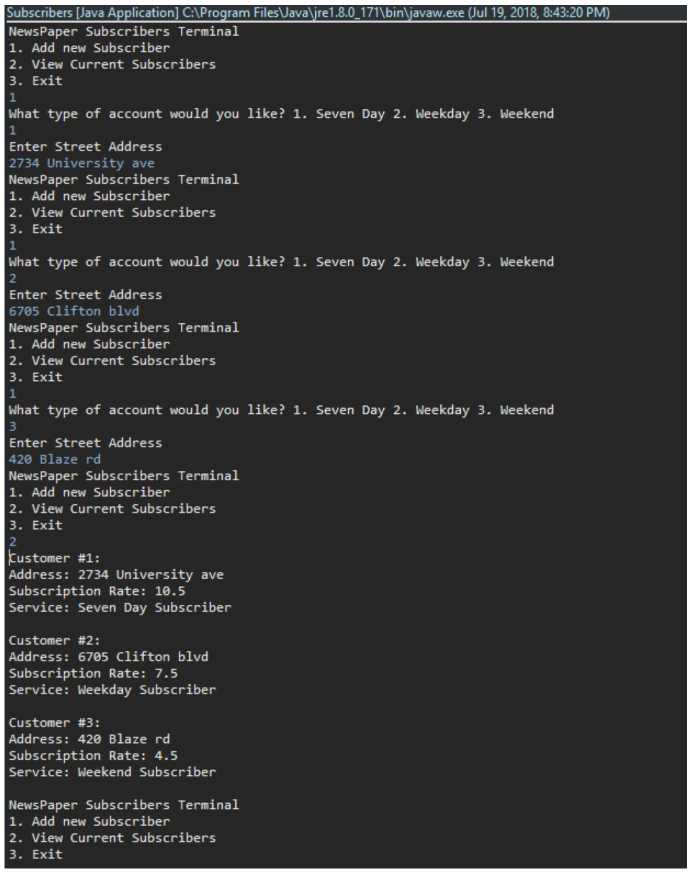
CECS 220 Assignment #5

Nicholas Gittings

July 19, 2018

For problem #1: First, create a class called NewsPaperSubscriber which has two private variables which are a String for the subscriber's address and a double for the subscription rate. Create setter and getters for the variables for encapsulation, next create a setRate method which will take a parameter double that sets the subscription rate. This will be used later in the child classes to set rate. Now, create an equals method which will check to see if two objects of NewsPaperSubscriber are equal and this will be done by checking the object's address. Next, create 3 child classes which inherit from NewsPaperSubscriber called SevenDaySubscriber, WeekdaySubscriber, and WeekendSubscriber. All of these child classes will have toString methods which return a formatted string of the address and subscription rate. The constructors for each child class will accept a String parameter that initializes street address and calls the setRate method to set the rate for the type of service. Lastly, create a Subscriber class which will act as a client and have a main method which will allow the user to use a menu to create and view subscribers.



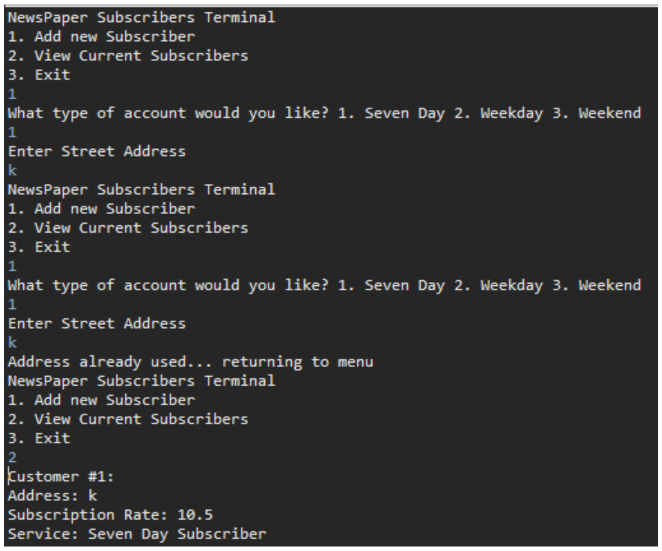
```
Subscribers [Java Application] C:\Program Files\Java\jre1.8.0_171\bin\javaw.exe (Jul 19, 2018, 8:43:20 PM)
NewsPaper Subscribers Terminal
1. Add new Subscriber
2. View Current Subscribers
3. Exit
1
What type of account would you like? 1. Seven Day 2. Weekday 3. Weekend
1
Enter Street Address
2734 University ave
NewsPaper Subscribers Terminal
1. Add new Subscriber
2. View Current Subscribers
3. Exit
1
What type of account would you like? 1. Seven Day 2. Weekday 3. Weekend
2
Enter Street Address
6705 Clifton blvd
NewsPaper Subscribers Terminal
1. Add new Subscriber
2. View Current Subscribers
3. Exit
1
What type of account would you like? 1. Seven Day 2. Weekday 3. Weekend
3
Enter Street Address
420 Blaze rd
NewsPaper Subscribers Terminal
1. Add new Subscriber
2. View Current Subscribers
3. Exit
2
Customer #1:
Address: 2734 University ave
Subscription Rate: 10.5
Service: Seven Day Subscriber

Customer #2:
Address: 6705 Clifton blvd
Subscription Rate: 7.5
Service: Weekday Subscriber

Customer #3:
Address: 420 Blaze rd
Subscription Rate: 4.5
Service: Weekend Subscriber

NewsPaper Subscribers Terminal
1. Add new Subscriber
2. View Current Subscribers
3. Exit
```

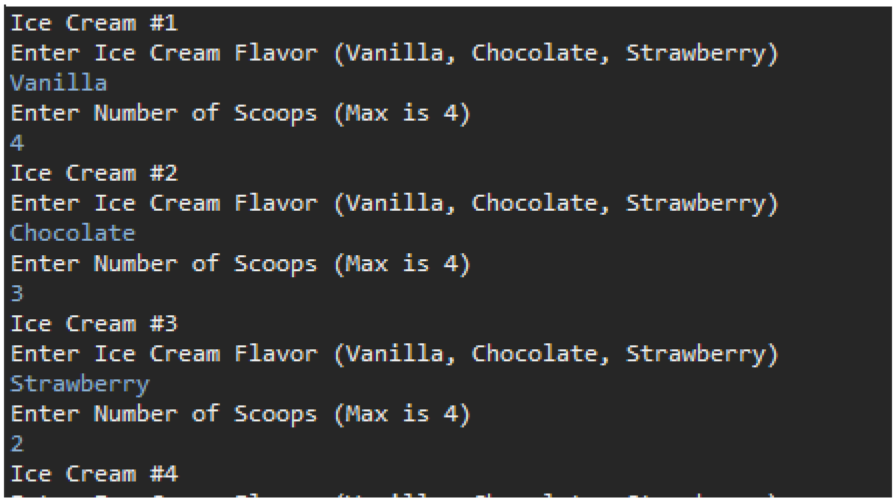
Screenshot of inputs and displaying list



```
NewsPaper Subscribers Terminal
1. Add new Subscriber
2. View Current Subscribers
3. Exit
1
What type of account would you like? 1. Seven Day 2. Weekday 3. Weekend
1
Enter Street Address
k
NewsPaper Subscribers Terminal
1. Add new Subscriber
2. View Current Subscribers
3. Exit
1
What type of account would you like? 1. Seven Day 2. Weekday 3. Weekend
1
Enter Street Address
k
Address already used... returning to menu
NewsPaper Subscribers Terminal
1. Add new Subscriber
2. View Current Subscribers
3. Exit
2
Customer #1:
Address: k
Subscription Rate: 10.5
Service: Seven Day Subscriber
```

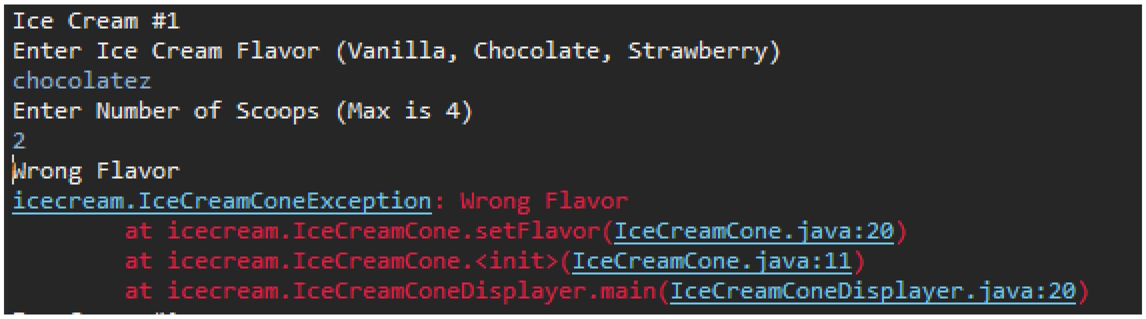
Screenshot of attempting to enter same address

For problem #2: First, create a class called IceCreamConeException which has a String parameter that takes a message to pass to the parent class Exception. Next, create a class called IceCreamCone which has a String called Flavor, Integer called Scoops, and a String ArrayList called FLAVORS. The constructor will take the String and Integer variables as parameters. Create setFlavor and setScoops methods which will set the variables or if setFlavor doesn't match flavor list or setScoops is greater than 4, it will throw an IceCreamConeException. Lastly, create a client class with a main method called IceCreamConeDisplayer. This class will create an ArrayList of IceCreamCone objects and allow user input to enter flavors and number of scoops. Create a while loop which will use a counter which will stop the input loop after it has taken 10 scoops. If the user inputs a wrong flavor or number of scoops over 4 it will throw the IceCreamConeException and display the error. Lastly, after 10 objects have been entered it will print the ArrayList of IceCreamCone objects.



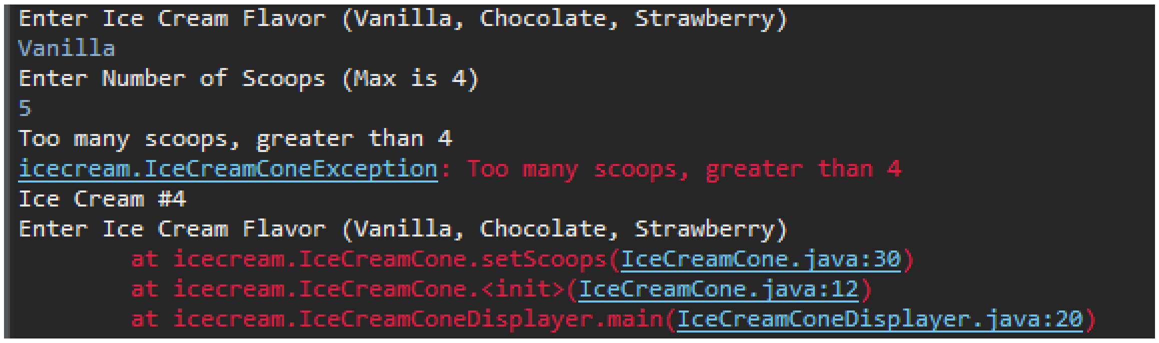
```
Ice Cream #1
Enter Ice Cream Flavor (Vanilla, Chocolate, Strawberry)
Vanilla
Enter Number of Scoops (Max is 4)
4
Ice Cream #2
Enter Ice Cream Flavor (Vanilla, Chocolate, Strawberry)
Chocolate
Enter Number of Scoops (Max is 4)
3
Ice Cream #3
Enter Ice Cream Flavor (Vanilla, Chocolate, Strawberry)
Strawberry
Enter Number of Scoops (Max is 4)
2
Ice Cream #4
```

Screenshot of successful inputs



```
Ice Cream #1
Enter Ice Cream Flavor (Vanilla, Chocolate, Strawberry)
chocolatez
Enter Number of Scoops (Max is 4)
2
Wrong Flavor
icecream.IceCreamConeException: Wrong Flavor
	at icecream.IceCreamCone.setFlavor(IceCreamCone.java:20)
	at icecream.IceCreamCone.<init>(IceCreamCone.java:11)
	at icecream.IceCreamConeDisplayer.main(IceCreamConeDisplayer.java:20)
```

Screenshot of if the user enters a wrong flavor



```
Enter Ice Cream Flavor (Vanilla, Chocolate, Strawberry)
Vanilla
Enter Number of Scoops (Max is 4)
5
Too many scoops, greater than 4
icecream.IceCreamConeException: Too many scoops, greater than 4
Ice Cream #4
Enter Ice Cream Flavor (Vanilla, Chocolate, Strawberry)
	at icecream.IceCreamCone.setScoops(IceCreamCone.java:30)
	at icecream.IceCreamCone.<init>(IceCreamCone.java:12)
	at icecream.IceCreamConeDisplayer.main(IceCreamConeDisplayer.java:20)
```

Screenshot of if the user enters too many scoops