```java
/*-------------------------------------------------------
My name: Andree Robert Agustian
My student number: 5182086
My course: CSIT111
My email address: ara968@uowmail.edu.au
Assignment number: 3
------------------------------------------------------*/
import java.util.Scanner; // program uses class Scanner

class MyCalendar {
        private static MyDate myDate; // instance variable of MyDate data type

        // enumeration data type which contains the days of the week
        enum Day {Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday}

        private Day day; // instance variable of Day data type

        public static void main(String[] args) { // main method
                String inputDate = args[0]; // input date from the command line
                int d = Integer.parseInt(inputDate.substring(0,2)); // extract the day from the input and conver
t to integer
                int m = Integer.parseInt(inputDate.substring(3,5)); // extract the month from the input and conv
ert to integer
                int y = Integer.parseInt(inputDate.substring(6,10)); // extract the year from the input and conv
ert to integer

                                new MyDate(d,m,y); // create object myDate from class MyDate with parameters d,m
,y

                                        new MyCalendar(myDate); // create myCalendar object to call method in cl
ass MyCalendar
                                new Scanner(System.in); // create a Scanner to obtain input from command window

                // a loop that detects if the input is not valid, it requires to re-input date
                while (myDate.isDateValid(myCalendar.getTotalDays(myDate.getMonth(),myDate.getYear())) == false)
 {
                        System.out.print(inputDate + " is not a valid date, please re−input a valid date: ");
                        inputDate = input.nextLine(); // read new input date from user
                        d = Integer.parseInt(inputDate.substring(0,2)); // extract the day from the input and co
nvert to integer
                                                                        // extract the month from the input and
convert to integer
                                                                         // extract the year from the input and
convert to integer
                                                 // assign the new (d,m,y) in myDate
                }

                int myDay = myDate.getDay(); // assign day to myDay
                int myMonth = myDate.getMonth(); // assign month to myMonth
                int myYear = myDate.getYear(); // assignm year to myYear

                // retrieves the total number of days in a month from the given month
                int myTotalDays = myCalendar.getTotalDays(myMonth,myYear);

                // assign calculated zellerCongruence to myZeller
                int myZeller = myCalendar.zellerCongruence(myDay,myMonth,myYear);

                // assign dayOfWeek according to calculated zellerCongruence to myDayOfWeek
                Day myDayOfWeek = myCalendar.dayOfWeek(myZeller);

                // retrieves the day of the week for the first day from the given month and year
                int myFirstDayOfMonth = myCalendar.zellerCongruence(01,myMonth,myYear);

                int myWeek = myCalendar.weekOfMonth(myFirstDayOfMonth,myDay); // assign the week number
                String myWeekName = myCalendar.weekName(myWeek); // assign the week in word format

                String myMonthName = myCalendar.getMonthName(myMonth); // assign month name

                System.out.println(inputDate + " is a " + myDayOfWeek + " and locates in the "
                                   + myWeekName + " week of " + myMonthName + " " + myYear);

                System.out.printf("The calendar of %s %d is:", myMonthName, myYear);

                System.out.println();
                System.out.println();

                // prints calendar of the given start day of the month and total days
                myCalendar.printCalendar(myFirstDayOfMonth,myTotalDays);
        } // end method main

        public MyCalendar(MyDate myDate) { // construction method of the class MyCalendar
                this.myDate = myDate;
        }

        public Day dayOfWeek(int dd) { // method to retrieve the day of the week from the given day
                switch (dd) {
                        case 0:
                                day = (Day.Saturday);
                                break;
                        case 1:
                                day = (Day.Sunday);
                                break;
                        case 2:
                                day = (Day.Monday);
                                break;
                        case 3:
                                day = (Day.Tuesday);
                                break;
                        case 4:
                                day = (Day.Wednesday);
                                break;
                        case 5:
                                day = (Day.Thursday);
                                break;
                        case 6:
                                day = (Day.Friday);
                                break;
                }
                return day;
        }

        // method to return the day of the week for the given date (dd,mm,yyyy)
        public int zellerCongruence(int q, int m, int y) {
                if (m == 01) // if month is January then variable month is changed to 13
                        m = 13;
                else if (m == 02) // if month is February then variable month is changed to 14
                        m = 14;
                else // else month stays the same
                        m = m;

                if (m == 13) // if month is January then variable year is deducted by 1
                        y = y-1;
                else if (m == 14) // if month is February then variable year is deducted by 1
                        y = y-1;
                else // else year stays the same
                        y = y;

                int K = y%100; // year mod 100
                int J = y/100; // year divide by 100

                // calculates Zeller’s Congruence
                int theDay = (q + (13*(m+1))/5 + K + (K/4) + (J/4) + 5*J)%7;

                return theDay;
        } // end method dayOfWeek

        // method to return the week of the month for the given date
        public int weekOfMonth(int firstDay, int calendarDay) {
                int week = 0;
                for (int i = 0; i <= 5; i++) { // row (week) of calender
                        for (int j = 1; j <= 7; j++) { // column (day) of calendar

                                int date = (i*7)+(j-6)-firstDay; // formula as calculated in method printCalendar()
                                if (date == calendarDay) { // if date equals the day then assign week to i (week)




                // fixes week due to Saturday is 0 but Saturday is at the end of the row (where int j at 7)
                // happens only when the first day of the month is a Saturday
                if (firstDay == 0)
                        week = week +1;
                return week;
        } // end method

        public String weekName(int week) { // method to return week in word format
                String weekName;
                if (week == 1) {
                        weekName = "first";
                } else if (week == 2) {
                        weekName = "second";
                } else if (week == 3) {
                        weekName = "third";
                } else if (week == 4) {
                        weekName = "fourth";
                } else if (week == 5) {
                        weekName = "fifth";
                } else {
                        weekName = "sixth";
                }
                return weekName;
        } // end method

        public String getMonthName(int month) { // method to retrieve month name from the given month
                String monthName = "";
                switch (month) {
                        case 1:
                                monthName = "January";
                                break;
                        case 2:
                                monthName = "February";
                                break;
                        case 3:
                                monthName = "March";
                                break;
                        case 4:
                                monthName = "April";
                                break;
                        case 5:
                                monthName = "May";
                                break;
                        case 6:
                                monthName = "June";
                                break;
                        case 7:
                                monthName = "July";
                                break;
                        case 8:
                                monthName = "August";
                                break;
                        case 9:
                                monthName = "September";
                                break;
                        case 10:
                                monthName = "October";
                                break;
                        case 11:
                                monthName = "November";
                                break;
                        case 12:
                                monthName = "December";
                                break;
                }
                return monthName;
        } // end method

        // method to retrieve the total number of days in the given month
        public int getTotalDays(int month, int year) {
                int totalDays = 0;
                switch (month) {
                        case 1: case 3: case 5: case 7: case 8: case 10: case 12:
                                totalDays = 31;
                                break;
                        case 4: case 6: case 9: case 11:
                                totalDays = 30;
                                break;
                        case 2:
                                if (year % 4 != 0) { // not divisible by 4
                                        totalDays = 28;
                                } else if (year % 100 != 0) { // not divisible by 100
                                        totalDays = 29;
                                } else if (year % 400 != 0) { // not divisible by 400
                                        totalDays = 28;
                                } else {
                                        totalDays = 29;
                                }
                                break;
                }
                return totalDays;
        } // end method

        // method to print the calendar
        public void printCalendar(int firstDay, int totalDay) {
                System.out.println("SUN  MON  TUE  WED  THU  FRI  SAT");
                for (int i = 0; i <= 5; i++) { // loop for each row (weeks), up to 6 weeks
                        for (int j = 1; j <= 7; j++) { // loop for each column (days), up to 7 days

                                /*
                                 * prints each row with multiple of 7, (i*7)
                                 * reformat each day and starts where zeller = 0 (Saturday), day 1 is Saturday, +(j-6)
                                 * moves day to the left depending location of first day of the month starts, -firstDay
                                 */

                                int date = (i*7)+(j-6)-firstDay;
                                if (date > totalDay) // breaks the loop when date has reached more than total day
                                        break;
                                if (date <= 0) { // prints blank when day is less than or equal to zero
                                        System.out.printf("   "); // column width is fixed to 3 char + 3 char
                                }
                                else
                                        System.out.printf("%3d  ", date); // column width is fixed to 3 characters

                                              // prints out the next row of the calendar
                }
        } // end method printCalendar

} // end class MyCalendar

class MyDate {
        private int day; // instance variable for day of the month
        private int month; // instance variable for month of the year
        private int year; // instance variable for year of the date
        private boolean DateValid; // instance variable for date validation

        public MyDate(int day, int month, int year) { // constructor of the class MyDate
                this.day = day;
                this.month = month;
                this.year = year;
        }

        public void setMyDate(int day, int month, int year) { // method to set the MyDate in the object
                this.day = day;
                this.month = month;
                this.year = year;
        }

        public MyDate() { // default constructor
                day = 0;
                month = 0;
                year = 0;
        }

        public int getDay() { // method to return the integer day value
```

```java
                return day;
        }

        public int getMonth() { // method to return the integer month value
                return month;
        }

        public int getYear() { // method to return the integer year value
                return year;
        }

        public boolean isDateValid(int totalDays) { // method to check if input date is a valid date
                if ((day <= totalDays) && (day != 0))
                        DateValid = true;
                else
                        DateValid = false;
                return DateValid;
        }

} // end class MyDate
```

```
------------------------------------------------------------------------------------------------------------------------
javac MyCalendar.java
No errors found.


------------------------------------------------------------------------------------------------------------------------
Test input:
java MyCalendar 29/02/2017
31/04/2017
14/06/2017

Expected outout:
29/02/2017 is not a valid date, please re-input a valid date:31/04/2017
31/04/2017 is not a valid date, please re-input a valid date:14/06/2017
14/06/2017 is a Wednesday and locates in the third week of June 2017
The calendar of June 2017 is:

SUN   MON   TUE   WED   THU   FRI   SAT
                        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

**********
java MyCalendar 29/02/2017
29/02/2017 is not a valid date, please re-input a valid date: 31/04/2017 is not a valid date, please re-input a valid da
te: 14/06/2017 is a Wednesday and locates in the third week of June 2017
The calendar of June 2017 is:

SUN   MON   TUE   WED   THU   FRI   SAT

                          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
```

ara968

```
Assignment Marking Result
========================================================================================================================
Assignment 3
***********************************************
General Requirements:
Failure of compilation:
Late penalty:
Enrolment.java:
***********************************************
Total marks:                                15 /15
***********************************************
Other comments:





========================================================================================================================
                                 If you have any inquiries about the marking results, please contact the markoing tutor.
                      All inquiries can only be made within a maimum of one week after the marking results are released.
                                                                      After 1 week, your inquiries may not be responded.
========================================================================================================================
Marked by: Xishun Wang       e-mail: xishun@uow.edu.au
```