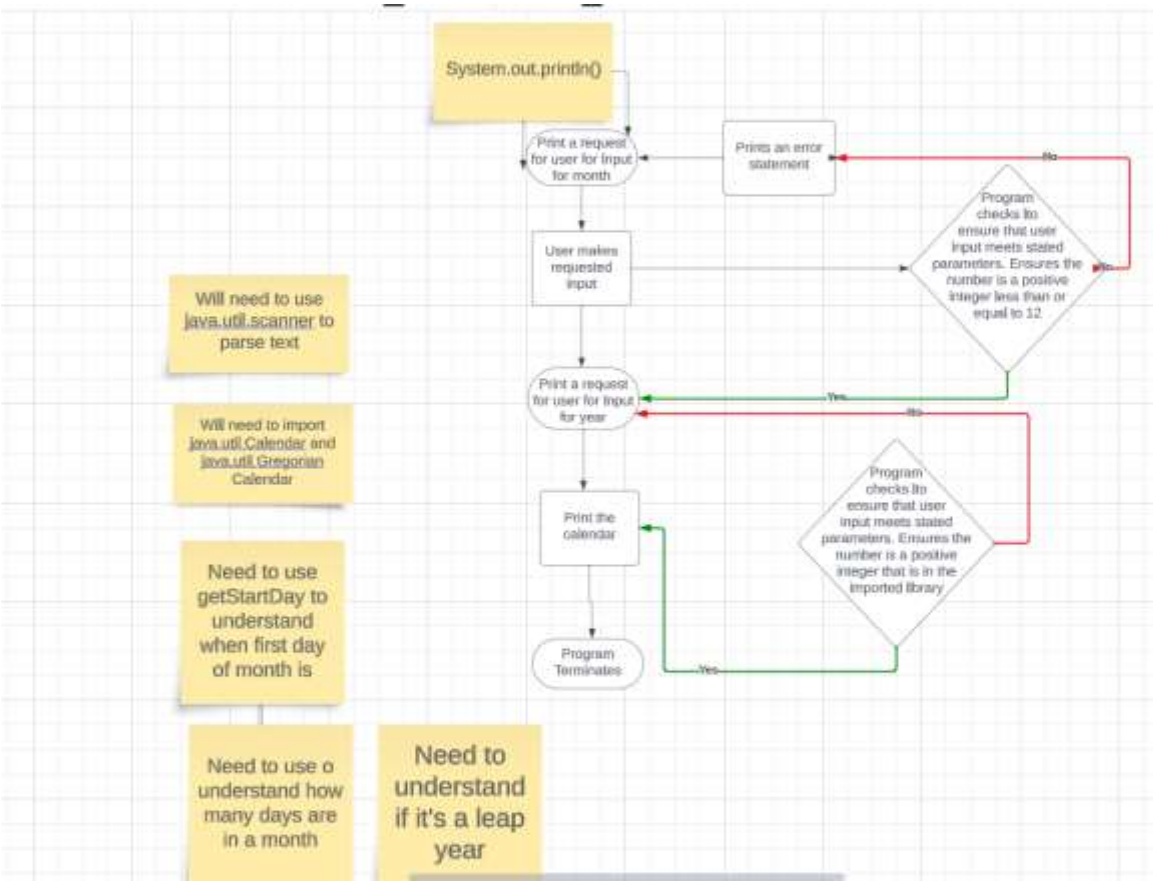
1. //Design


System.out.println()
Print a request for user for input for month
Prints an error statement
No
User makes requested input
Program checks to ensure that user input meets stated parameters. Ensures the number is a positive integer less than or equal to 12
No
Yes
Print a request for user for input for year
No
Print the calendar
Program checks to ensure that user input meets stated parameters. Ensures the number is a positive integer that is in the imported library
Yes
Program Terminates
Will need to use java.util.scanner to parse text
Will need to import java.util.Calendar and java.util.Gregorian Calendar
Need to use getStartDay to understand when first day of month is
Need to use o understand how many days are in a month
Need to understand if it's a leap year

//Code

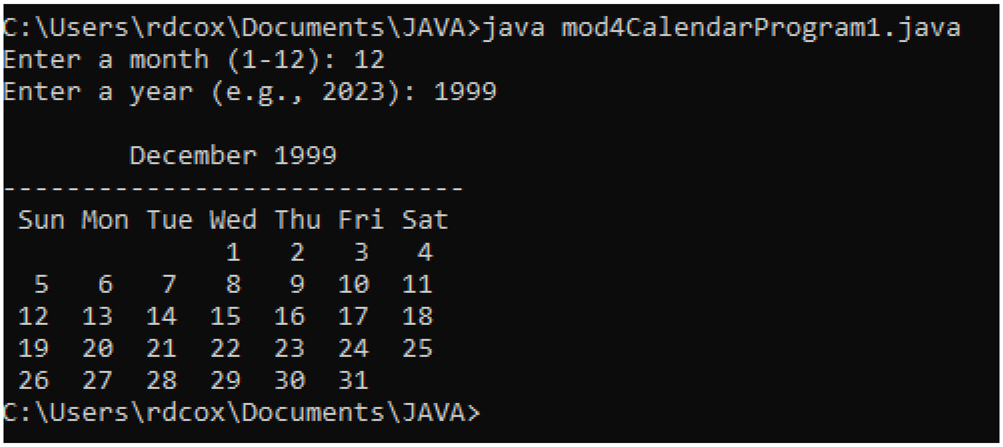
```
import java.util.Scanner;

import java.util.Calendar;

import java.util.GregorianCalendar;


public class mod4CalendarProgram1 {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);


        int month = getMonthFromUser(input);

        int year = getYearFromUser(input);
```

```
   printMonthCalendar(month, year);


   input.close();
}


private static int getMonthFromUser(Scanner input) {
   System.out.print("Enter a month (1-12): ");
   return input.nextInt();
}


private static int getYearFromUser(Scanner input) {
   System.out.print("Enter a year (e.g., 2023): ");
   return input.nextInt();
}


private static void printMonthCalendar(int month, int year) {
   printMonthHeader(month, year);
   printMonthBody(month, year);
}


private static void printMonthHeader(int month, int year) {
   System.out.println("\n" + "\t" + getMonthName(month) + " " + year);
        System.out.println("----------------------------");
   System.out.println(" "+"Sun Mon Tue Wed Thu Fri Sat");
}


private static void printMonthBody(int month, int year) {
        int startDay = getStartDay(month, year);
        int numDaysInMonth = getNumDaysInMonth(month, year);

        // Print leading spaces for the first week
        for (int i = 1; i < startDay; i++) {
            System.out.print("    ");
        }

        // Print the days of the month
        for (int day = 1; day <= numDaysInMonth; day++) {
            System.out.printf("%3d ", day);

            // Move to the next day
            startDay++;
            if (startDay > 7) {
                startDay = 1;
                System.out.println();
            }
        }
    }

    private static String getMonthName(int month) {
        String[] monthNames = {
            "January", "February", "March", "April", "May", "June",
            "July", "August", "September", "October", "November", "December"
        };
        return monthNames[month - 1];
    }
```

```
private static int getStartDay(int month, int year) {
   // Adjust month number & year to fit Zeller's numbering system
        if (month < 3) {
        month += 12;
        year -= 1;
        }
        int centuryYear = year % 100; // Calculate year within century
        int centuryTerm = year / 100; // Calculate century term
        int firstDayInMonth = 0; // Day number of first day in month 'm'

        firstDayInMonth = (1 + // to shift index 0 to the 1-7 return range
        (13 * (month + 1) / 5)
        + centuryYear +
        (centuryYear / 4) +
        (centuryTerm / 4) +
        (5 * centuryTerm)) % 7;

        // Convert Zeller's value to ISO value (1 = Mon, ... , 7 = Sun )
        int dayNum = ((firstDayInMonth + 5) % 7) + 1;
        return dayNum;
}

private static int getNumDaysInMonth(int month, int year) {
   Calendar calendar = new GregorianCalendar(year, month - 1, 1);
   return calendar.getActualMaximum(Calendar.DAY_OF_MONTH);
}
```

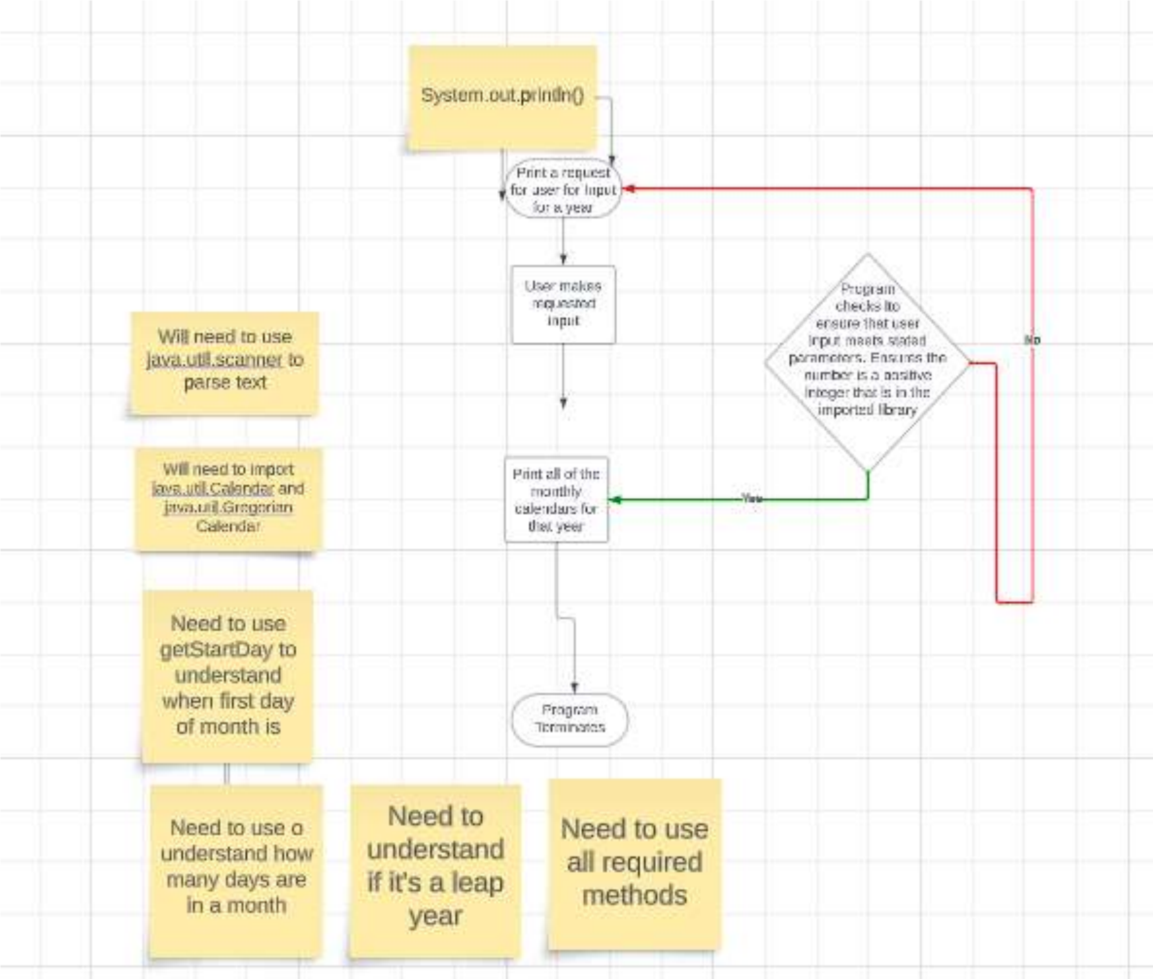
```
  private static boolean isLeapYear(int year) {

    return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);

  }//end main

}//end class
```

//output



```
C:\Users\rdcox\Documents\JAVA>java mod4CalendarProgram1.java
Enter a month (1-12): 12
Enter a year (e.g., 2023): 1999

        December 1999
-----------------------------
 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  31
C:\Users\rdcox\Documents\JAVA>
```

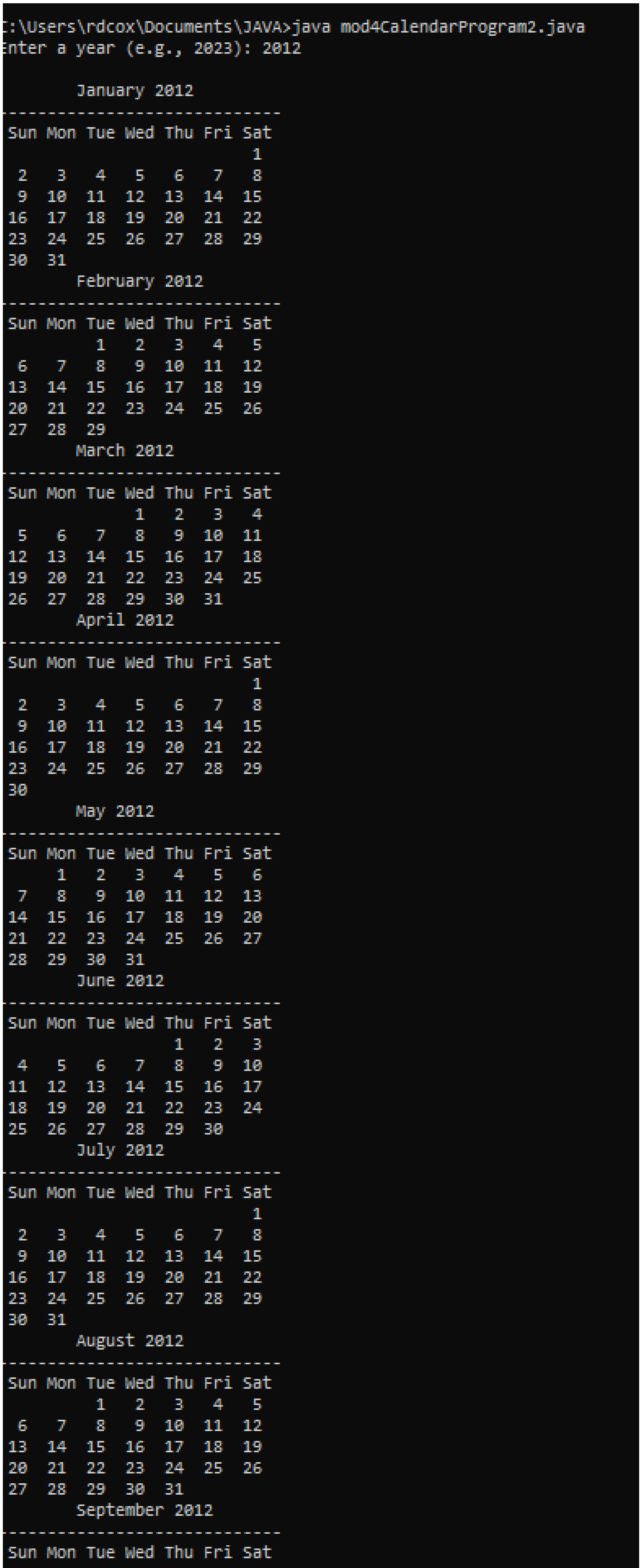
2. //Design


System.out.println()
Print a request for user for input for a year
User makes requested input
Program checks to ensure that user input meets stated parameters. Ensures the number is a positive integer that is in the imported library
No
Yes
Print all of the monthly calendars for that year
Program Terminates
Will need to use java.util.scanner to parse text
Will need to import java.util.Calendar and java.util.Gregorian Calendar
Need to use getStartDay to understand when first day of month is
Need to use o understand how many days are in a month
Need to understand if it's a leap year
Need to use all required methods

//Code

```
import java.util.Scanner;

import java.util.Calendar;

import java.util.GregorianCalendar;


public class mod4CalendarProgram2 {


    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a year (e.g., 2023): ");
        int year = scanner.nextInt();

        for (int month = 1; month <= 12; month++) {
            printMonthCalendar(month, year);
        }
    }

    private static void printMonthCalendar(int month, int year) {
        printMonthHeader(month, year);
        printMonthBody(month, year);
    }

    private static void printMonthHeader(int month, int year) {
        System.out.println("\n" + "\t" + getMonthName(month) + " " + year);
              System.out.println("----------------------------");
        System.out.println(" "+"Sun Mon Tue Wed Thu Fri Sat");
    }

   private static void printMonthBody(int month, int year) {
        int startDay = getStartDay(month, year);
        int numDaysInMonth = getNumDaysInMonth(month, year);

        // Print leading spaces for the first week
        for (int i = 1; i < startDay; i++) {
            System.out.print("    ");
        }
        // Print the days of the month
        for (int day = 1; day <= numDaysInMonth; day++) {
            System.out.printf("%3d ", day);

            // Move to the next day
            startDay++;
            if (startDay > 7) {
                startDay = 1;
                System.out.println();
            }
        }
    }

    private static String getMonthName(int month) {
        String[] monthNames = {
            "January", "February", "March", "April", "May", "June",
            "July", "August", "September", "October", "November", "December"
        };
        return monthNames[month - 1];
    }

    private static int getStartDay(int month, int year) {
        // Adjust month number & year to fit Zeller's numbering system
              if (month < 3) {
              month += 12;
              year -= 1;
              }
              int centuryYear = year % 100; // Calculate year within century
            int centuryTerm = year / 100; // Calculate century term
            int firstDayInMonth = 0; // Day number of first day in month 'm'

            firstDayInMonth = (1 + // to shift index 0 to the 1-7 return range
            (13 * (month + 1) / 5)
            + centuryYear +
            (centuryYear / 4) +
            (centuryTerm / 4) +
            (5 * centuryTerm)) % 7;

            // Convert Zeller's value to ISO value (1 = Mon, ... , 7 = Sun )
            int dayNum = ((firstDayInMonth + 5) % 7) + 1;
            return dayNum;
      }

      private static int getNumDaysInMonth(int month, int year) {
        Calendar calendar = new GregorianCalendar(year, month - 1, 1);
        return calendar.getActualMaximum(Calendar.DAY_OF_MONTH);
      }

      private static boolean isLeapYear(int year) {
        return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);
      }//end main
    }//end class
```

// output - continues on past september, but could not capture.



```
:\Users\rdcox\Documents\JAVA>java mod4CalendarProgram2.java
nter a year (e.g., 2023): 2012

        January 2012
-----------------------------
 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  31
        February 2012
-----------------------------
 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
        March 2012
-----------------------------
 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  31
        April 2012
-----------------------------
 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
        May 2012
-----------------------------
 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  31
        June 2012
-----------------------------
 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
        July 2012
-----------------------------
 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  31
        August 2012
-----------------------------
 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  31
        September 2012
-----------------------------
 Sun Mon Tue Wed Thu Fri Sat
```