

java.util.Calendar

- description : java.util.Calendar
- author :
- email : shlim@repia.com
- lastupdate : 2022-06-30 Thu

java.util.Calendar

Calendar

가

가

Calendar

,

```
Calendar cal = new Calendar(); // 가(X)
Calendar cal = Calendar.getInstance(); // Calendar (0)
```

```
import java.util.Calendar;

public class CalendarEx {

    public static void main(String[] args) {
        Calendar time = Calendar.getInstance();
        System.out.println("time>> " +time);
        //time>>
        java.util.GregorianCalendar[time=1656579711756,areFieldsSet=true,areAllField
        sSet=true,
        //lenient=true,zone=sun.util.calendar.ZoneInfo[id="Asia/Seoul",offset=324000
        00,dstSavings=0,
        //useDaylight=false,transitions=30,lastRule=null],firstDayOfWeek=1,minimalDa
        ysInFirstWeek=1,
        //ERA=1,YEAR=2022,MONTH=5,WEEK_OF_YEAR=27,WEEK_OF_MONTH=5,DAY_OF_MONTH=30,DA
        Y_OF_YEAR=181,
        //DAY_OF_WEEK=5,DAY_OF_WEEK_IN_MONTH=5,AM_PM=1,HOUR=6,HOUR_OF_DAY=18,MINUTE=
        1,SECOND=51,
        //MILLISECOND=756,ZONE_OFFSET=32400000,DST_OFFSET=0]
        System.out.println("Date is before add is" + time.getTime());
```

```
//Date is before add isThu Jun 30 18:07:10 KST 2022

time.add(Calendar.DATE, 7);
// Calendar.DATE    DATE    DAY_OF_MONTH    30

//      6    30    + 7                7    7

System.out.println("Date is after add is " + time.getTime());
//Date is after add is Thu Jul 07 18:06:05 KST 2022
}
```

Field Detail

MONTH

Field number for get and set indicating the month. This is a calendar-specific value. The first month of the year in the Gregorian and Julian calendars is JANUARY which is 0; the last depends on the number of months in a year.

() get set
1 0 JANUARY 1

DATE

Field number for get and set indicating the day of the month. This is a synonym for DAY_OF_MONTH. The first day of the month has value 1.

() get, set . Date DAY_OF_MONTH
1

DAY_OF_MONTH

Field number for get and set indicating the day of the month. This is a synonym for DATE. The first day of the month has value 1.

() get, set . DAY_OF_MONTH DATE
1

DAY_OF_YEAR

Field number for get and set indicating the day number within the current year. The first day of the year has value 1.

get, set 1

DAY_OF_WEEK

Field number for get and set indicating the day of the week. This field takes values SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, and SATURDAY.

get, set
 , 5= , 6= , 7= . 1= , 2= , 3= , 4=

```
import java.util.Calendar;

public class CalendarEx1 {

    public static void main(String[] args) {

        Calendar today = Calendar.getInstance();

        System.out.println(today.getTime());
        //Thu Jun 30 19:31:24 KST 2022

        System.out.println("        (YEAR) : " +
today.get(Calendar.YEAR));
        //        (YEAR) : 2022

        System.out.println("        ( , MONTH) : " +
today.get(Calendar.MONTH));
        //        ( , MONTH) : 5

        System.out.println("        (DATE) : " + today.get(Calendar.DATE));
        //        (DATE) : 30

        System.out.println("        (DAY_OF_MONTH) : " +
today.get(Calendar.DAY_OF_MONTH));
        //        (DAY_OF_MONTH) : 30

        System.out.println("        (DAY_OF_WEEK) : " +
today.get(Calendar.DAY_OF_WEEK));
        //        (DAY_OF_WEEK) : 5

        System.out.println("        ( )        (DAY_OF_WEEK_IN_MONTH) : "
+ today.get(Calendar.DAY_OF_WEEK_IN_MONTH));
        //        ( )        (DAY_OF_WEEK_IN_MONTH) : 5( )

        System.out.println("        ?        ?(AM_PM) : " +
today.get(Calendar.AM_PM));
        //        ?        ?(AM_PM) : 1 // 0: , 1:

        System.out.println("        (0~11, HOUR) : " + today.get(Calendar.HOUR));
        //        (0~11, HOUR) : 7

        System.out.println("        (0~23, HOUR_OF_DAY) : " +
```

```
today.get(Calendar.HOUR_OF_DAY));
    // (0~23, HOUR_OF_DAY) : 19

    System.out.println(" (0~59, MINUTE) : " +
today.get(Calendar.MINUTE));
    // (0~59, MINUTE) : 31

    System.out.println(" (0~59, SECOND) : " +
today.get(Calendar.SECOND));
    // (0~59, SECOND) : 24

    System.out.println("1000    1 (0~999, MILLISECOND) : " +
today.get(Calendar.MILLISECOND));
    //1000    1 (0~999, MILLISECOND) : 733

    System.out.println("                (    ): " +
today.getActualMaximum(Calendar.DATE));
    //                (    ): 30
}
}
```

Ref Link

[java.util.Calendar](#)

[TCP School - Calendar](#)

„ [java.util.Calendar](#)

From:
<https://125.132.25.164/dokuwiki/> -
2023.12

Permanent link:
<https://125.132.25.164/dokuwiki/doku.php?id=wiki:miscellaneous:java.util.calendar>

Last update: **2023/01/13 18:44**

