2020-2 Python Programming

HW3

Due: Nov. 15 (Sun.) 23:50

1. Define a function to take a date and a date format as its parameters and return True if the formats are consistent.
   1. If the format of the given date and the given date format is consistent, return True. Ex) 2020-11-05 & YYYY-MM-DD 🡺 True, 11/05/2020 & MM/DD/YYYY 🡺 True
   2. Y presents a digit of year, M represents a digit of month, and D represents a digit of day. Other characters used are all for decorative purposes.
   3. Hint: take each character in the format string and compare it with the corresponding character in the date string. (Y, M, D 🡪 integer, others 🡪 same character)
   4. If the given date does not match with the date format, print an error message and return False.
2. Define a function to take a date and a date format as its parameters and return True if the date is a valid date.
   1. A month should be between 1 and 12.
   2. A day depends on the month. (an odd month – 31 days, an even month except February – 30 days, February – 28 days)
   3. Hint: in order to get the value for month, find the position of ‘M’ in the format string and take 2 characters from the position and plus one in the date string. Finding the value for day is same.
   4. If the given date is not a valid date, print an error message and return False.
3. Define a function to take a date and return its weekday name. (ex. Monday, Tuesday, …)
4. Ask the user to choose a date format. There are 2 options: YYYY-MM-DD, MM/DD/YYYY. Before asking, show these options with the order number. The user must choose one of them by its order number. Keep repeating this procedure if the input number is not valid.
5. Receive a date from the user.
   1. Check if the format is consistent with the choice made at Q4 using the function defined in Q1. If not, keep repeating this procedure.
   2. Check if the date is valid using the function defined in Q2. If not, keep repeating this procedure.
6. Print what weekday the given day is using the function defined in Q3.
7. Print what weekday it will be, a year after the given day, using the function defined in Q3. Print the date, too.
8. Ask the user if he/she wants to test another date. If y, repeat Q4~Q7.
9. If not y, print 'The random auto-testing is starting...'
10. Define a function to return a randomly composed date. It has no parameter.
    1. The year must be 4 digits between 2000 and 2020 (both inclusive), the month is 2 digits between 01 and 19 (both inclusive), and the day is 2 digits between 01 and 49 (both inclusive).
    2. Hints:
       1. First choose a format randomly: 1 or 2.
       2. If the chosen format is 1, add up a 4-digits random number for year + ‘-‘ + a 2-digits random number for month + ‘-‘ + a 2-digits random number for day
       3. If the chosen format is 2, add up a 2-digits random number for month + ‘/ + a 2-digits random number for day + ‘/ + a 4-digits random number for year
       4. If the random value for month or day is just a single digit (ex, 3), add ‘0’ before it in order to make it a 2 digit value (ex. 03).
11. Compose 10 random dates using the function defined in Q10. (ex. 2020-11-49, 12/25/1998, 03/02/2001, 2011-15-31, ..... ). For each random date,
    1. Randomly choose a date format to test the format consistency.
    2. Print the date generated by Q10 function and the test format chosen at A.
    3. Check their format consistency using Q1 function. Print the result.
    4. Check the validity of the random date using Q2 function. Print the result.
    5. If the date is a valid date, print its weekday.

