1. Date类
2. 定义

代表了一个特定的时间，精确到毫秒

1. 构造方法
2. public Date()

构造并初始化一个Date对象，表示该对象创建时的时间，精确到毫秒

1. public Date(long date)

构造并初始化一个Date对象，传入参数date，将其初始化为标准基准时间延后date毫秒的日期。标准基准时间为1970年7月1日0:00，中国时区为东8区，因此起始时间为1970年7月1日8:00。

示例：

public class Date类 {  
 public static void main(String[] args) {  
 Date d1 = new Date();  
 System.*out*.println(d1);  
  
// long date = 1000 \* 60 \* 60\*60;  
 long date = 0;  
 Date d2 = new Date(date);  
 System.*out*.println(d2);  
 }  
}

1. Date类常用方法
2. Public long getTime()

获取该日期对象从标准起始时间到现在的毫秒值。

1. Public void setTime(long time)

传入毫秒值time,设置时间为从标准起始时间开始偏移time毫秒数的时间。

示例：

public class Date类 {  
 public static void main(String[] args) {  
 Date d1 = new Date();  
 System.*out*.println(d1);  
  
// long date = 1000 \* 60 \* 60\*60;  
 long date = 0;  
 Date d2 = new Date(date);  
 System.*out*.println(d2);  
  
 System.*out*.println(d2.getTime());  
 //long time = System.currentTimeMillis();  
 long time = 1000 \* 60 \* 60;  
 d2.setTime(time);  
 System.*out*.println(d2);  
 }  
}

1. SimpleDateFormat类

该类用于设置日期的具体格式和解析日期，相当于该类只是用于设置日期的格式。

1. 构造方法
2. Public SimpleDateFormat()

使用默认模式和日期格式

1. Public SimpleDateFormat(String pattern)

使用给定的日期格式

1. 常用方法
2. 格式化

从Date对象转换为String对象，将日期格式化为时间字符串

Public final String format(Date date)

1. 解析

从String对象转换为Date对象，将时间字符串转换为日期对象

Public Date parse(String source)

1. 示例

package 日期类;  
  
import java.text.ParseException;  
import java.text.SimpleDateFormat;  
import java.util.Date;  
  
public class SimpleDateFormat类 {  
 public static void main(String[] args) throws ParseException {  
 Date d = new Date();  
 SimpleDateFormat sdf = new SimpleDateFormat("yyyy年MM月dd日 HH:mm:ss");  
 String s = sdf.format(d);  
 System.*out*.println(s);  
  
 String ss= "2012-01-01 11:11:11";  
 SimpleDateFormat sdf2 = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");  
 Date dd = sdf2.parse(ss);  
 System.*out*.println(dd);  
 }  
}