Zeller’s Congruence | Find the Day for a Date

Zeller’s congruence is an algorithm devised by Christian Zeller to **calculate the day of the week for any Julian or Gregorian calendar date**. It can be considered to be based on the conversion between Julian day and the calendar date.  
It is an algorithm to find the day of the week for any date.  
**For the Gregorian calender it is:**  
  
**For the julian calender it is:**

where,

1. h is the day of the week (0 = Saturday, 1 = Sunday, 2 = Monday, …, 6 = Friday)
2. q is the day of the month
3. m is the month (3 = March, 4 = April, 5 = May, …, 14 = February)
4. K the year of the century ( year % 100).
5. J is the zero-based century (actually ⌊ year/100 ⌋ ) For example, the zero-based centuries for 1995 and 2000 are 19 and 20 respectively (to not be confused with the common ordinal century enumeration which indicates 20th for both cases).

NOTE: In this algorithm January and February are

counted as months 13 and 14 of the previous

year.E.g. if it is 2 February 2010, the

algorithm counts the date as the second day

of the fourteenth month of 2009 (02/14/2009

in DD/MM/YYYY format)

For an ISO week date Day-of-Week d (1 = Monday to 7 = Sunday), use

d = ((h+5)%7) + 1

|  |
| --- |
| // Java program to find Find the Day  // for a Date  import java.util.\*;    class GFG  {      // Print Day for a Date      static void Zellercongruence(int day, int month,                                   int year)      {          if (month == 1)          {              month = 13;              year--;          }          if (month == 2)          {              month = 14;              year--;          }          int q = day;          int m = month;          int k = year % 100;          int j = year / 100;          int h = q + 13\*(m + 1) / 5 + k + k / 4 + j / 4 + 5 \* j;          h = h % 7;          switch (h)          {              case 0 : System.out.println("Saturday"); break;              case 1 : System.out.println("Sunday"); break;              case 2 : System.out.println("Monday"); break;              case 3 : System.out.println("Tuesday"); break;              case 4 : System.out.println("Wednesday"); break;              case 5 : System.out.println("Thursday"); break;              case 6 : System.out.println("Friday"); break;          }      }        // Driver code      public static void main(String[] args)      {          Zellercongruence(22, 10, 2017); //date (dd/mm/yyyy)      }  } |

**Output:**

Sunday