

java日期处理

1、应用场景

Java中日期经常使用以下五个方面：

- A.创建日期
- B.日期格式化显示
- C.日期的转换（主要是和字符串之间的相互转换）
- D.日期中年.月.日.时.分.秒.星期.月份等获取。
- E.日期的大小比较.日期的加减。

2、日期处理的相关API

在java中提供了日期处理的相关API，具体的有：

- 1、 java.util.Date
- 2、 java.text.SimpleDateFormat
- 3、 java.util.Calendar

3、 java.util.Date

java.util.Date 表示特定的瞬间，精确到毫秒。提供了很多的方法，但是很多已经过时，不推荐使用。

```
//下面的代码得到的一个数字。它表示的：当前时间-1970.1.1 中所差的毫秒数。其返回值是一个long类型
System.currentTimeMillis()

//直接new Date, 获取当前时间
Date date = new Date();
System.out.println(date);
//根据当前时间的毫秒数, 转换成Date类型
System.out.println(System.currentTimeMillis());
//可以随意指定一个long类型的数字, 转换成Date类型
System.out.println(new Date(1231));

Date date1 = new Date(15509466176801);
System.out.println(date1);

//判断时间的前后
```

```
System.out.println(date.after(date1));
System.out.println(date.before(date1));

//compareTo方法, 返回-1表示小于给定时间, 0-相等, 1-大于给定时间
System.out.println(date.compareTo(date1));
```

4、java.text.SimpleDateFormat

SimpleDateFormat 是一个用来格式化和分析日期的具体类。

它允许进行格式化（日期 -> 字符串）、解析（字符串 -> 日期）和规范化。

```
//当创建SimpleDateFormat对象的时候, 如果不指定格式, 则采用默认格式
SimpleDateFormat format = new SimpleDateFormat();
//format()方法, 将日期转成指定格式的字符串
String dateStr = format.format(date);
//默认的格式: 19-4-11 上午9:58
System.out.println(dateStr);
```

上述的格式还不是我们想要的, 如果想要转换成指定格式, 如下:

```
//需要注意的是: 大小写。 y是小写, M要大写表示月份, d表示日。h小时, m小写表示分, s秒
SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd hh:mm:ss");
String dateStr = format.format(date);
System.out.println(dateStr);
```

格式大全

/**

SimpleDateFormat函数语法:

G 年代标志符

y 年

M 月

d 日

h 时 在上午或下午 (1~12)

H 时 在一天中 (0~23)

m 分

s 秒

S 毫秒

E 星期

D 一年中的第几天

F 一月中第几个星期几

w 一年中第几个星期

W 一月中第几个星期

a 上午 / 下午 标记符

k 时 在一天中 (1~24)

K 时 在上午或下午 (0~11)

z 时区

*/

```
SimpleDateFormat myFmt4 = new SimpleDateFormat("一年中的第 D 天 一年中第w个星期 一月中第W个星期 在一天中k时 z时区");
```

```

        String format2 = myFmt4.format(date);
        System.out.println(format2);

//将字符串转成Date
String dateStr1 = "2018-04-20 20:08:45";
try {
    Date parse = format.parse(dateStr1);
    System.out.println(parse);
} catch (ParseException e) {
    e.printStackTrace();
}

```

5、java.util.Calendar

```

package com.psfed.util.date;

import java.util.Calendar;
import java.util.Date;

public class CalendarDemo {
    public static void main(String[] args) {
        Calendar instance = Calendar.getInstance();
        Date time = instance.getTime();
        System.out.println(time);
        System.out.println(instance.get(Calendar.YEAR));
        System.out.println(instance.get(Calendar.MONTH));
        System.out.println(instance.get(Calendar.DAY_OF_MONTH));
        System.out.println(instance.get(Calendar.DAY_OF_WEEK));
        System.out.println(instance.get(Calendar.DAY_OF_YEAR));
        System.out.println(instance.get(Calendar.DAY_OF_WEEK_IN_MONTH));
        System.out.println(instance.get(Calendar.HOUR));
        System.out.println(instance.get(Calendar.MINUTE));
        System.out.println(instance.get(Calendar.SECOND));
        System.out.println(instance.get(Calendar.MILLISECOND));

    }

}

```

使用Calendar类来进行日期计算

```

private static Date extracted(Date d, int day) {
    Calendar now = Calendar.getInstance();
    now.setTime(d);
    System.out.println(now.getTime());
    //年份+1
    now.add(Calendar.YEAR, 1);
    System.out.println(now.getTime());
    //月份+1

```

```
now.add(Calendar.MONTH, 2);  
return now.getTime();  
}
```

作业

- 1、计算从2011-10-25 19:30:45算起，1500天之后是几月几号，并格式化xxxx年xx月x日的形式打印出来。
- 2、编写一个日期计算程序。

设计要求:

- 1)、用户输入一个月份，屏幕上输出这个月的月历，每星期一行，从星期日开始到星期六结束

格式如下: 日 一 二 三 四 五 六 1 2 3 4 5 6

7 8 9 10 11 12 13

- 2)、用户输入一个日期，屏幕打印是星期几。
- 3)、用户输入两个日期，计算着两个日期之间相差几天。
- 4)、输入某年某月某日，判断这一天是这一年的第几天？

输入时的校验功能要求:

- 1、日期正误功能：判断year是否大于0，是否为闰年；
判断month是否大于0且小于等于12；
判断day是否大于0且小于该月份的最大天数；
- 2、重新输入功能：如果日期有误，返回上层重新输入日期；
- 3、计算某年某月的天数。

在控制台输入年份和月份值，然后计算该年该月的天数，并输出在控制台