```
 1 ////////////////////////////////////////////////////////////////////////////////
 2 // Lab number: 3                                                              //
 3 // Name: Richard Wright                                                       //
 4 // Module Description: Print Calendar                                         //
 5 // Date: 09/21/16                                                             //
 6 ////////////////////////////////////////////////////////////////////////////////
 7
 8 import java.util.Calendar;
 9
10 public class PrintCalendar{
11
12    private int year;
13    private final JulianDate jd = new JulianDate();
14
15    public static void main(String[] args){
16
17       int inputYear;
18
19       PrintCalendar myCal = new PrintCalendar();
20
21       //Checks if user run an argument, else current year is used.
22       if(args.length == 0){
23          myCal.setYear(Calendar.getInstance().get(Calendar.YEAR));
24       }else{
25          myCal.setYear(Integer.parseInt(args[0]));
26       }
27
28       System.out.println("\n");
29
30       myCal.printCal();
31
32    }//end main
33
34    //Sets the inputted year to the year variable used by PrintCalendar
35    public void setYear(int inputYear){
36
37       this.year = inputYear;
38
39    }//end setYear
40
41    //Prints the calendar, and if the year printed is the current year, highlights the curre
nt month
42    public void printCal(){
43
44       System.out.printf("%11d\n\n", year );
45
46       for(int i=1; i<=12; i++){
47
48          if(Calendar.getInstance().get(Calendar.YEAR) == this.year && (Calendar.getInstance
().get(Calendar.MONTH)) + 1 == i){
49             System.out.print("-----------------------\n");
50          }
51
52          printMonth(i);
53
54          if(Calendar.getInstance().get(Calendar.YEAR) == this.year && (Calendar.getInstance
().get(Calendar.MONTH)) + 1 == i){
55             System.out.println("\n");
56             System.out.print("-----------------------\n");
57          }else{
58             System.out.print("\n\n");
59          }
60
61
62       }//end for
63
64    }//end printCal
65
```

```
66    //Method to label months with correct month name (eg month 1 = January)
67    private void printMonth(int month){
68
69       String sMonth = "";
70
71       switch(month){
72          case 1: sMonth = "Jan"; break;
73          case 2: sMonth = "Feb"; break;
74          case 3: sMonth = "Mar"; break;
75          case 4: sMonth = "Apr"; break;
76          case 5: sMonth = "May"; break;
77          case 6: sMonth = "Jun"; break;
78          case 7: sMonth = "Jul"; break;
79          case 8: sMonth = "Aug"; break;
80          case 9: sMonth = "Sep"; break;
81          case 10: sMonth = "Oct"; break;
82          case 11: sMonth = "Nov"; break;
83          case 12: sMonth = "Dec"; break;
84       }//end switch
85
86       System.out.printf("%10s\n\n", sMonth);
87
88       this.printDaysOfWeek();
89
90       this.printDayNumbers(month);
91
92    }//end printMonth
93
94    //Method to label days of week (Su, M, T, etc)
95    private void printDaysOfWeek(){
96
97       for(int i = 0; i <= 6; i++){
98
99          switch(i){
100            case 0: System.out.printf("%3s ", "Su"); break;
101            case 1: System.out.printf("%2s ", "M"); break;
102            case 2: System.out.printf("%2s ", "Tu"); break;
103            case 3: System.out.printf("%2s ", "W"); break;
104            case 4: System.out.printf("%2s ", "Th"); break;
105            case 5: System.out.printf("%2s ", "F"); break;
106            case 6: System.out.printf("%2s \n", "Sa"); break;
107         }//end switch
108
109      }//end for
110
111   }//end printDaysOfWeek
112
113   //Method to number months
114   private void printDayNumbers(int month){
115
116      int numberOfDays = getNumberOfDays(month); //get number of days in month
117
118      //Function to determine what day of the week the first of the month falls on (eg. 0 i
s Sunday, 1 is Monday, etc)
119      int firstDay = ((jd.toJulian(year, month, 1) + 1) % 7) + 1;
120
121      //From result of above function, formats so the first day is under the correct day of
 the week
122      if((firstDay - 1) > 0){
123         System.out.printf("%" + ((firstDay-1)*3) + "c", ' ');
124      }
125
126      //While loop to number the days of the week
127      int currentDay = firstDay;
128      int j = 1;
129
130      while(j <= numberOfDays){
131
```

```
132          System.out.printf( "%3d", j);
133
134          /*if currentDay is 7, makes a new line and resets currentDay value to 0
135            This formats the numbers to drop to a new week when they reach end of current we
ek*/
136          if(currentDay == 7){
137             System.out.println();
138             currentDay = 0;
139          }
140
141          currentDay++;
142          j++;
143
144       }//end while
145
146    }//end printDayNumbers
147
148    //Hard coded number of days in each month (except Feburary). Passed an integer 1-12 and
returns number for resulting month.
149    private int getNumberOfDays(int month){
150
151       int numberOfDays = 0;
152
153       switch(month){
154          case 1: numberOfDays = 31; break;
155          case 2: numberOfDays = this.getFebDays(); break;
156          case 3: numberOfDays = 31; break;
157          case 4: numberOfDays = 30; break;
158          case 5: numberOfDays = 31; break;
159          case 6: numberOfDays = 30; break;
160          case 7: numberOfDays = 31; break;
161          case 8: numberOfDays = 31; break;
162          case 9: numberOfDays = 30; break;
163          case 10: numberOfDays = 31; break;
164          case 11: numberOfDays = 30; break;
165          case 12: numberOfDays = 31; break;
166       }//end switch
167
168       return numberOfDays;
169
170    }//end getNumberOfDays
171
172    //Returns correct days in Feburary, depending on if its a leap year or not
173    private int getFebDays(){
174
175       int n = 28; //default, non-leap year days
176
177       //function to determine if its a leap year. If it is, increase n to 29 (29 days in Fe
b if leap year)
178       if(this.year % 4 == 0 && (this.year % 100 != 0 || this.year % 400 == 0)){
179          n++;
180       }
181
182       return n;
183
184    }//end getFebDays
185
186 }//end PrintCalendar class
```