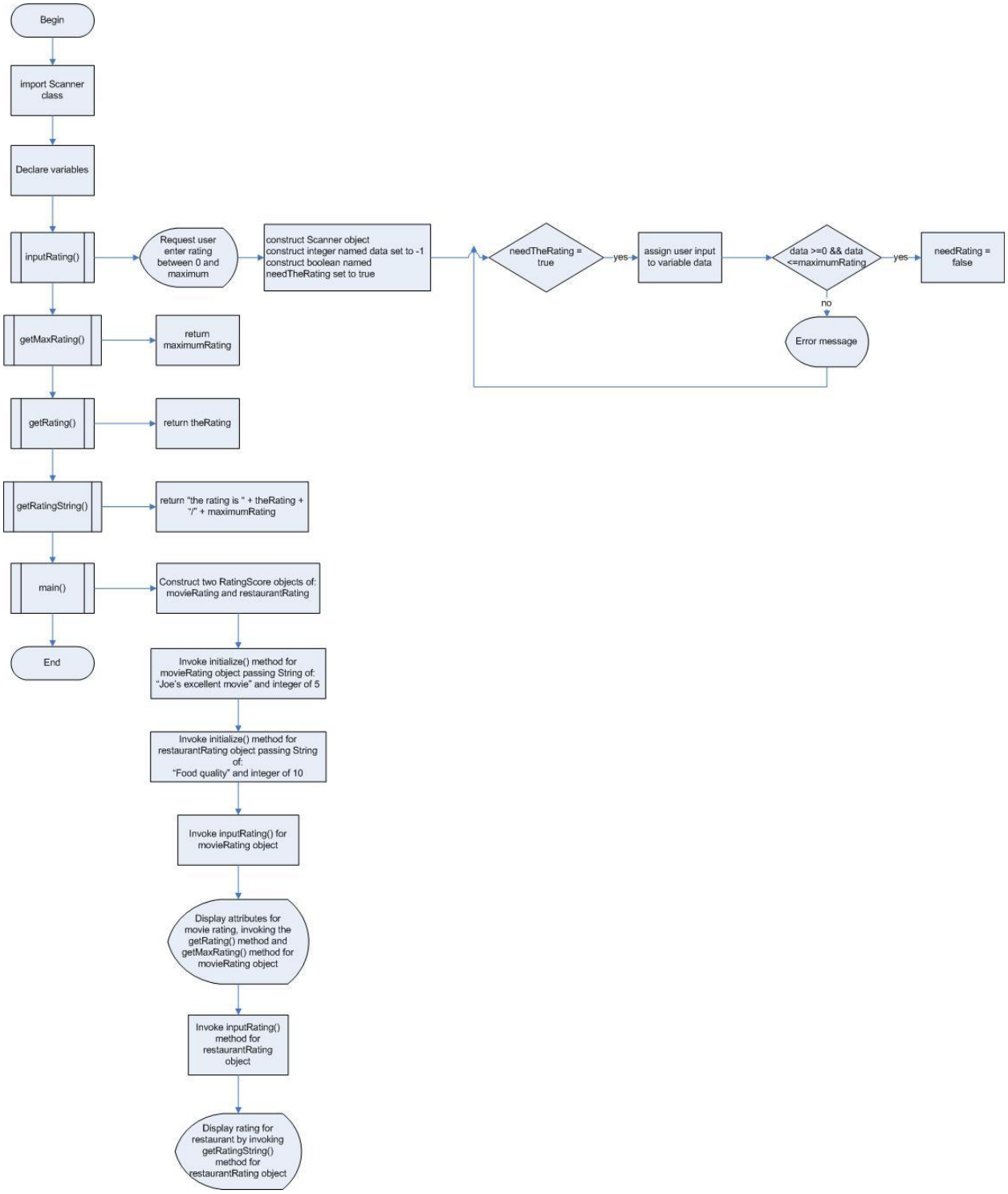
CIT 149: Java I
Chapter 5 Lab 1

In this lab we will write a program that allows you to rate a movie and restaurant. All actions will be placed in separate methods and will be invoked by the main method. The flowchart for this lab is:


Begin
import Scanner class
Declare variables
inputRating()
Request user enter rating between 0 and maximum
construct Scanner object
construct integer named data set to -1
construct boolean named needTheRating set to true
needTheRating = true
yes
assign user input to variable data
data >=0 && data <=maximumRating
yes
needRating = false
no
Error message
getMaxRating()
return maximumRating
getRating()
return theRating
getRatingString()
return "the rating is " + theRating + "/" + maximumRating
main()
Construct two RatingScore objects of: movieRating and restaurantRating
End
Invoke initialize() method for movieRating object passing String of: "Joe's excellent movie" and integer of 5
Invoke initialize() method for restaurantRating object passing String of: "Food quality" and integer of 10
Invoke inputRating() for movieRating object
Display attributes for movie rating, invoking the getRating() method and getMaxRating() method for movieRating object
Invoke inputRating() method for restaurantRating object
Display rating for restaurant by invoking getRatingString() method for restaurantRating object

1. Open a new document window in TextPad and save the program as RatingScore.java.

2. Enter the block comments that will include the programmer's name, the program name, the date, and the purpose of the program.

3. Type the import statement that will import the Scanner class as done in previous chapters.

4. Type the class header, and opening brace

5. Type the following code that will declare several variables:

```
private String description;
private int maximumRating;
private int theRating;
```

   - Notice that each of these variables are private? This variables are instance variables. This will force the program to access the variables through a class's methods.

6. First we will create a method named initialize(). This method will receive the value of a String and integer from the point where the method is invoked. It will set the *description* variable to the value of the String passed to it, and set the *maximumRating* variable to the value of the integer passed to it. It will also set the *theRating* variable to -1. Type:

```
public void initialize(String desc, int max)
{
    description = desc;
    maximumRating = max;
    theRating = -1;
}
```

   - When you have methods with an argument of different variables it means that this method receives values from another place. Wherever this method is invoked, the place where it is invoked must pass values to the method for the method to handle.

7. We will next create a method named inputRating(). This method will have no argument, but will simply perform a function. Wherever you see a method listed as void it means the method performs its code and does not return a value anywhere. Type:

```
public void inputRating()
{
    System.out.println("What is your rating for " + description + "?");
    System.out.println("Please enter an integer from 0 to " + maximumRating);

    Scanner keyboard = new Scanner(System.in);

    int data = -1;
    boolean needTheRating = true;

    while(needTheRating)
    {

        data = keyboard.nextInt();

        if(data>=0 && data<=maximumRating)
        {
            needTheRating = false;
        }
        else
        {
            System.out.println("Sorry, that rating is out of range.");
            System.out.println("Please enter an integer from 0 to " + maximumRating);
        }
    }
    theRating = data;
}
```

- Here we request the user enter an integer from 0 to the value of *maximumRating*
- A Scanner object is constructed
- An integer named *data* is constructed and set to -1
- A boolean is constructed and set to true

- A while loop will run as long as *needTheRating* is true. Within the loop:
    - The user input is applied to the variable *data*
    - if *data* is greater or equal to zero AND *data* is also less than or equal to *maximumRating* then:
        - *needTheRating* is set to false
    - else a message is displayed to the user
- *theRating* variable is set to the value of *data*

8. Our next two methods simply return the current value of two variables. Type:

```
public int getMaxRating()
{
    return maximumRating;
}

 public int getRating()
{
    return theRating;
}
```

- These are accessor methods in that they access values of variables and return the value where the method is invoked. Accessor methods must always have the type of data it will return in its method header. They cannot every be void.

9. Our next method is also an accessor method but it returns both text and the value of a variable. Type:

```
public String getRatingString()
{
    return "the rating is " + theRating + "/" + maximumRating;
}
```

10. Our last method is the main method. Type:

```
public static void main(String[] args)
{
    RatingScore movieRating = new RatingScore();
   RatingScore restaurantRating = new RatingScore();

   movieRating.initialize("Joe's excellent movie", 5);
   restaurantRating.initialize("Food quality", 10);

   movieRating.inputRating();

   System.out.println("Displaying the attributes for the movie rating.");
   System.out.println("The rating is " + movieRating.getRating() + " out of "
   + movieRating.getMaxRating());
   System.out.println();

   restaurantRating.inputRating();

   System.out.println("Displaying the rating for the restaurant.");
   System.out.println( restaurantRating.getRatingString());

}
```

- First we create two RatingScore objects which are instances of this class.
- We invoke the initialize() method for the *movieRating* object and pass to that method a String and an integer. The integer will become the value of the variable *maximumRating*.
- We invoke the initialize() method for the *restaurantRating* object. Again we pass a String and an integer to the method. This time the *maximumRating* variable for this single object will be 10.
- We invoke the inputRating() method for the *movieRating* object.
- We display the attributes for this object. Notice that within the System.out.println() we invoke the getRating(), and getMaxRating() methods for this object? These methods will return the current value of *maximumRating* and *theRating* for this object.

- We invoke the inputRating() method for the *restaurantRating* object and display the attributes for this object. Within the System.out.println() we invoke the getRatingString() method for this object. This method will return text and the value of *maximumRating*.

11. Close the class.

12. Compile the program. Fix any error if necessary.

13. Compress the .java and .class files into a single zip or rar file and submit to the appropriate drop box.