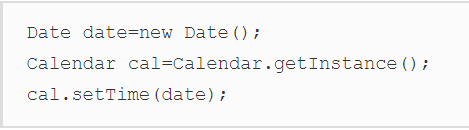
## 6.4 Date类

1. String dateStr = "";
2. Date date = **new** Date();
3. //format的格式可以任意
4. DateFormat sdf = **new** SimpleDateFormat("yyyy/MM/dd HH:mm:ss");
5. DateFormat sdf2 = **new** SimpleDateFormat("yyyy-MM-dd HH/mm/ss");
6. **try** {
7. dateStr = sdf.format(date);
8. System.out.println(dateStr);
9. dateStr = sdf2.format(date);
10. System.out.println(dateStr);
11. } **catch** (Exception e) {
12. e.printStackTrace();
13. }

Calerdar转换为Date



Date转化为Calendar



## 6.5 Integer类

* 1. Integer(实例).compareTo()：用于在数值上比较两个Integer对象，2个必须都是Integer对象才能进行对比。
  2. Integer.equals() ：Java中判断Integer是否相等可以用equals()或者“==”,“==”是进行地址及值比较，equals方法是数值比较，当Integer的值不在-128到127的时候，会新new一个对象，因此这个时候如果用“==”进行判断就会报错。
  3. Integer(实例).doubleValue()/floatValue()/intValue()/longValue/toString()：Integer类可以很方便的转换成double，float，int，long，string等类型，都有固定的方法进行转换。
  4. Integer.parseInt()一般用于将字符串转成基本数据类型int类型，在进行转换的时候parseInt每次都会构造一个常量值。
  5. Integer.toBinaryString()用于将十进制转成二进制输出。
  6. Integer.toHexString()用于将十进制转成十六进制输出。
  7. Integer.toHexString()用于将十进制转成八进制输出。
  8. Integer.valueOf()用于返回一个Integer（整型）对象，当被处理的字符串在-128和127（包含边界）之间时，返回的对象是预先缓存的。

## 6.6 Calendar类

## 6.7 Number的子类：Byte、Short、Integer、Long、Float、Double

## 6.8 Math类

## 6.9 String类

String.format()：方法用于创建格式化的字符串以及连接多个字符串对象

