CSC116: Comprehensive Exercise - Client Ideas

Created by Dr. Jessica Schmidt

A Calendar-Making Program

Create an application that accepts a year and displays a 12-month planning calendar. Each month should be displayed separately. For example, for 2007, in a text format:

```
            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
28    29    30    31
            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.
```

…

Note that you may choose to use a graphic format.

Allow a user to specify any number of dates to be noted with the month (birthdays, anniversaries, and so on). For example, a user should be able to request that January 8 be displayed with the note: “Elvis’s Birthday,” or February 18, “Take Dog to Groomer,” or December 25, “Christmas.” A list of annotated dates should appear with that month’s calendar. For example, the first month of the annotated calendar might look like this:

```
            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
28    29    30    31

January 1: New Year’s Day
January 8: Elvis’ Birthday; visit Graceland
January 23: Get Fifi a trim at the Pet Central
January 31: Phantom of the Opera
```

Extension

Allow multiple events to occur on a single date. If multiple events occur on the same date, then the events should be given in a bulleted list (or graphical equivalent):

```
            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
28    29    30    31

January 1: New Year’s Day
January 8:
 - Elvis’ Birthday; visit Graceland
 - Brandon’s Birthday
 - Lance’s Birthday
January 23: Get Fifi a trim at the Pet Central
January 31: Phantom of the Opera
```

A Daily Planner

To manage my schedule, I need to keep track of day-to-day events. An event might be an appointment, an errand, a reminder, or whatever I need to remember.

Create a planner application to help me manage my daily events. Each event may include a date, time (0– 23, military style), description, and location.

For example, an event might look like this (if I a textual interface is implemented):

```
11   16   1959
Sister’s Birthday
```

or this:

```
12   25   2008  18
Christmas Dinner at Grandma’s
```

When I enter an event, the application should check that the time of the event does not conflict with another event. The planner, if queried, should be able to list all events for a particular date or range of dates. The planner should read events from a file when the program starts and write the new list of events to a new file when the program ends.

Extension

Allow multiple events to occur on a single date as long as the times do not overlap, which means each event with time should have an end time or duration. If multiple events occur on the same date, then the events should be given in a bulleted list format (or graphical equivalent):

```
11   26   2015
 - All Day: Thanksgiving Day
 - 9: Gobble Wobble
 - 13: Thanksgiving Lunch at Grandma’s
 - 18: Thanksgiving Dinner at Aunt’s
```

Bunco

Create an application of the game, Bunco. Bunco is a family-friendly dice game that requires no skill and is all luck. The two-player game is played with three dice and consists of six rounds. Each round will have a target value—e.g., first round has target value of one, second round has target value of two. During a round, all players take turns rolling the dice until one of the players has 21 points. A player’s turn consists of rolling the three dice together. For each dice that is the target value, the player gets one point added to his/her round score. An exception to getting one point for each dice that is the target value is if all three dice are the target value; this is a Big Bunco and is worth five points. A player’s turn continues until he/she rolls the dice and none of the dice are the target value. In addition to getting points when rolling the target value, if all three dice have the same value (that is not the target value) then this is a Little Bunco, which is worth three points but ends a player’s turn. Each player will have a name, total score, current round score, number of rounds won, number of big buncos, and number of little buncos. The scores and counts for each player should be output at the end of the game.

Extension

I want to be able to play Bunco with more friends (game with 2-10 players). I also want to be able to use dice other than the typically six-sided dice. I want to be able to use any dice that has between (inclusive) six and twenty-six sides labelled with numbers. With the change in dice type, I would like the number of rounds to be based on the number of sides on the dice used—one round for each side of the dice being used.

Connect Four

I loved playing the two-player game Connect Four as a child and want you to write a Java implementation of the game. With the game, there is an 8x8 vertical grid on which players take turns placing their pieces. The goal of the game is to have four of your game pieces connected---in a row, column, or diagonally. When a player places a piece in a column, the piece will land in the bottom-most available row (see example below).
In the following 4x4 grid, if I were to add a piece to the first column, it would land in the bottom row since nothing else is in the column. If I were to add a piece to the second column, it would land in the second row on top of the ‘O’ since it is the bottom-most available row.

	O		
	X		

As the players play the game, they should be updated with how many pieces they have placed and the max number of connected game pieces they currently have (this will be less than four until the game is won).

Extension

I want to improve the game by enlarging the grid and number of pieces that must be connected. The user should be able to set the number of pieces that must be connected (greater than four). The grid size should adjust to the number of required pieces such that it is always double the number of required pieces. For example, if required number of pieces connected is 6, then the grid is 12x12. Stats for players should be kept such that the two players can play multiple games and they can see how many games each player has won.