CS417 Lab #10

Getting Started

Create a folder for your work, then download this file into it:

- `dates.py`

Goals

In this lab, you will learn how to

- create `datetime.date` objects,
- work with day offsets (`datetime.timedelta`),
- work with day-of-the-week information, and
- handle leap years

Tasks

Edit the file `dates.py`, and implement these functions:

1. `text_to_ymd(text)`: The `text` will be a string like `"2019-12-25"`, or like `"1/25/2019"`. You should split the string into three fields (three `str` variables), convert them all to `int` (three `int` variables), and then return the three `int`s.

2. `new_years_day()`: Get today's date with `datetime.date.today()`, get its `.year` part, and use that to return the first date of *this* year.

3. `n_weeks_after_newyears(n_weeks)`: Get this year's first date, and add `n_weeks` worth of days to it. You can't add `int`s to `date`s. You must add `datetime.timedelta(X)` where `X` is the number of days.

4. `first_thursday(month)`: obtain the first Thursday of the given month, in the present year. Get the value of `datetime.date(year, month, 1)`, and check its `.weekday()`. If it's not 3 (the 4th day of the week), keep adding one `datetime.timedelta(1)` to it.

5. `thanksgiving_thursday()`: get the first Thursday of the 11th month of this year (call `first_thursday(11)`), and add 14 days to it.

6. `days_until_thanksgiving()`: subtract today's date from thanksgiving's date. This gives you a `datetime.timedelta` object. Return its `.days` field.

7. `election_day(year)`: Election day is the first Tuesday after the first Monday in November. You can figure this out.

8. `is_leap_year(year)`: Return `True/False` if the given year is/isn't a leap year. In our (Gregorian) calenday, a year has a February 29th (is a leap year) if it is divisible by 4, except that years divisible by 100 are *not* leap years, except that years divisible by 400 *are* leap years (1900 was not a leap year, but 2000 *was* a leap year).

Turn in your work

To turn you work in, go to mycourses.unh.edu, find CS417 and the lab, click the "Submit" button, and upload `dates.py` . At the end of the lab session, submit any work you have completed. You can submit again until midnight, with no lateness penalty.