Project 1 of CSC 295-01 (Spring 2017)

Dr. Liudong Zuo (lzuo@csudh.edu)

Deadline is March. 31, 2017.

Assignment Instruction:

1. Please only create one project and then create one package for each problem. For example, suppose we have three problems here, then create three packages. Give each package a meaningful name, such as problem1, problem2, etc., then put all java files of one problem inside the package of that problem.
2. Export the entire project and only submit one single zip file. Submitting multiple .java file will delay the grading and you will be asked to resubmit your programs. Please refer to the following page for how to export a project: `http://agile.csc.ncsu.edu/SEMaterials/tutorials/import_export/`

Problem 1 Calendar Design (50 points)

Write an application that accepts a year and displays a 12-month planning calendar. Each month should be printed separately, one below the next. For example, Fig. 1 shows the January and February of 2007. Note that using existing Java Calendar class is prohibited.

Hint:

- Please refer to the program in the following webpage to determine the day of the week for any date (month/day/year).

  `http://helpdesk.objects.com.au/java/zellers-congruence-in-java`

- You need to consider how many days February has (28 or 29). Do some research and see how to determine the number of days February has in one specific year.

Problem 2 Shopping Checkout Simulator (50 points)

Design a shopping checkout simulator as shown in Fig. 2

January 2007

Sun	Mon	Tues	Wed	Thurs	Fri	Sat
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27

February 2007

Sun	Mon	Tues	Wed	Thurs	Fri	Sat
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28			

etc.

…

Figure 1: January and February of 2007.

For the first window, please add five more items. After user clicks the “Calculate total” button, the total should be displayed. After user clicks the reset button, all user inputs including the total will be reset. After use clicks the “Checkout” button, we go to next check out window.

User inputs their payment information in the second window. For the state drop-down list, you only need to add several, for example, 5 states are enough. User’s visa number should not be directly shown. Use Password Field for the Visa number to hide user’s visa number like the picture shows. An order number should be generated. After user clicks submit, we store user’s order information in a list or in a text file, and close current window. After user clicks the search button, we go to next window.

The third window allows us to search the order using either the name or the order number. If we do not have a match, an error dialog box should pop up; if we have a match, the search result would be displayed in the “search result” area. After user clicks the “Delete” button, current order information is deleted and current window closes. After user clicks the “Modify” button, another window (last picture) pops up and the order information is

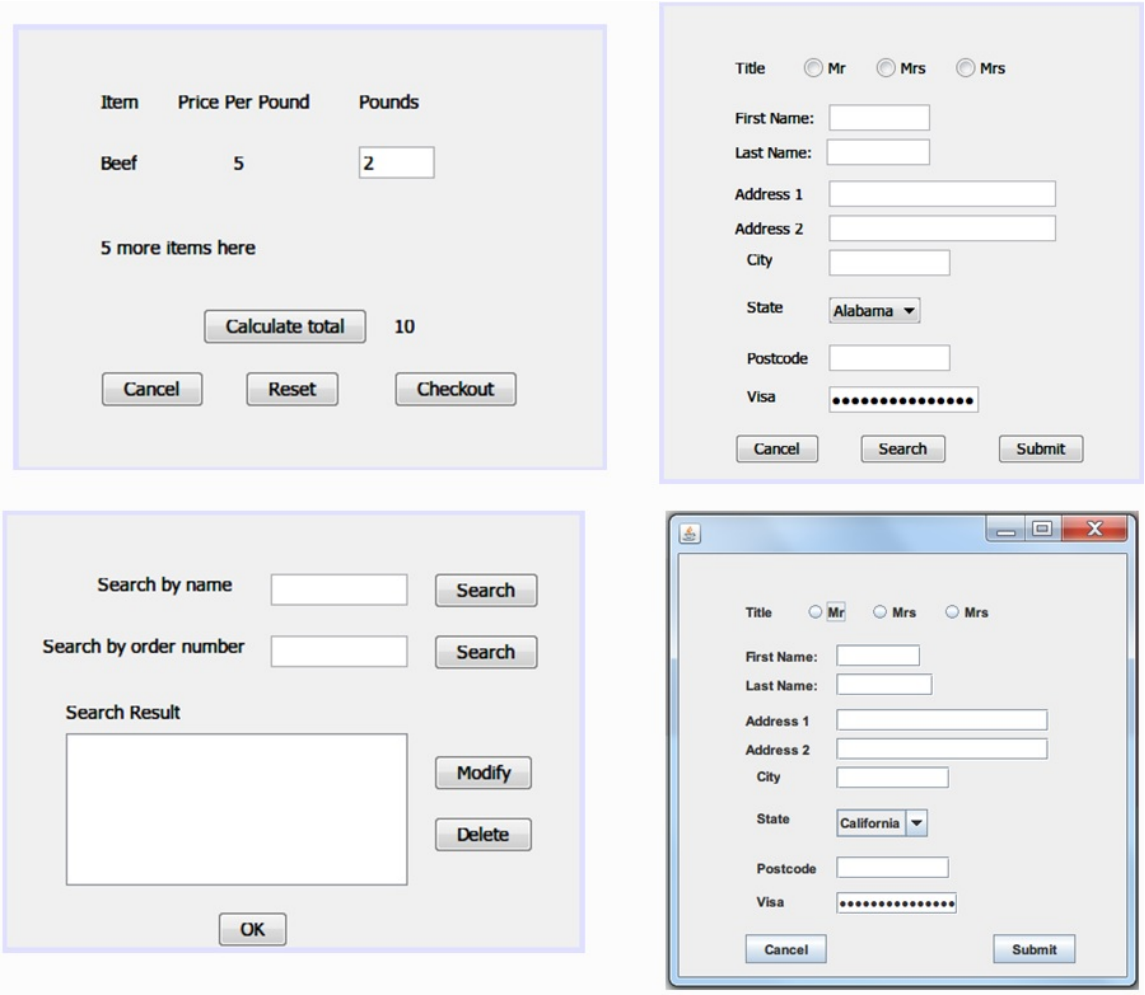

Figure 2: Shopping Checkout Simulator.

automatically filled. For example, if we have an order shown in the picture, all of the information is automatically filled in the window when the window pops up. User can only modify the information he/she would like to change. After clicking the “Submit” button, we update the order information and current window closes.