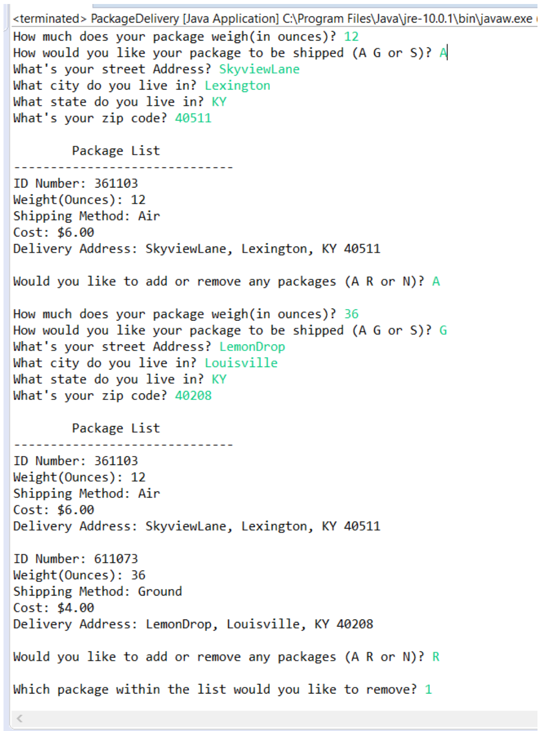
Jeremy Boyd

Assignment 03

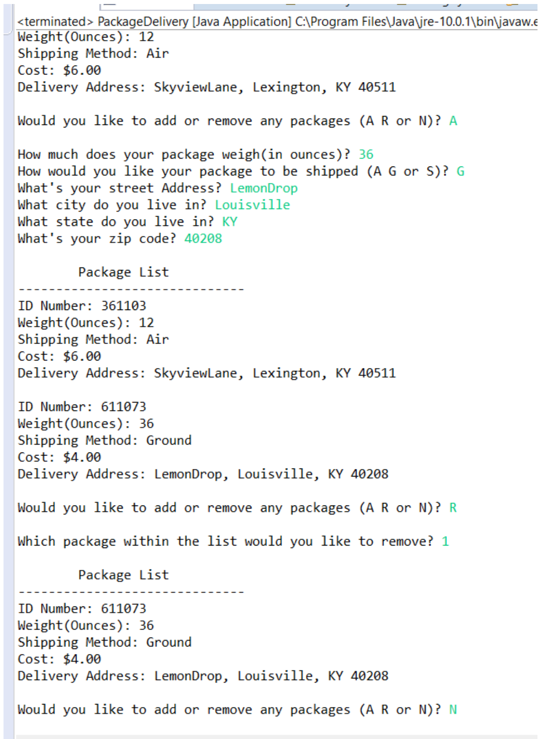
6/6/2018

Problem 1

This problem had its fair share of difficulties and was much more complex than I originally thought it would be. To start off I created the *Address* class and created the necessary instance variables *streetAddress*, *city*, *state*, and *zip code.* Then I created a constructor for the class and created a toString method that would format the output of the address nicely. Next up I created the *Package* class and created the necessary instance variables *TrackingID*, *Weight*, *Cost*, *Destination*, and the enum type *Shipping* which held the three methods of shipment: Air, Ground, and Sea. I also created the instance variable *shipMethod* of type Shipping to be my variable for my enum values. I created a constructor for the class that would accept values for the weight, shipping method, and destination. After this I created the method *calculateCost(),* it's made up of nested if statements that test what the value of the variable *Weight* is and based off its value test the method of shipment as either, air, ground, or sea. And then assigns the proper cost to the package and returns it. When I first created the class, I imported the random number generator for use in my *GenerateID()* method. Within it I initialized a random variable and set the variable *TrackingID* as a random 6-digit number that could range from 100000-999999 and returned the variable. Lastly, I created a toString method that would organize the output of the package in a nice manner. Next up I created the *PackageDelivery* child class and extended it to the *Package* class so it could use the contents of the parent class and created a method using the keyword *super* in order to gain use of the variables within the superclass. I also imported the scanner class so user input could be read in and the ArrayList class so I could create my array. Next, I created a scanner object along with multiple local variables that would be used to collect data. Then I created my ArrayList of type Package and named the it *packageList*, next I printed out a series of questions and used the scanner to read in the data. I then put the information inside a *Pakcage* variable and stored that package variable in the ArrayList. Next, I created a for loop that would print out each instance within the array. After this I asked the user if they wanted to add or remove any of the packages they created. I used switch statement that would perform each action depending on the user's choice and put it within a while loop to keep asking until the user is finished.



```
<terminated> PackageDelivery [Java Application] C:\Program Files\Java\jre-10.0.1\bin\javaw.exe
How much does your package weigh(in ounces)? 12
How would you like your package to be shipped (A G or S)? A
What's your street Address? SkyviewLane
What city do you live in? Lexington
What state do you live in? KY
What's your zip code? 40511

        Package List
------------------------------
ID Number: 361103
Weight(Ounces): 12
Shipping Method: Air
Cost: $6.00
Delivery Address: SkyviewLane, Lexington, KY 40511

Would you like to add or remove any packages (A R or N)? A

How much does your package weigh(in ounces)? 36
How would you like your package to be shipped (A G or S)? G
What's your street Address? LemonDrop
What city do you live in? Louisville
What state do you live in? KY
What's your zip code? 40208

        Package List
------------------------------
ID Number: 361103
Weight(Ounces): 12
Shipping Method: Air
Cost: $6.00
Delivery Address: SkyviewLane, Lexington, KY 40511

ID Number: 611073
Weight(Ounces): 36
Shipping Method: Ground
Cost: $4.00
Delivery Address: LemonDrop, Louisville, KY 40208

Would you like to add or remove any packages (A R or N)? R

Which package within the list would you like to remove? 1
```



```
<terminated> PackageDelivery [Java Application] C:\Program Files\Java\jre-10.0.1\bin\javaw.e
Weight(Ounces): 12
Shipping Method: Air
Cost: $6.00
Delivery Address: SkyviewLane, Lexington, KY 40511

Would you like to add or remove any packages (A R or N)? A

How much does your package weigh(in ounces)? 36
How would you like your package to be shipped (A G or S)? G
What's your street Address? LemonDrop
What city do you live in? Louisville
What state do you live in? KY
What's your zip code? 40208

        Package List
------------------------------
ID Number: 361103
Weight(Ounces): 12
Shipping Method: Air
Cost: $6.00
Delivery Address: SkyviewLane, Lexington, KY 40511

ID Number: 611073
Weight(Ounces): 36
Shipping Method: Ground
Cost: $4.00
Delivery Address: LemonDrop, Louisville, KY 40208

Would you like to add or remove any packages (A R or N)? R

Which package within the list would you like to remove? 1

        Package List
------------------------------
ID Number: 611073
Weight(Ounces): 36
Shipping Method: Ground
Cost: $4.00
Delivery Address: LemonDrop, Louisville, KY 40208

Would you like to add or remove any packages (A R or N)? N
```

Problem 2

Overall this problem wasn't too difficult but the final result looked pretty cool, first thing I did was import all the necessary classes to create the shapes, color, random number generator, etc. Next within the *primaryStage* I created a variable to be my random number generator called *random* and created a group variable called *circles* that would hold all my circle objects. Then I created a for loop and initialized the variable *count* within it to keep count of how many circles had been created and set it so it wouldn't terminate till *count* reached 100. Within the for loop I initialized three integer variables that would represent the red green and blue sections that make up each color. And set them as random number generators from 0-255 to encompass all the shades per channel. Next, I initialized two more integer variables that would represent the centered X and Y position of each circle. I created them as random number generators from 40 to 425 to make sure every circle would be inside the designated 500 by 500 window. Then I initialized another integer variable to represent the radius and set it as a random number generator from 1 to 40. Next, I initialized a color variable called *fill* as null, and then I created an if statement that would test the radius value for each circle and if it was less than or equal to thirty. The *fill* variable would be set equal to the random *r g* and *b* variables respectively in order to randomize the fill color of each variable but the circle would remain colorless if it's radius was greater than 30. After this I created a circle variable and set its parameters to my *x*, *y*, and *radius* variables so the parameters for each circle would be randomized. Next I set the stroke to black for each circle and set the fill color equal to my *fill* variable. And then I used the *getChildren()* method to add each circle to the group *circles* as it's created. And lastly, I created a scene variable which included my *circles* group that holds all the circles and set the background color to white.