Programming Exercises - SDJ - Session 10

Exercise 10.01

Write a program that asks the user to input an `int`, n, and then prints out the following:

a) All the numbers 1, 2, 3, …, n
b) All the numbers 2, 4, 6, …, 2*n
c) All the numbers 1, 4, 9, …, n^2

Exercise 10.02

Our current concept of leap years was started in 1582 when the Gregorian calendar was introduced. From previous exercises you should have a `MyDate` class with an `isLeapYear()` method. Use that for the following:

a) Create a program with a main method that uses a `for`-loop and a `MyDate` object to calculate how many leap years there have been since 1582, and then prints out the result. Hint: set the `MyDate` object to all the years, and check which ones are leap years.
b) Modify the program by adding a `Scanner` object, allowing the user to input a start year and an end year. Then the program should calculate the number of leap years within that range.

Exercise 10.03

Add another method to your `MyDate` class called `nextDay()`. The method should change the date to one day later than the current date. Then in your test class create a `MyDate` object with the date of your birthday, and use a loop and the `nextDay()` method to find out how many days it has been since you were born.

Exercise 10.04

The Fibonacci numbers is a special sequence of numbers. They are defined so that the first two numbers are 1, and all following numbers are the sum of the two preceding numbers. E.g.
The 0^{th} Fibonacci number is 1.
The 1^{st} Fibonacci number is also 1.
The 2^{nd} Fibonacci number is 1 + 1 = 2.
The 3^{rd} Fibonacci number is 1 + 2 = 3.
The 4^{th} Fibonacci number is 2 + 3 = 5.
Etc.

Write a program that prints out the first 20 Fibonacci numbers, in the following format:
Output:

```
Fibonacci(0) = 1
Fibonacci(1) = 1
Fibonacci(2) = 2
Fibonacci(3) = 3
Fibonacci(4) = 5
Fibonacci(5) = 8
Fibonacci(6) = 13
...
```

Exercise 10.05

Write a program that prompts the user for a string, and prints its reverse. E.g. the string `"Hello"` typed on the keyboard will be printed out as `"olleH"`. Keep doing this over and over, until the user enters the string `"quit"`.

Note that the method `charAt(index)` in the `String` class can be used to get the character at a specific index in the string, and the method `length()` can be used to get the number of characters in the string. When comparing strings, the method `equals()` might be useful.

Exercise 10.06

Guess a number part 2. Write a program that generates a random number between 1 and 1000, and then asks the user to try and guess it. Every time the user makes a wrong guess, display whether the guess was too high or too low, and ask the user to make another guess. When the user guesses the right number, display how many attempts were needed.