Homework 3

COMSC-122

Fall 2017

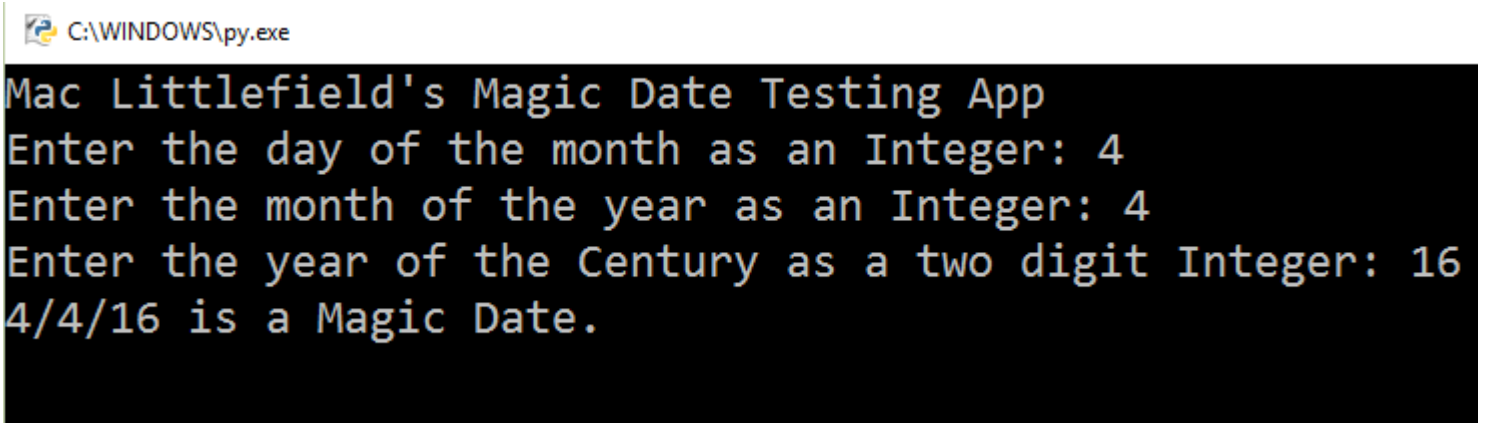
Homework 3: Magic Dates

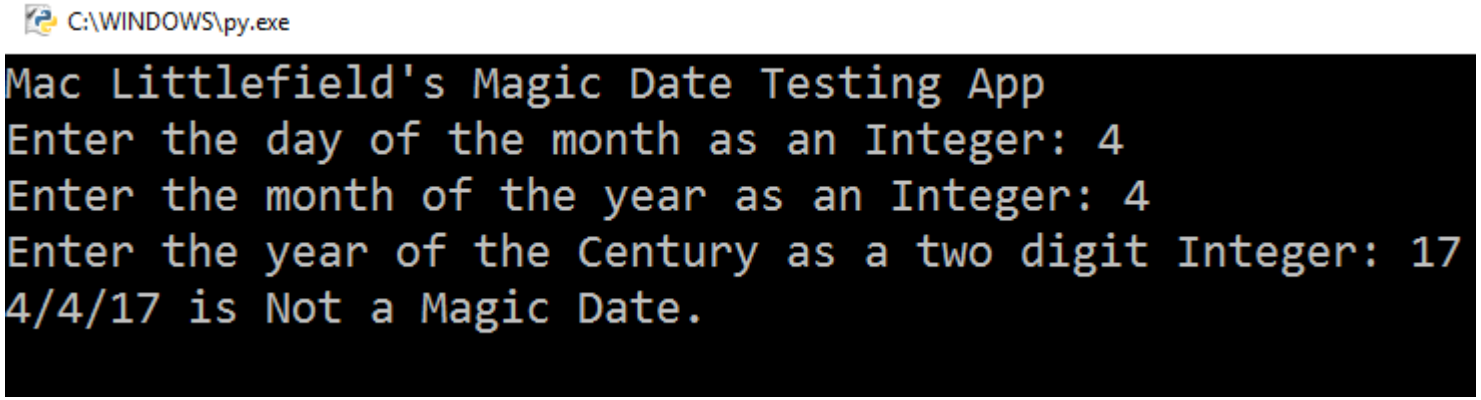
- The date June 10, 2060, is special because when it is written in the following format, the month times the day equals the year: 6/10/60 .
- Design a program (Homework-3A & Homework-3B that asks the user to enter (in numeric form): a month; a day; and a two-digit year.
- The program should then determine whether the month times the day equals the year.
  - If so, it should display a message saying the date is magic.
  - Otherwise, it should display a message saying the date is not magic.

Homework-3A: Magic Date Program

- Write a program that will test the date typed in to determine if it is a Magic Date or not.
- Call this program *YourName*-Homework3A.py
- Your programs should always have at least three comments:
  - Your Name
  - The program Name
  - What the program does.
- Be sure to end your program with the statement:
  - input(‘’)
  - This will hold the output on the screen when you run it from Windows.
- The first line of your screen output should contain your own name – see the following sample output.

Two Sample Outputs





Homework 3B: Valid Date Test

- Add to Homework-3A some additional code that will determine if the date that the user types in is indeed a valid date.
  - Your program should not even attempt to test to see if the date is a Magic Date if the date typed in is not a valid date.
  - If it is a valid date, print out that the date is valid, then continue checking to see if it is a Magic Date.
  - Remember the following rules with respect to dates:
    - 00 <= year <= 99
    - 01 <= month <= 12
    - All days must be: day >= 1
      - For months: 1, 3, 5, 7, 8, 10, 12, day <= 31
      - For months: 4, 6, 9, 11, day <=30
      - For month = 2:
        - If year is evenly divisible by 4, day <= 29
        - Otherwise, day <= 28
  - You must test for all these conditions to verify if a date is a valid date.
- Name your program: *YourName*Homework-3B.py
- Again, your name needs to appear on the output
  - See the following two examples.

Two Sample Outputs