	Computer Science Department/College of Engineering and Computer Science
	CSc 20: Programming Concepts and Methodology II
	Lab 3 – Java statements (Fall 2016)

Objective:

In this lab, you are to write a Java program to accept an integer, indicating a year, from the command line and then print the calendar for that year. If an integer is not available from the command line, your program will use the current year as input (no hardcode value please!). A sample output is given below.

Overview:

This lab's objective is to exercise with usages of Java's control statements. You are suggested to use exactly one while statement, one for statement and one switch statement. You will also practice on how to use some basic input methods of the Scanner class and some formatting techniques of method printf().

Activities:

1. Copy instructor's JulianDate class from SacCT into your working directory. The JulianDate class is used to determine the day of the week for the 1st day of January.

JulianDate JD = new JulianDate();

int date = JD.toJulian(yr,1,1);

int dayOfWeek = (date+1)%7; // 0 means Sunday, 1 means Monday, etc.

2. Develop your program according the pseudo code given in class. You are welcome to develop your program from afresh. However, if there is a need for hints, please refer to the pseudo code (PrintCalendar.java) provided in SacCT.

Requirements:

1. No arrays are allowed in this lab.
2. Your output should be closely similar to the output of the instructor's sample program (see below).
3. To determine whether a year is a leap year or not:

   a. If the year is a century year, the year must be divisible by 400.
   b. If the year is not a century year, the year only needs to be divisible by 4.

4. To determine the current year, please use this code.

```
currentYear = Calendar.getInstance().get(Calendar.YEAR);
```

(note: please import java.util.Calendar;)

5. Allow user to input the selected year using Jgrasp's "run argument" as an option, if an user chooses to do so.

6. To indicate the current month, if the input year matches the current year i.e. that is 2016 (no hardcode please), mark the current month (i.e September) with dashed-lines (top/bottom lines).   Otherwise, the marker lines should not be displayed.

Sample output: (Example: showed only January to May). In this example, we assume the current month is Febuary. However, the actual month is September as today.

```
2016

        January

 S  M Tu  W Th  F  S
               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

--------------------
        February

 S  M Tu  W Th  F  S
    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
--------------------

        March

 S  M Tu  W Th  F  S
       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


        April

 S  M Tu  W Th  F  S
               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


        May

 S  M Tu  W Th  F  S
 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
```