

## Date, Calendar & DateFormat Objects

`java.util.Date`:  
a specific instant in time.

`java.util.Calendar`:  
conversion of `Date` to integer fields.

`java.text.DateFormat`:  
format and parse date/time strings.

1

---

---

---

---

---

---

---

---

## Date

A `Date` object represents a specific instant in time.

- In old versions of Java the `Date` object was used for lots of things, now `Calendar` does most of the work.

Most of the methods are deprecated!

Non-deprecated use includes:

- conversion to/from the number of milliseconds since Jan 1, 1970.
- Comparison of dates (after, before, equals, ...)

2

---

---

---

---

---

---

---

---

## Print Current Date/Time

```
import java.util.*;
class DatePlay {
    static public void main(String []args) {
        Date d = new Date();
        System.out.println("The date is " + d);
    }
}
```

```
> java DatePlay
The date is Sat Mar 01 20:46:16 EST 2003
```

3

---

---

---

---

---

---

---

---

## Calendar

abstract class!

- `GregorianCalendar` is the only standard derived class included (1.4).
- Object Factory:  
`static Calendar getInstance();`
- `Locale` and `TimeZone` Objects

The idea is use a `Calendar` object to take care of date/time handling details in a manner that supports the conventions/calendar type of the user.

4

---

---

---

---

---

---

---

---

## Calendar Object

Can set/get integer values for day, month, year, hour, minute, second, day of the week, etc.

Two (overloaded) methods:

```
set(int field, int value);  
get(int field);
```

You have to know what the value of `field` should be to get what you want!

- The `Calendar` class provides some static int fields that make this possible.

5

---

---

---

---

---

---

---

---

## Calendar Fields

<code>Calendar.DATE</code>	<code>Calendar.MONTH</code>
<code>Calendar.YEAR</code>	<code>Calendar.DAY_OF_WEEK</code>
<code>Calendar.DAY_OF_MONTH</code>	<code>Calendar.DAY_OF_YEAR</code>
<code>Calendar.HOUR</code>	<code>Calendar.MINUTE</code>
<code>Calendar.SECOND</code>	<code>Calendar.MILLISECOND</code>

There are lots more!

6

---

---

---

---

---

---

---

---

## Calendar Example

```
import java.util.*;
class CalPlay {
    static public void main(String []args) {
        Calendar c = Calendar.getInstance();
        System.out.println("Today is " +
            (c.get(Calendar.MONTH)+1) + "/" +
            c.get(Calendar.DATE) + "/" +
            c.get(Calendar.YEAR));
    }
}
```

7

---

---

---

---

---

---

---

---

## Another Example

```
class CalPlay {
    final static String[] DAYS ={"Sunday","Monday",
        "Tuesday", "Wednesday", "Thursday",
        "Friday", "Saturday" };

    static public void main(String []args) {
        Calendar c = Calendar.getInstance();
        System.out.println("Today is " + dayName(c));
    }
    static String dayName( Calendar c) {
        return(DAYS[c.get(Calendar.DAY_OF_WEEK)-1]);
    }
}
```

8

---

---

---

---

---

---

---

---

## Even Better Example

tomorrow.java and bettertomorrow.java

- prints out the date tomorrow.

needs to watch for new month, year, etc.

Does this for any date entered as month day year  
on the command line (as three integers).

```
> java tomorrow 12 31 2001
Tomorrow is Tuesday January 1, 2002
```

9

---

---

---

---

---

---

---

---

## Homework

Write a program that prints out the (single month) calendar for any month.

- Just like the Unix “cal” command.

```
> java cal 3 2003
March 2003
Su Mo Tu We Th Fr Sa
                1
 2  3  4  5  6  7  8
 9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30 31
```

```
> java cal 1 2025
January 2025
Su Mo Tu We Th Fr Sa
                1  2  3  4
 5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31
```

10

---

---

---

---

---

---

---

---

## DateFormat

Another abstract class

Factory method:

- `static DateFormat getDateInstance()`

Uses a Calendar to format dates.

- associated with a `Locale`

A number of pre-defined formats (styles):

- short, medium, long, full
- `DateFormat.SHORT`, `DateFormat.MEDIUM`, ...

11

---

---

---

---

---

---

---

---

## DateFormat Example

```
public static void main(String [] args) {
    Date now = new Date();
    // try with default Locale
    showDate(DateFormat.SHORT,now);
    showDate(DateFormat.MEDIUM,now);
    showDate(DateFormat.LONG,now);
    showDate(DateFormat.FULL,now);
}

static void showDate( int style, Date d) {
    System.out.println(
        DateFormat.getDateInstance(style).format(d));
}
```

```
3/1/03
Mar 1, 2003
March 1, 2003
Saturday, March 1, 2003
```

12

---

---

---

---

---

---

---

---

## DateFormat Exemple (Locale for France)

```
public static void main(String [] arg) {
    Date now = new Date();
    showDate(DateFormat.SHORT,now);
    showDate(DateFormat.MEDIUM,now);
    showDate(DateFormat.LONG,now);
    showDate(DateFormat.FULL,now);
}
static void showDate( int style, Date d) {
    System.out.println(
        DateFormat.getDateInstance(style,
            Locale.FRANCE).format(d));
}
```

```
01/03/03
1 mars 2003
1 mars 2003
samedi 1 mars 2003
```

13

---

---

---

---

---

---

---

---