

chapter10 날짜와 시간 & 형식화

ch10-1 날짜와 시간

java.util.Date - 날짜와 시간을 다룰 목적인 클래스

95년 버전 여전히 사용하고 있다.

java.util.Calendar - Date클래스를 개선한 클래스

java.time 패키지

-Date와 Calendar의 단점을 개선한 새로운 클래스 제공

ch10-2 Calendar클래스와 get메서드

-추상 클래스이므로 객체를 생성할 수 없다.

-getInstance()를 통해 구현된 객체를 얻어야 한다.

```
Calendar cal = new Calendar(); //에러
```

```
Calendar cal = Calendar.getInstance();
```

get()으로 날짜와 시간 필드 가져오기 - int get(int field)

```
Calendar cal = Calendar.getInstance(); // 현재 날짜와 시간으로 셋팅됨
int thisYear = cal.get(Calendar.YEAR); // 올해가 몇년인지 알아낸다.
int lastDayOfMonth = cal.getActualMaximum(Calendar.DATE); // 이 달의 마지막날
```

• Calendar에 정의된 필드

필드명	설명	필드명	설명
YEAR	년	HOUR	시간(0~11)
MONTH	월(0부터 시작)	HOUR_OF_DAY	시간(0~23)
WEEK_OF_YEAR	일	MINUTE	분
WEEK_OF_MONTH	그 달의 몇 번째 주	SECOND	초
DATE	일	MILLISECOND	천분의 일초
DAY_OF_MONTH	그 달의 몇 번째일	ZONE_OFFSET	GMT기준 시차(천분의 일초 단위)
DAY_OF_YEAR	그 해의 몇 번째일	AM_PM	오전/오후
DAY_OF_WEEK	요일		
DAY_OF_WEEK_IN_MONTH	그 달의 몇 번째 요일		

ch10-4 Calendar클래스의 set메서드

-set()으로 날짜와 시간지정하기

-날짜 지정하는 방법. 월(MONTH)이 0부터 시작한다는 점에 주의

```
Calendar date1 = Calendar.getInstance();
date1.set(2017, 7, 15); // 2017년 8월 15일(7월 아님)
// date1.set(Calendar.YEAR, 2017);
// date1.set(Calendar.MONTH, 7);
// date1.set(Calendar.DATE, 15);
```

-시간 지정하는 방법.

```
Calendar time1 = Calendar.getInstance();
time1.set(Calendar.HOUR_OF_DAY, 10); // time1을 10시 20분 30초로 설정
time1.set(Calendar.MINUTE, 20);
time1.set(Calendar.SECOND, 30);
```

ch10-4 Calendar클래스의 clear메서드

-Calendar객체의 모든 필드를 초기화

객체 생성 후 반드시 클리어를 항상 해줘야한다.

```
Calendar dt = Calendar.getInstance();

// Tue Aug 29 07:13:03 KST 2017
System.out.println(new Date(dt.getTimeInMillis()));

dt.clear(); // 모든 필드를 초기화
// Thu Jan 01 00:00:00 KST 1970
System.out.println(new Date(dt.getTimeInMillis()));
```

-clear(int field)는 Calendar객체의 특정 필드를 초기화

```
Calendar dt = Calendar.getInstance();

// Tue Aug 29 07:13:03 KST 2017
System.out.println(new Date(dt.getTimeInMillis()));

dt.clear(Calendar.SECOND); // 초를 초기화
dt.clear(Calendar.MINUTE); // 분을 초기화
dt.clear(Calendar.HOUR_OF_DAY); // 시간을 초기화
dt.clear(Calendar.HOUR); // 시간을 초기화

// Tue Aug 29 00:00:00 KST 2017
System.out.println(new Date(dt.getTimeInMillis()));
```

ch10-6,7 Calendar클래스의 add, roll메서드

-add()는 필드의 값을 증가 또는 감소 (다른 필드에 영향o)

-roll()는 필드의 값을 증가 또는 감소 (다른 필드에 영향x)

ch10-8 Date와 Calendar간의변환

Date의 메서드는 대부분 deprecated되었지만 계속사용

```
1. Calendar를 Date로 변환
Calendar cal = Calendar.getInstance();
...
Date d = new Date(cal.getTimeInMillis()); // Date(long date)

2. Date를 Calendar로 변환
Date d = new Date();
...
Calendar cal = Calendar.getInstance();
cal.setTime(d)
```

ch10-9 형식화 클래스

-java.text패키지의 DecimalFormat,SimpleDateFormat

-숫자와 날짜를 원하는 형식으로 쉽게 출력 바꾸는 것

<숫자,날짜 -> 형식 문자열>

```
double number = 1234567.89;
DecimalFormat df = new DecimalFormat("#.##E0");
String result = df.format(number); // result = "1.2E6"
```

<형식 문자열 -> 숫자,날짜>

```
DecimalFormat df = new DecimalFormat("#,###.##");
Number num = df.parse("1,234,567.89");
double d = num.doubleValue(); // 1234567.89
```

ch10-10 DecimalFormat

-숫자를 형식화 (다양한 형식화 가능)

ch10-11 SimpleDateFormat

-날짜와 시간을 다양한 형식으로 출력

```
Date today = new Date();
SimpleDateFormat df = new SimpleDateFormat("yyyy-MM-dd");

// 오늘 날짜를 yyyy-MM-dd형태로 변환하여 반환한다.
String result = df.format(today);
```

기호	의미	보기
G	연대(BC, AD)	AD
y	년도	2006
M	월(1~12 또는 1월~12월)	10 또는 10월, OCT
w	년의 몇 번째 주(1~53)	50
W	월의 몇 번째 주(1~5)	4
D	년의 몇 번째 일(1~366)	100
d	월의 몇 번째 일(1~31)	15
F	월의 몇 번째 요일(1~5)	1
E	요일	월

-특정 형식으로 되어있는 문자열에서 날짜와 시간을 뽑아 낼 수 도 있다.

```
DateFormat df = new SimpleDateFormat("yyyy년 MM월 dd일");
DateFormat df2 = new SimpleDateFormat("yyyy/MM/dd");
Date d = df.parse("2015년 11월 23일"); // 문자열을 Date로 변환
String result = df2.format(d);
```

a	오전/오후(AM, PM)	PM
H	시간(0~23)	20
k	시간(1~24)	13
K	시간(0~11)	10
h	시간(1~12)	11
m	분(0~59)	35
s	초(0~59)	55
S	천분의 일초(0~999)	253
z	Time zone(General time zone)	GMT+9:00
Z	Time zone(RFC 822 time zone)	+0900
'	escape문자(특수문자를 표현하는데 사용)	없음