Chapter 4 Lab
Selection Control Structures

Lab Objectives

- Be able to construct boolean expressions to evaluate a given condition
- Be able to compare Strings
- Be able to use a flag
- Be able to construct if and if-else-if statements to perform a specific task
- Be able to construct a switch statement
- Be able to format numbers

Introduction

Up to this point, all the programs you have had a sequential control structure. This means that all statements are executed in order, one after another. Sometimes we need to let the computer make decisions, based on the data. A selection control structure allows the computer to select which statement to execute.

In order to have the computer make a decision, it needs to do a comparison. So we will work with writing boolean expressions. Boolean expressions use relational operators and logical operators to create a condition that can be evaluated as true or false.

Once we have a condition, we can conditionally execute statements. This means that there are statements in the program that may or may not be executed, depending on the condition.

We can also chain conditional statements together to allow the computer to choose from several courses of action. We will explore this using nested if-else statements as well as a switch statement.

In this lab, we will be editing a pizza ordering program. It creates a Pizza object to the specifications that the user desires. It walks the user through ordering, giving the user choices, which the program then uses to decide how to make the pizza and how much the cost of the pizza will be. The user will also receive a $2.00 discount if his/her name is Mike or Diane.

Task #1 The if Statement, Comparing Strings, and Flags

1. Copy the files Pizza.java (see code listing 4.1) and PizzaOrder.java (see code listing 4.2) from the Student CD or as directed by your instructor. Make sure to place them both in the same directory.
2. Pizza.java is correct, so you will not be editing this file. You only need to compile it. Compile and run PizzaOrder.java. You will be able to make

selections, but at this point, you will always get a 12 inch Hand-tossed pizza no matter what you select, but you will be able to choose toppings. You will also notice that the output does not look like money. So we need to edit PizzaOrder.java to complete the program so that it works correctly.

3. Construct a simple if statement. The condition will compare the String input by the user as his/her first name with the first names of the owners, Mike and Diane. Be sure that the comparison is not case sensitive.
4. If the user has either first name, set the discount flag to true.

Task #2 The if-else-if Statement

1. Write an if-else-if statement that lets the computer choose which statements to execute by the user input size (10, 12, 14, or 16). For each option, two statements need to be executed:
   a) A call to the setSize method passing in the size indicated.
   b) A call to the setCost method passing in the appropriate adjustment. Notice that in the Pizza.java program, the constructor creates a 12 inch Hand-tossed pizza for $12.99. The setCost method **adjusts** the cost, so a 10 inch pizza will need its cost decreased by 2, while the 16 inch pizza cost will need to increase by 4.
2. The default else of the above if-else-if statement should print a statement that the user input was not one of the choices, so a 12 inch pizza will be made.
3. Compile, debug, and run. You should now be able to get correct output for size and price (it will still have Hand-tossed crust, the output won't look like money, and no discount will be applied yet). Run your program multiple times ordering a 10, 12, 14, 16, and 17 inch pizza.

Task #3 Switch Statement

1. Write a switch statement that compares the user's choice with the appropriate characters (make sure that both capital letters and small letters will work).
2. Each case will call the setCrust method passing in the appropriate String indicating crust type.
3. The default case will print a statement that the user input was not one of the choices, so a Hand-tossed crust will be made.
4. Compile, debug, and run. You should now be able to get crust types other than Hand-tossed. Run your program multiple times to make sure all cases of the switch statement operate correctly.

Task #4 Using a Flag as a Condition

1. Write an if statement that uses the flag as the condition. Remember that the flag is a Boolean variable, therefore is true or false. It does not have to be compared to anything.

2. The body of the if statement should contain two statements:
   a) A statement that prints a message indicating that the user is eligible for a $2.00 discount.
   b) A statement that reduces the variable cost by 2.
3. Compile, debug, and run. Test your program using the owners' names (both capitalized and not) as well as a different name. The discount should be correctly at this time.

Task #5 Formatting Numbers

1. Add an import statement to use the DecimalFormat class as indicated above the class declaration.
2. Create a DecimalFormat object that always shows 2 decimal places.
3. Edit the appropriate lines in the main method so that any monetary output has 2 decimal places.
4. Compile, debug, and run. Your output should be completely correct at this time, and numeric output should look like money.

Code Listing 4.1 (Pizza.java)

```
//This program will represent a pizza

public class Pizza
{
	private double cost;				//the cost of the pizza
	private String crust;			//the type of crust
	private int size;					//the diameter in inches
	private int numToppings;		//the number of toppings
	private String toppingList;	//a list of the toppings

	//Constructor creates a 12" Hand-tossed pizza
	public Pizza()
	{
		cost = 12.99;
		crust = "Hand-tossed";
		size = 12;
		numToppings = 0;
		toppingList = null;
	}

	//adds the parameter amount to the cost
	public void setCost (double amount)
	{
		cost += amount;
	}
```

```
//sets the crust type
public void setCrust (String type)
{
	crust = type;
}

//changes the size of the pizza to the parameter diameter
public void setSize (int diameter)
{
	size = diameter;
}

//sets the number of toppings to the parameter number
public void setNumToppings(int number)
{
	numToppings = number;
}

//sets the list of toppings
public void setToppingList (String newTopping)
{
	toppingList = newTopping;
}

//returns the cost of the pizza
public double getCost()
{
	return cost;
}

//returns the crust type
public String getCrust()
{
	return crust;
}

//returns the size of the pizza
public int getSize()
{
	return size;
}

//returns the number of toppings
public int getNumToppings()
{
```

```
        return numToppings;
    }

    //returns the list of toppings
    public String getToppingList()
    {
        return toppingList;
    }
}
```

Code Listing 4.2 (PizzaOrder.java)

```
//This program allows the user to order a pizza

import java.util.Scanner;
//TASK #5 add an import statement to use the DecimalFormat class

public class PizzaOrder
{
    public static void main (String [] args)
    {
        //TASK #5 Create a DecimalFormat object with 2 decimal places

        //Create a Scanner object to read input
        Scanner keyboard = new Scanner (System.in);

        //Create an instance of a Pizza
        Pizza order = new Pizza ();

        String firstName;           //user's first name
        boolean discount = false;//flag, true if user is eligible for discount
        int inches;                 //size of the pizza
        char crustType;             //type of crust
        double cost;                //cost of the pizza
        final double TAX_RATE = .08;//sales tax rate
        double tax;                 //amount of tax
        char choice;                //user's choice
        String input;               //user input
        String toppings = "Cheese ";//list of toppings
        int numberOfToppings = 0;//number of toppings

        //prompt user and get first name
        System.out.println("Welcome to Mike and Diane's Pizza");
        System.out.print("Enter your first name:  ");
        firstName = keyboard.nextLine();
```

```
//determine if user is eligible for discount by
//having the same first name as one of the owners
//ADD LINES HERE FOR TASK #1

//prompt user and get pizza size choice
System.out.println("Pizza Size (inches)   Cost");
System.out.println("       10          $10.99");
System.out.println("       12          $12.99");
System.out.println("       14          $14.99");
System.out.println("       16          $16.99");
System.out.println("What size pizza would you like?");
System.out.print("10, 12, 14, or 16 (enter the number only): ");
inches = keyboard.nextInt();

//set price and size of pizza ordered
//ADD LINES HER FOR TASK #2

//consume the remaining newline character
keyboard.nextLine();

//prompt user and get crust choice
System.out.println("What type of crust do you want? ");
System.out.print("(H)Hand-tossed, (T) Thin-crust, or " +
        "(D) Deep-dish (enter H, T, or D): ");
input = keyboard.nextLine();
crustType = input.charAt(0);

//set user's crust choice on pizza ordered
//ADD LINES FOR TASK #3

//prompt user and get topping choices one at a time
System.out.println("All pizzas come with cheese.");
System.out.println("Additional toppings are $1.25 each,"
                + " choose from");
System.out.println("Pepperoni, Sausage, Onion, Mushroom");

//if topping is desired,
//add to topping list and number of toppings
System.out.print("Do you want Pepperoni?  (Y/N):  ");
input = keyboard.nextLine();
choice = input.charAt(0);
if (choice == 'Y' || choice == 'y')
{
        numberOfToppings += 1;
        toppings = toppings + "Pepperoni ";
```

```
}
System.out.print("Do you want Sausage?  (Y/N):  ");
input = keyboard.nextLine();
choice = input.charAt(0);
if (choice == 'Y' || choice == 'y')
{
        numberOfToppings += 1;
        toppings = toppings + "Sausage ";
}
System.out.print("Do you want Onion?  (Y/N):  ");
input = keyboard.nextLine();
choice = input.charAt(0);
if (choice == 'Y' || choice == 'y')
{
        numberOfToppings += 1;
        toppings = toppings + "Onion ";
}
System.out.print("Do you want Mushroom?  (Y/N):  ");
input = keyboard.nextLine();
choice = input.charAt(0);
if (choice == 'Y' || choice == 'y')
{
        numberOfToppings += 1;
        toppings = toppings + "Mushroom ";
}

//set number of toppings and topping list on pizza ordered
order.setNumToppings (numberOfToppings);
order.setToppingList(toppings);

//add additional toppings cost to cost of pizza
order.setCost(1.25*numberOfToppings);

//display order confirmation
System.out.println();
System.out.println("Your order is as follows: ");
System.out.println(order.getSize() + " inch pizza");
System.out.println(order.getCrust() + " crust");
System.out.println(order.getToppingList());

//display cost of pizza
cost = order.getCost();

//apply discount if user is elibible
//ADD LINES FOR TASK #4 HERE
```

```
            //EDIT PROGRAM FOR TASK #5
            //SO ALL MONEY OUTPUT APPEARS WITH 2 DECIMAL PLACES
            System.out.println("The cost of your order is: $" + cost);

            //calculate and display tax and total cost
            tax = cost * TAX_RATE;
            System.out.println("The tax is:  $" + tax);
            System.out.println("The total due is:  $" + (tax+cost));

            System.out.println("Your order will be ready" +
                    " for pickup in 30 minutes.");
        }
}
```