1. 獲取系統當前時間：System類中的currentTimeMillis()

**public** **void** test1(){

**long** time = System.*currentTimeMillis*();

//返回當前時間與1970年1月1日0時0分0秒之間以毫秒為單位的時間差。

//稱為時間戳

System.***out***.println(time);

1. java.util.Data類與java.sql.Data類

java.util.Date類

|---java.sql.Date類

1.兩個構造器的使用

>構造器一：Date()：創建一個對應當前時間的Date對象

>構造器二：創建指定毫秒數的Date對象

2.兩個方法的使用

>toString():顯示當前的年、月、日、時、分、秒

>getTime():獲取當前Date對象對應的毫秒數。（時間戳）

3. java.sql.Date對應著數據庫中的日期類型的變量

>如何實例化

>如何將java.util.Date對象轉換為java.sql.Date對象

\*/

@Test

**public** **void** test2(){

//構造器一：Date()：創建一個對應當前時間的Date對象

Date date1 = **new** Date();

System.***out***.println(date1.toString());//Sat Feb 16 16:35:31 GMT+08:00 2019

System.***out***.println(date1.getTime());//1550306204104

//構造器二：創建指定毫秒數的Date對象

Date date2 = **new** Date(155030620410L);

System.***out***.println(date2.toString());

//創建java.sql.Date對象

java.sql.Date date3 = **new** java.sql.Date(35235325345L);

System.***out***.println(date3);//1971-02-13

//如何將java.util.Date對象轉換為java.sql.Date對象

//情況一：

// Date date4 = new java.sql.Date(2343243242323L);

// java.sql.Date date5 = (java.sql.Date) date4;

//情況二：

Date date6 = **new** Date();

java.sql.Date date7 = **new** java.sql.Date(date6.getTime());

}

1. java.text.simpleDataFormat類

*SimpleDateFormat对日期Date类的格式化和解析*

*1.两个操作：  
1.1 格式化：日期 --->字符串  
1.2 解析：格式化的逆过程，字符串 ---> 日期  
  
2.SimpleDateFormat的实例化:new + 構造器*

*//\*\*\*\*\*\*\*\*\*\*\*\*\*按照指定的方式格式化和解析：调用带参的构造器\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
// SimpleDateFormat sdf1 = new SimpleDateFormat("yyyyy.MMMMM.dd GGG hh:mm aaa");* SimpleDateFormat sdf1 = new SimpleDateFormat("yyyy-MM-dd hh:mm:ss");  
 *//格式化* String format1 = sdf1.format(date);  
 System.*out*.println(format1);*//2019-02-18 11:48:27  
 //解析:要求字符串必须是符合SimpleDateFormat识别的格式(通过构造器参数体现),  
 //否则，抛异常* Date date2 = sdf1.parse("2020-02-18 11:48:27");  
 System.*out*.println(date2);  
 }

小練習:

*/\*  
 练习一：字符串"2020-09-08"转换为java.sql.Date  
 练习二："三天打渔两天晒网" 1990-01-01 xxxx-xx-xx 打渔？晒网？  
 举例：2020-09-08 ？ 总天数  
 总天数 % 5 == 1,2,3 : 打渔 总天数 % 5 == 4,0 : 晒网  
 总天数的计算？  
方式一：( date2.getTime() - date1.getTime()) / (1000 \* 60 \* 60 \* 24) + 1  
方式二：1990-01-01 --> 2019-12-31 + 2020-01-01 -->2020-09-08  
 \*/* @Test  
 public void testExer() throws ParseException {  
 String birth = "2020-09-08";  
 SimpleDateFormat sdf1 = new SimpleDateFormat("yyyy-MM-dd");  
 Date date = sdf1.parse(birth);  
*// System.out.println(date);* java.sql.Date birthDate = new java.sql.Date(date.getTime());  
 System.*out*.println(birthDate);  
 }

1. Calendar類：日曆類、抽象類
2. *//1.实例化  
    //方式一：创建其子类（GregorianCalendar）的对象  
    //方式二：调用其静态方法getInstance()* Calendar calendar = Calendar.*getInstance*();  
   *// System.out.println(calendar.getClass());  
    //2.常用方法  
    //get()* int days = calendar.get(Calendar.*DAY\_OF\_MONTH*);  
    System.*out*.println(days);  
    System.*out*.println(calendar.get(Calendar.*DAY\_OF\_YEAR*));  
    *//set()  
    //calendar可变性* calendar.set(Calendar.*DAY\_OF\_MONTH*,22);  
    days = calendar.get(Calendar.*DAY\_OF\_MONTH*);  
    System.*out*.println(days);  
     
    *//add()* calendar.add(Calendar.*DAY\_OF\_MONTH*,-3);  
    days = calendar.get(Calendar.*DAY\_OF\_MONTH*);  
    System.*out*.println(days);  
    *//getTime():日历类---> Date* Date date = calendar.getTime();  
    System.*out*.println(date);  
    *//setTime():Date ---> 日历类* Date date1 = new Date();  
    calendar.setTime(date1);  
    days = calendar.get(Calendar.*DAY\_OF\_MONTH*);  
    System.*out*.println(days);