DAT-119 – Python 1

Spring 2019

**Homework 4**

**Don’t forget to plan before you code!**

1. Please put together a program that takes in a year from the user and tells them how many days February had in that year. (History buffs: let’s assume we’ve been using the Gregorian calendar forever, OK?)

While this is *always* true for *every* assignment (unless I explicitly say otherwise), this is *especially* true for this program: *you may* ***not*** *use Google or any other search engine on any step of this program*.

* There are 28 days in February, unless it is a leap year.
* If it is a leap year, there are 29 days in February. Leap year criteria, from [TimeAndDate.com](https://www.timeanddate.com/date/leapyear.html):
  + The year can be evenly divided by 4;
  + If the year can be evenly divided by 100, it is NOT a leap year, **unless**;
  + The year is also evenly divisible by 400. Then it is a leap year.

(There’s some subtlety to the logic, here. Map it out for yourself before you try coding it. This is where your planning document is GOLDEN. Run the test cases below through your logic before you try to code anything. Trust me.)

**Test cases:** the years 2000 and 2400 are leap years, while 1800, 1900, 2100, 2200, 2300 and 2500 are NOT leap years. 1924 was a leap year.

1. Let’s say you have a friend who plans a lot of events, at a lot of different venues. Write a script that asks the user for the number of people attending an event, the number of seats available at each table in the venue (assume all tables are the same size), and the maximum number of tables available at the venue.

The program should output “This party won’t fit at this venue” or something similar if the available tables won’t fit the party size.

*If the party will fit*, the program should output the number of tables that will be required to seat the party. It should also output how many spaces the smallest table would have open, if all of the other tables were filled.