

IST 256
Lab Week 3 – Monday, September 12, 2011

A. Lab Exercise 1. Writing a GUI program – Each student should do their own program for this exercise, led by the instructor.

1. Login to lab computers

2. Open NetBeans and create a new project

Go to the Program menu and find NetBeans. Start a new project in NetBeans and create a Java GUI Application called PartyPlanner. If you need to look at the details, refer back to the instructions from last week. But remember the three main steps:

- create the new project,
- create the new JFrameForm, name it something like PartyPlannerUI, and put it into the package for partyplanner and
- set the Properties under Run to have the UI as the main class.

3. The program specification (statement of problem)

This program should allow the user to type in the number of people coming to a party. If they want to find out how many pizzas to order, they can type “yes” to request pizza. Then when they click on the request button, they will get a recommendation for the number of pizzas to order. (If they don’t request pizza, the number of pizzas will be 0.) There should also be a button to clear the form so that they can start their request over. The number of pizzas should be enough so that every person gets at least $\frac{1}{4}$ of a pizza.

4. Create the Form Design.

- **Add a label to the top of the form** by dragging it from the Palette pane (upper right hand side).
 - Edit the text to have a title such as “Party Planner”
- **Add another label to the form** and edit the text to say something like “Type number of people”
- **Add a textfield to the form** by dragging it from the Palette pane.
 - Change the text property to be 0 (and hit enter). Then resize the textfield on the form with your mouse.
- **Add another label to the form** and edit the text to say something like “Pizza? (type yes or no)”. (Note that in the future, we’ll use checkboxes.)
- **Add another textfield** beside this label and make it be initially blank.
- **Add a button to the form** by dragging it from the Palette pane.
 - Change the text property of the button to be an instruction to the user, e.g. Request Recommendation.
- **Add another label** to the form below the button to show the result of the request.
- **Add another button** below that and edit the text to say “Clear”

- **To add a program to be executed** for each button, right click on each button on your form and select Event -> Action -> actionPerformed.

- **Note that there will now be two button ActionPerformed methods in your code.**

5. Add the Program to each Button

Go to the Source tab and find the actionPerformed method for the Request Button and add a comment before it. My comment (and I also put a comment at the top of the program.)

```
/*  
 * Gets the number of people and whether to order food items from the form  
 * Displays the recommended number of food items  
*/
```

Add code right after the comment with TODO:

For each step, add the comment and then add the line(s) of code to perform the computation described.

Step 1: // declare variables for number of people and pizzas and initialize to zero
add declarations with initialization for variables called numpizza and numpeople

Step 2: // declare a variable for whether to order pizza, value will be “yes” or “no”

Suggested code:

```
String pizzaflag;
```

Step 3: // get the number of people and the desired food from the form
Add the code to set the values of numpeople and pizzaflag from the textfields.

Step 4: // if pizza requested, calculate recommended number of pizzas
write an if statement that tests if the pizzaflag is equal to “yes” and put in the body of the if test **one of the two following ways** to compute numpizzas. Both of these give the number of pizzas as the number of people divided by 4 if it divides evenly, but gives the next bigger integer if it doesn't.

Alternative 1:

```
if ((numpeople % 4) == 0) // numpeople divides evenly by 4  
{  
    numpizzas = (numpeople / 4); // integer division  
}  
else  
{  
    numpizzas = (numpeople / 4) + 1;  
}
```

Alternative 2:

```
double pizzaAmount;  
// double division gives amount with decimal fraction  
pizzaAmount = numpeople / 4.0;  
// use the ceil (ceiling) function to go up to next number and convert to int  
numpizzas = (int) Math.ceil(pizzaAmount);
```

Step 5: // display the recommended number of food items
jLabel4.setText("Number of pizzas to order is " + numpizzas);

(Check the number of the Label and if yours is a different number, change the last statement.)

- **Find the actionPerformed method for the Clear Button and add a comment before it that says that this will clear the TextFields.**
 - Add the code right after the TODO comment:
jTextField1.setText("0");
jTextField2.setText("");
- **Run this program** by the little green triangle on the top bar.
 - When the program executes, it should bring up the user interface form, and you should type in an integer into the text field. Then click on the button to get the number of pizzas.
- **Test your program with at least 3 trial runs:**
 - **Write down what number of people and pizza here. Test different numbers and at two “yes” and one “no” Do the results look plausible?**

Test Run	People	Pizza?	NumPizzas
1			
2			
3			

B. Lab Exercise 2 – Modifying the PartyPlanner program – work on your own:

1. Program Specification

Change the PartyPlanner program so that it also allows the user to also type in whether they want to order Wings for their party. The number of wings per person is 5, but if pizza was also ordered, then the numbers of wings per person is 3.

2. Change the form:

Add another label right under the Pizza? label to give a similar label for Wings? and a third textfield beside it that starts as an empty textfield.

Add another label under the requested number of pizzas label to get the requested number of wings also.

3. Add code to your program.

Go to the **actionPerformed** function that you wrote before for the **Request** button.

1. Add a third variable declaration for a variable of type `int` that will have the number of wings and initialize this to 0.
2. Add a second variable declaration for a variable of type `String` to be the wings flag.
3. Get the wingsflag from the third Textfield (`(jTextField3`)
4. After the IF statement to compute the number of pizzas, add another IF statement to compute the number of wings, using the description and using either a nested IF statement or a condition with `&&`.
5. Add another statement to display the number of wings in the last label on the form. (Check the number of the label to get the right one.)

Go to the **actionPerformed** function that you wrote before for the **Clear** button.

Add a statement to clear the third TextField.

- **Run this program** by the little green triangle on the top bar.
- **Test your program:**
 - **Perform at least 4 tests, designing the input to have different numbers of people and requests for pizza and wings.**

Test Run	Number People	Pizza?	Wings?	NumPizza	NumWings
1					
2					
3					
4					

Hand in this lab sheet by the beginning of class on Monday, September 19. Copy and paste the code from the Request button to a document that can be printed, and attach that code to this lab.