

常用类

System

- System.in
- System.out
- System.getenv()
- System.getProperties()
- System.currentTimeMillis()
- System.gc()

```
package javaadvanceday03;

import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.util.Map;
import java.util.Map.Entry;
import java.util.Properties;
import java.util.Scanner;
import java.util.Set;

public class SystemDemo {

    public static void main(String[] args) throws FileNotFoundException,
    IOException {
        long currentTimeMillis = System.currentTimeMillis();
        Map<String, String> map = System.getenv();    //获取系统环境变量
        Set<Entry<String, String>> entrySet = map.entrySet();
        entrySet.forEach(System.out::println);

        System.out.println("=====");

        Properties properties = System.getProperties();    //Java运行环境的属性
        properties.store(new FileOutputStream("d:\\sys.properties"), "");

        System.out.println("=====");
        System.gc();    //触发垃圾回收

        System.out.println("计算程序运行的时间: " + (System.currentTimeMillis()-
        currentTimeMillis));    //计算程序运行所花的时间
    }
}
```

Runtime

```

Runtime runtime = Runtime.getRuntime();
System.out.println(runtime.availableProcessors()); //返回处理器（CPU）的数量
（逻辑处理器，比如双核返回2）
System.out.println(runtime.freeMemory()); //返回虚拟机的可用内存
System.out.println(runtime.maxMemory()); //最大管理内存
System.out.println(runtime.totalMemory()); //虚拟机总内存

runtime.gc(); //触发垃圾回收

runtime.exec("notepad.exe"); //执行windows命令

```

String

```

package javaadvanceday03;

public class StringDemo {

    public static void main(String[] args) {
        String s1 = "This is a test.";
        String s2 = "这是一个测试";

        char c = s1.charAt(0); //返回指定下标的字符（下标从0开始）
        System.out.println(c);
        System.out.println(s1.charAt(s1.length()-1));

        System.out.println(s1.concat(s2)); //连个字符串连接在一起，返回一个新字符串
        System.out.println(s1);

        System.out.println(s1.contains("is")); //是否包含指定字符串，返回布尔型

        System.out.println(s1.endsWith("test.")); //是否以指定字符串结尾
        System.out.println(s1.startsWith("This")); //是否以指定的字符串开头

        System.out.println(s1.indexOf("is")); //返回指定字符串在原字符串中起始位置（从0
        开始），如果没找到返回-1
        System.out.println(s1.lastIndexOf("is")); //返回指定字符串在原字符串中最后一次出
        现的位置，如果找不到返回-1

        System.out.println(s1.substring(5)); //截取并返回从指定位置开始到结束字符串
        System.out.println(s1.substring(5, 9)); //截取并返回从指定开始位置到指定结束位
        置（不包含）之间的字符串

        System.out.println(s1.toLowerCase()); //转小写
        System.out.println(s1.toUpperCase()); //转大写

        System.out.println(s1.replace("is", "are")); //用新的字符串替换所有的旧的字符
        串

        String strim = "      blank      ";
        System.out.println("-" + strim.trim() + "-"); //去掉字符串前后的空格

        String[] split = s1.split(" "); //按指定字符拆分字符串为数组，如果找不到指定字
        符，数组只有一个元素（原字符串）；如果用""作为指定字符拆分，按字符拆分成数组
        for(String s : split) {
            System.out.println(s);
        }
    }
}

```

```

    }

    byte[] bytes = s1.getBytes(); //转byte数组
    for(byte b : bytes) {
        System.out.println(b);
    }

    System.out.println("=====打印xxx在如下字符串中出现的所有位置:方法一=====");
    //打印xxx在如下字符串中出现的所有位置
    String s3 = "fdjsxxxaofpxxx0oti4k4;lxxx47392lxxx74093785xxx84309中文";
    int index = 0; //每次找到的新位置
    int offset = 0; //下一次开始查找的位置
    while((index = s3.indexOf("xxx", offset))>=0) {
        System.out.println(index);
        offset = index + 3; //xxx长度为3, 所以offset就是index+3
    }

    System.out.println("=====打印xxx在如下字符串中出现的所有位置:方法二=====");
    int index2 = -1;
    String stemp = s3;
    while((index2 = stemp.lastIndexOf("xxx"))>=0) {
        System.out.println(index2);
        stemp = s3.substring(0, index2);
    }

    System.out.println("=====把下列字符串中的xxx替换为***, 不用replace=====");
    String s4 = "fdjsxxxaofpxxx0oti4k4;lxxx47392lxxx74093785xxx84309中文";
    String[] split2 = s4.split("xxx");
    String snw2 = "";
    for(String s : split2) {
        snw2 = snw2 + s + "***";
    }
    System.out.println(snw2.substring(0, snw2.length()-3));
}

}

```

StringBuffer, StringBuilder

```

package javaadvanceday03;

public class StringBufferStringBuilder {

    public static void main(String[] args) {
        System.out.println("====用字符串+拼接字符串====");
        long currentTimeMillis = System.currentTimeMillis();
        String str = "";
        for(int i=0; i<1000; i++) {
            str += i;
        }
        System.out.println(System.currentTimeMillis()-currentTimeMillis);

        System.out.println("====用StringBuffer拼接字符串,线程安全的====");
        currentTimeMillis = System.currentTimeMillis();
        StringBuffer stringBuffer = new StringBuffer();
    }
}

```

```

        for(int i=0; i<100000; i++) {
            stringBuffer.append(i);
        }
        System.out.println(stringBuffer.toString().length()); //toString返回字符串
        System.out.println(System.currentTimeMillis()-currentTimeMillis);

        System.out.println("====用StringBuilder拼接字符串，线程不安全的====");
        StringBuilder stringBuilder = new StringBuilder();
        stringBuilder.append("Hello ");
        stringBuilder.append(" HQYJ.");
        stringBuilder.insert(5, ","); //在指定位置插入字符串
        System.out.println(stringBuilder.toString());

    }

}

```

BigDecimal

```

        BigDecimal decimal = new BigDecimal(0.1); //不要用double作为参数初始化
        BigDecimal实例
        System.out.println(decimal);

        //推荐使用下面两种方式
        BigDecimal decimal2 = new BigDecimal("0.8");
        System.out.println(decimal2);
        BigDecimal decimal3 = BigDecimal.valueOf(0.7);
        System.out.println(decimal3);

        //加法
        System.out.println(decimal2.add(decimal3));
        //减法
        System.out.println(decimal2.subtract(decimal3));
        //乘法
        System.out.println(decimal2.multiply(decimal3));
        //除法
        //除法要指定小数位数（精度），舍弃小数的方式（BigDecimal.ROUND_HALF_UP，四舍五入）
        System.out.println(decimal2.divide(decimal3, 2,
        BigDecimal.ROUND_HALF_UP));

        //保留2位数小数，四舍五入
        BigDecimal decimal4 = BigDecimal.valueOf(0.785);
        System.out.println(decimal4.setScale(2, BigDecimal.ROUND_HALF_UP));
    }
}

```

Calendar

```

        Calendar calendar = Calendar.getInstance();
        System.out.println(calendar.getTime()); //返回日历对象的日期时间

        calendar.add(Calendar.DAY_OF_MONTH, 2); //按指定的方式（年，月，日，时，分，
        秒。。。）增减（数值为负数）日期时间
        System.out.println(calendar.getTime());
    }
}

```

```

System.out.println(calendar.get(Calendar.YEAR)); //获取年份
System.out.println(calendar.get(Calendar.MONTH)); //获取月份（0~11）
System.out.println(calendar.get(Calendar.DAY_OF_MONTH)); //获取日
System.out.println(calendar.get(Calendar.HOUR_OF_DAY)); //获取小时
System.out.println(calendar.get(Calendar.MINUTE)); //获取分钟
System.out.println(calendar.get(Calendar.SECOND)); //获取秒
System.out.println(calendar.get(Calendar.DAY_OF_WEEK)); //获取星期几（从周日（1）开始，周六是7）
System.out.println(calendar.get(Calendar.WEEK_OF_YEAR)); //获取当年的第几周
System.out.println(calendar.getMaximum(Calendar.DAY_OF_MONTH)); //当月的天数

calendar.set(Calendar.DAY_OF_MONTH, 1); //设置日期中的某段（年，月，日。。。）的值
System.out.println(calendar.getTime());
System.out.println(calendar.getActualMaximum(Calendar.WEEK_OF_MONTH));
//获取一个月有几周

```

今日作业

1. 打印HTML页面

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <h1>Hello HQYJ.</h1>
  </body>
</html>

```

2. 打印SQL语句. 姓名，年龄，学号从控制台输入

```

update student set
  stu_name='Tom', stu_age=22
where stu_num='111'

```

3. 用BigDecimal实现收银程序。从控制台输入商品数量，输入商品单价（小数），计算总金额（保留两位小数）并打印
4. 从控制台输入年份，月份。打印当月的日历（星期，公历日期，不是当月的日期留白，如下图：红色框内容不需要）

一	二	三	四	五	六	日
30 十六	1 十七	2 十八	3 十九	4 二十	5 廿一	6 廿二
7 大雪	8 廿四	9 廿五	10 廿六	11 廿七	12 廿八	13 廿九
14 三十	15 十一月	16 初二	17 初三	18 初四	19 初五	20 初六
21 冬至	22 初八	23 初九	24 初十	25 初十	26 十二	27 十三
28 十四	29 十五	30 十六	31 十七	1 十八	2 十九	3 二十