**Week 12 In-Class Exercises (Extra)**

## Q1: To-Do List [ \*\*\* ]

**Part (A)**

Write a program that allows a user to enter his/her tasks. The program asks for a task as well as the due date (which contains a month and a date). If the user enters an invalid month or invalid date, the program has to re-prompt the user until the information is valid. (A valid month is an integer between 1 and 12. A valid date is an integer between 1 and the maximum number of days of that month. E.g., if month is 10, then the date has to be between 1 and 31. If month is 4, then the date has to be between 1 and 30. For February, the date has to be between 1 and 29.) The program keeps prompting the user for more tasks and their due dates until the user chooses to stop. The program then displays the list of all tasks, grouped by their due dates.

You can assume that all the tasks are due in the same year. Therefore for all due dates, the year is omitted.

A sample run of the program is shown below. Texts shown in bold are user inputs.

|  |
| --- |
| Do you want to enter a task? [Y|N]:**Y**  Enter task description:**IS111 Lab2** Enter the month it is due:**13** Sorry, invalid month!  Enter the month it is due:**abc** Sorry, invalid month!  Enter the month it is due:**11**  Enter the date it is due:**31** Sorry, invalid date!  Enter the date it is due:**$$$** Sorry, invalid date!  Enter the date it is due:**3**  IS111 Lab 2 due on 3/11 has been recorded.  Do you want to continue? [Y|N]:**Y**  Enter task description:**Top up EZ-link card** Enter the month it is due:**10** Enter the date it is due:**10**  top up EZ-link card due on 10/10 has been recorded.  Do you want to continue? [Y|N]:**Y**  Enter task description: **ECON101 HW#3** Enter the month it is due:**11** Enter the date it is due:**3**  ECON101 HW#3 due on 3/11 has been recorded.  Do you want to continue? [Y|N]:**N**  Here are the tasks you've entered:  Due on 3/11:  IS111 Lab 2  ECON101 HW#3  Due on 10/10:  top up EZ-link card |

**Part (B)**

Improve the program above such that when listing all the tasks entered, they’re ordered by the due dates in chronological order, i.e., the tasks that are due earlier are listed earlier.

For example, given the tasks shown in the sample run above, the program should display the tasks as follows in the end:

|  |
| --- |
| Here are the tasks you've entered:  Due on 10/10:  top up EZ-link card  Due on 3/11:  IS111 Lab 2  ECON101 HW#3 |

**Part (C)** (Out of Scope)

Further improve the program such that if a due date is in the past, it is not accepted. For example, assuming that today’s date is November 10. The code below shows the expected error messages:

|  |
| --- |
| ...  Enter task description:**IS111 Lab2** Enter the month it is due:**9** Sorry, the month has passed.  Enter the month it is due:**11** Enter the date it is due:**9** Sorry, the date has passed.  Enter the date it is due:**12**  ... |

To get today’s date, you can use the following code:

|  |
| --- |
| import datetime  date\_of\_today = str(datetime.date.today()) |

The code above gives you the date in 'YYYY-MM-DD' format.