```
 1: // $Id: epochdates.java,v 1.93 2013-03-28 15:49:06-07 - - $
 2: //
 3: // Prints out some dates and times.
 4: // Illustrates the use of a date formatter.
 5: //
 6:
 7: import java.text.*;
 8: import java.util.*;
 9: import static java.lang.Math.*;
10: import static java.lang.System.*;
11:
12: class epochdates {
13:    static final GregorianCalendar CHANGE_DATE
14:                 = new GregorianCalendar (1752, Calendar.SEPTEMBER, 14);
15:    static final double BIG_BANG = -13.798e9; //years
16:    static final double RED_GIANT = 5e9; //years
17:    static final double YEAR_SEC = 365.2422 * 24 * 60 * 60;
18:
19:    static long make_calendar (int year, int month, int day) {
20:       GregorianCalendar cal = new GregorianCalendar(0,0,0,0,0,0);
21:       cal.setGregorianChange (CHANGE_DATE.getTime());
22:       if (year > 0) {
23:          cal.set (Calendar.ERA, GregorianCalendar.AD);
24:          cal.set (year, month, day);
25:       }else if (year < 0) {
26:          cal.set (Calendar.ERA, GregorianCalendar.BC);
27:          cal.set (-year, month, day);
28:       }else {
29:          throw new IllegalArgumentException ("year == 0");
30:       }
31:       return cal.getTimeInMillis();
32:    }
33:
34:    static long[] times = {
35:       Long.MIN_VALUE,
36:       make_calendar (-1178, Calendar.APRIL,    16),
37:       make_calendar ( -753, Calendar.APRIL,    21),
38:       make_calendar (    1, Calendar.JANUARY,   1),
39:       make_calendar ( 1066, Calendar.OCTOBER,  14),
40:       Integer.MIN_VALUE * 1000L,
41:       0L,
42:       currentTimeMillis(),
43:       Integer.MAX_VALUE * 1000L,
44:       make_calendar ( 9999, Calendar.DECEMBER, 31),
45:       Long.MAX_VALUE,
46:    };
47:
```

```
48:
49:    public static void main (String[] args) {
50:       TimeZone gmt = new SimpleTimeZone (0, "GMT");
51:       Calendar cal = new GregorianCalendar ();
52:       out.printf ("%,24.0f = %-19s%,16.0f BCE%n", BIG_BANG * YEAR_SEC,
53:                   "Big Bang", BIG_BANG);
54:       for (long time : times) {
55:          cal.setTimeInMillis (time);
56:          cal.setTimeZone (gmt);
57:          String date = String.format ("%1$tA, %1$tB %1$te,", cal);
58:          out.printf ("%,24.0f = %-24s", time / 1e3, date);
59:          int year = cal.get (Calendar.YEAR);
60:          out.printf (abs (year) <= 9999 ? "%11d" : "%,11d", year);
61:          out.printf (" %s", cal.get (Calendar.ERA)
62:                       == GregorianCalendar.AD ? "CE" : "BCE");
63:          if (time >= Integer.MIN_VALUE * 1000L &&
64:              time <= Integer.MAX_VALUE * 1000L) {
65:             out.printf (" %1$tT %1$TZ", cal);
66:          }
67:          out.printf ("%n");
68:       }
69:       out.printf ("%,24.0f = %-19s%,16.0f CE%n", RED_GIANT * YEAR_SEC,
70:                   "Sun is Red Giant", RED_GIANT);
71:    }
72:
73: }
74:
75: //TEST// ./epochdates >epochdates.out 2>&1
76: //TEST// mkpspdf epochdates.ps epochdates.java* epochdates.out
77:
```

```
1: @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: starting epochdates.java
2: epochdates.java:
3:      $Id: epochdates.java,v 1.93 2013-03-28 15:49:06-07 - - $
4: javac -Xlint epochdates.java
5: jar cfm epochdates Manifest epochdates.class
6: chmod +x epochdates
7: rm -f epochdates.class
8: @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: finished epochdates.java
```

```
    1: -435,422,466,051,839,940 = Big Bang            -13,798,000,000 BCE
    2:   -9,223,372,036,854,776 = Sunday, December 2,     292,269,055 BCE
    3:          -99,301,564,800 = Wednesday, April 16,            1178 BCE
    4:          -85,889,088,000 = Monday, April 21,                753 BCE
    5:          -62,135,740,800 = Saturday, January 1,               1 CE
    6:          -28,502,208,000 = Saturday, October 14,           1066 CE
    7:           -2,147,483,648 = Friday, December 13,            1901 CE 20:45:
52 GMT
    8:                        0 = Thursday, January 1,            1970 CE 00:00:
00 GMT
    9:            1,420,252,452 = Saturday, January 3,            2015 CE 02:34:
12 GMT
   10:            2,147,483,647 = Tuesday, January 19,            2038 CE 03:14:
07 GMT
   11:          253,402,243,200 = Friday, December 31,            9999 CE
   12:    9,223,372,036,854,776 = Sunday, August 17,        292,278,994 CE
   13:  157,784,630,399,999,968 = Sun is Red Giant        5,000,000,000 CE
```