Computer Science 112: Lab 5 JTerm 2022
Due Friday, Jan 14
Inheritance
YOU WILL WORK WITH A PARTNER ON THE FIRST PART OF THIS LAB

Download the java files TestDate.java, DateFactory.java and GregorianDate.java from moodle. Do not download the others for now.

a. Compile and run TestDate.java. Then examine TestDate.java, DateFactory.java and GregorianDate.java Make sure that both of you understand the programs. (This has nothing to do with inheritance,)

b. Download SpainDate and ScotlandDate from moodle. SpainDate.java will not compile(neither will ScotlandDate.java), try to figure out why. If you cannot, please ask for an explanation.

c. Modify SpainDate and ScotlandDate so that they will compile, and so that their constructors will set the day, month, and year fields in Gregorian date correctly.

d. Discuss with your partner how to modify DateFactory so that createDate will return a GregorianDate, a ScotlandDate or a SpainData depending on the value of the parameter where. (No coding)

Discuss what will happen if you override (in SpainData or ScotlandDate) one or more of the methods isLeap, isLastday and isChangedForSpecialDay, but do NOT change makeTomorrow. For example, what is the effect of adding a version of isLeap in SpainData? (No coding)

Tell a teacher what you decided. If you are working in lab session DO NOT SKIP this checkin.

By the time you break up with your partner, you should understand how makeTomorrow works with the other methods.

Now work on our own. (You will need to fix SpainData and ScotlandDate in your account if you have not yet done so.)

e. Modify createDate so that it works as described above, and modify TestDate so that it generates test cases of each type of date. Be sure to chose dates that will test makeTomorrow for Scotland and Spain.

f. Modify SpainData and ScotlandDate so that makeTomorrow works correctly for all three. **Do not change or override** makeTomorrow. You will need to override other methods. There should not be any if statements depending on the location outside of DateFactory. You will need to add appropriate getter/setter methods to GregorianDate.

Use the following info, which is an expansion of what you did in Lab 2:

For Scotland:

- The day after September 2, 1752 was September 15, 1752

- For years before 1600, the last day of the year was March 24, so, for example, March 24, 1500 was followed by March 25, 1501. In 1600 and after, the last day of the year was December 31.

- Before 1752, but after 1599, all years divisible by 4 were leap years, before 1600, all years preceding years divisible by 4 were leap years. Since 1752, Scotland has used the modern definition.

For Spain:

- The day after Oct 4, 1582 was Oct 15, 1582
- Before 1556 the first day of the new year was 25 March. In 1556, it was moved to January 1.
- The leap year reform was done at the same time in 1582 as the days were omitted. That is, after that the leap years followed the current rule, before that, all years divisible by 4 were leap years.

Do not worry too much about what happens in the years near the New Year's day move.

Note: Many of you made errors in the implementation of the leap year algorithms for Spain and Scotland. You do NOT need to fix those errors in this lab. I will not take off for such errors in this lab. However, you must at least try to get it right.

If you are working from the command line, when you compile this lab, make sure that you compile like this:

```
javac *.java
```

If you do not, some of your classes may not be recompiled.

Submit this on moodle.