java.util.Calendar

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

java.util.Calendar

Calendar 클래스는 자바에서 날짜와 시간에 관한 데이터를 손수비게 처리할 수 있도록 제공해주는 추상 클래스입니다.
이 클래스가 추상 클래스롤 선언된 이유는 나라마다 사용한느 달력 체계가 조금씩 다를 수 있기 때문입니다.

Calendar 클래스는 추상클래스이기 때문에 직접 객체를 생성할 수 없고, 메소드를 통해서 완전히 구현된 클래스의 인스턴스를 얻어야 합니다.

```
Calendar cal = new Calendar();    // 에러 발생, 추상클래스는 인스턴스 생성 불가(X)

Calendar cal = Calendar.getInstance();   // Calendar 클래스의 인스턴스 반환 (O)
```

```
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,areAllFieldsSet=true,
//lenient=true,zone=sun.util.calendar.ZoneInfo[id="Asia/Seoul",offset=32400000,dstSavings=0,
//useDaylight=false,transitions=30,lastRule=null],firstDayOfWeek=1,minimalDaysInFirstWeek=1,
//ERA=1,YEAR=2022,MONTH=5,WEEK_OF_YEAR=27,WEEK_OF_MONTH=5,DAY_OF_MONTH=30,DAY_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.DATE에서 DATE는 DAY_OF_MONTH와 동일하며 이 값은 위에서 30이라는 것을 확인할 수 있다.
        // 그래서 6월 30일 + 7일을 하여 아래와 같이 7월 7일이 나온다.

        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.
달(월)을 나타내는 get 및 set의 필드 번호입니다. 이것은 캘린더의 특정 값입니다. 그레고리력과 율리우스력에서 연도의 1년의 첫 번째 달은 0인 JANUARY입니다. 마지막은 1년의 개월 수에 따라 다릅니다.

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.
달(월)의 날짜를 나타내는 get, set의 필드 번호입니다. Date는 DAY_OF_MONTH와 동의어입니다. 해당 달의 첫 번째 날의 값은 1입니다.

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.
달(월)의 날짜를 나타내는 get, set의 필드 번호입니다. DAY_OF_MONTH는 DATE와 동의어입니다. 해당 달의 첫 번째 날의 값은 1입니다.

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.
현재 연도 내의 날짜 번호를 나타내는 get, set의 필드 번호입니다. 연도의 첫 번째 날짜의 값은 1입니다.

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.
요일을 나타내는 get, set의 필드 번호입니다. 이 필드의 값은 1=일요일, 2=월요일, 3=화요일, 4=수요일, 5=목요일, 6=금요일, 그리고 7=토요일입니다.

```
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("올 해의 연도(YEAR) : " +
today.get(Calendar.YEAR));
        //올 해의 연도(YEAR) : 2022

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

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

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

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

        System.out.println("현재 달(월)의 몇 번째 요일(DAY_OF_WEEK_IN_MONTH) : "
+ today.get(Calendar.DAY_OF_WEEK_IN_MONTH));
        //현재 달(월)의 몇 번째 요일(DAY_OF_WEEK_IN_MONTH) : 5(목요일)

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

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

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

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

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

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

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

        System.out.println("이 달의 마지막 날(일수): " +
today.getActualMaximum(Calendar.DATE));
        //이 달의 마지막 날(일수): 30
    }
}
```

Ref LInk

java.util.Calendar

TCP School - Calendar 클래스

오션,, java.util.Calendar

From:
http://125.132.25.164/dokuwiki/ - **문제를 잘 정의하는 것은 문제를 절반 해결한 것이다. - 2023.12**

Permanent link:
http://125.132.25.164/dokuwiki/doku.php?id=wiki:miscellaneous:java.util.calendar&rev=1656585356

Last update: **2022/06/30 19:35**