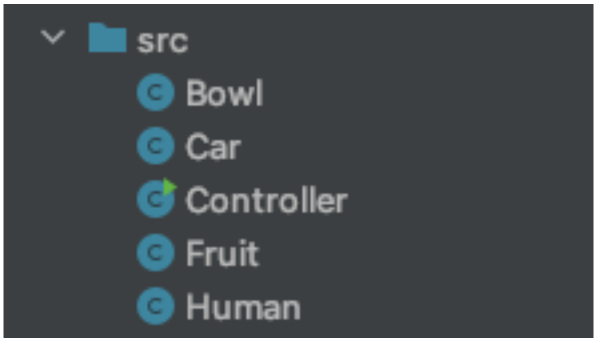
COP 2210 Lab 10

IMPORTANT: The goal is to work through the lab and get it running. If you are getting stuck ASK THE LAB ASSISTANT OR OTHER STUDENTS to help you. It is a good idea to work together in the lab. You will learn a lot this way.

Start by opening Intellij and create a new Project and name the project **COP2210_Lab9** then add the needed package and classes to the src folder. When you are done your project should look like the image below. **If you do not know how to do this look at the earlier labs or ask a TA.**

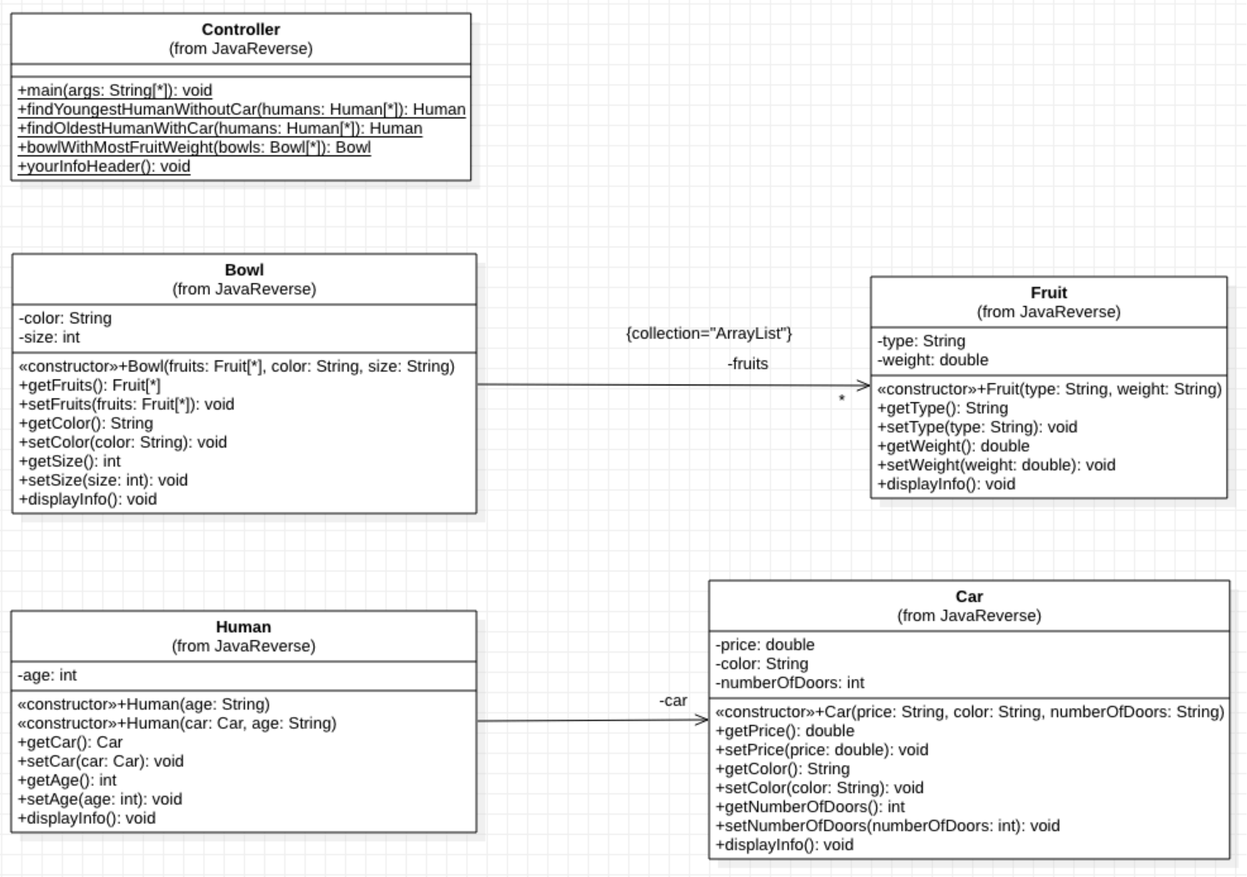


```
src
    Bowl
    Car
    Controller
    Fruit
    Human
```

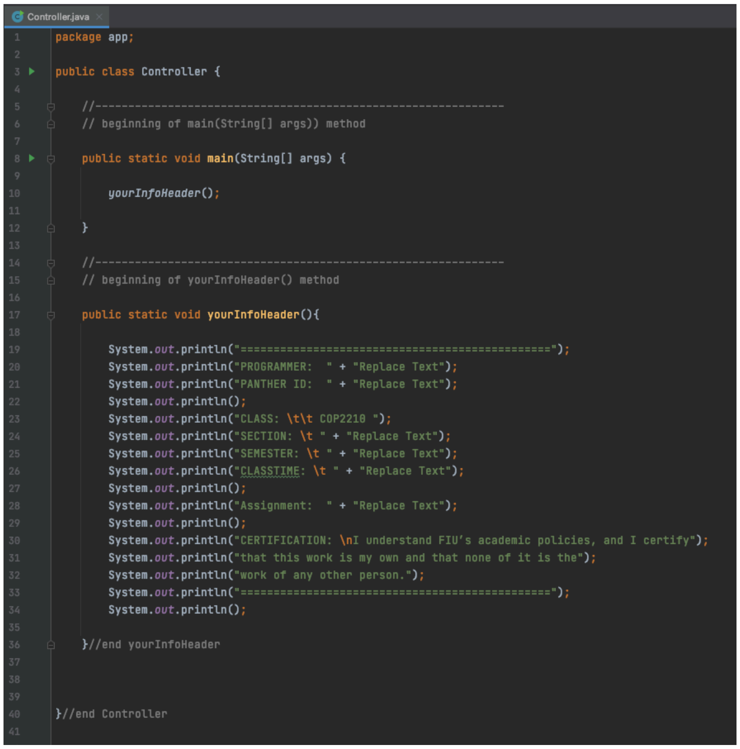
Here is the text in data.txt :

```
Fruit,Pear,67.6
Fruit,Apple,126.5
Fruit,Mango,87.1
Human,25,Car->YES,Car,51907.68,White,4
Fruit,Orange,266.3
Bowl,Red,12
Car,48533.78,Blue,2
Fruit,Orange,416.6
Car,50172.71,Red,2
Fruit,Orange,232.5
Bowl,Green,10
Bowl,Red,13,Fruit->YES,Fruit,Orange,408.0,Fruit,Pear,293.9,Fruit,Orange,140.7,Fruit,Mango,207.7
Fruit,Mango,115.3
Fruit,Orange,198.1
Bowl,Red,13
Bowl,Blue,12,Fruit->YES,Fruit,Orange,287.6,Fruit,Mango,190.8,Fruit,Apple,385.8,Fruit,Pear,183.1
Human,48
Bowl,White,10
Human,12
Fruit,Apple,115.8
Bowl,Blue,11,Fruit->YES,Fruit,Orange,129.0
Bowl,Blue,13,Fruit->YES,Fruit,Pear,68.4,Fruit,Apple,357.2,Fruit,Pear,361.5,Fruit,Apple,249.9
Car,56287.36,White,4
Human,45
Human,17
Bowl,Green,12
Bowl,White,11,Fruit->YES,Fruit,Mango,73.9,Fruit,Mango,360.6,Fruit,Mango,413.1,Fruit,Pear,203.9
Car,41774.36,White,2
Car,51720.78,Blue,2
Car,63641.93,Black,4
Fruit,Orange,309.4
Human,51,Car->YES,Car,31717.69,Blue,2
Bowl,Blue,12,Fruit->YES,Fruit,Pear,363.0
Bowl,Black,12,Fruit->YES,Fruit,Apple,384.5,Fruit,Pear,368.4,Fruit,Orange,184.7
Fruit,Pear,91.2
Fruit,Pear,394.0
Car,48310.36,Black,4
Human,22,Car->YES,Car,26886.42,Green,2
Fruit,Orange,337.5
Bowl,Green,12
Car,40748.31,Black,2
Human,67
Human,23,Car->YES,Car,29767.56,White,4
Bowl,Blue,12,Fruit->YES,Fruit,Orange,401.4
Car,39897.81,Black,4
Car,41360.77,White,2
Human,69,Car->YES,Car,63158.19,Red,2
Car,34347.53,Red,2
Human,49,Car->YES,Car,52668.61,White,2
Human,33
```

Here is the UML Diagram of the classes:


Controller
(from JavaReverse)
+main(args: String[*]): void
+findYoungestHumanWithoutCar(humans: Human[*]): Human
+findOldestHumanWithCar(humans: Human[*]): Human
+bowlWithMostFruitWeight(bowls: Bowl[*]): Bowl
+yourInfoHeader(): void
Bowl
(from JavaReverse)
-color: String
-size: int
«constructor»+Bowl(fruits: Fruit[*], color: String, size: String)
+getFruits(): Fruit[*]
+setFruits(fruits: Fruit[*]): void
+getColor(): String
+setColor(color: String): void
+getSize(): int
+setSize(size: int): void
+displayInfo(): void
{collection="ArrayList"}
-fruits
*
Fruit
(from JavaReverse)
-type: String
-weight: double
«constructor»+Fruit(type: String, weight: String)
+getType(): String
+setType(type: String): void
+getWeight(): double
+setWeight(weight: double): void
+displayInfo(): void
Human
(from JavaReverse)
-age: int
«constructor»+Human(age: String)
«constructor»+Human(car: Car, age: String)
+getCar(): Car
+setCar(car: Car): void
+getAge(): int
+setAge(age: int): void
+displayInfo(): void
-car
Car
(from JavaReverse)
-price: double
-color: String
-numberOfDoors: int
«constructor»+Car(price: String, color: String, numberOfDoors: String)
+getPrice(): double
+setPrice(price: double): void
+getColor(): String
+setColor(color: String): void
+getNumberOfDoors(): int
+setNumberOfDoors(numberOfDoors: int): void
+displayInfo(): void

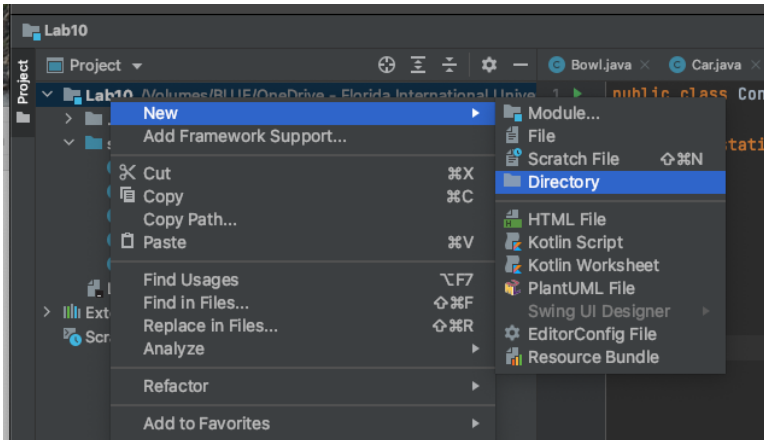
Step 0:
Add the following code to you Controller class. In yourInfoHeader() method replace the placeholders with your information and then run your code to see if it works.

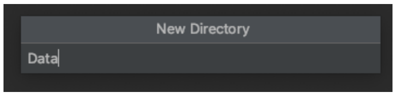


```java
package app;

public class Controller {

    //--------------------------------------------------------------
    // beginning of main(String[] args)) method

    public static void main(String[] args) {

        yourInfoHeader();

    }

    //--------------------------------------------------------------
    // beginning of yourInfoHeader() method

    public static void yourInfoHeader(){

        System.out.println("===============================================");
        System.out.println("PROGRAMMER:  " + "Replace Text");
        System.out.println("PANTHER ID:  " + "Replace Text");
        System.out.println();
        System.out.println("CLASS: \t\t COP2210 ");
        System.out.println("SECTION: \t " + "Replace Text");
        System.out.println("SEMESTER: \t " + "Replace Text");
        System.out.println("CLASSTIME: \t " + "Replace Text");
        System.out.println();
        System.out.println("Assignment:  " + "Replace Text");
        System.out.println();
        System.out.println("CERTIFICATION: \nI understand FIU’s academic policies, and I certify");
        System.out.println("that this work is my own and that none of it is the");
        System.out.println("work of any other person.");
        System.out.println("===============================================");
        System.out.println();

    }//end yourInfoHeader



}//end Controller
```

Step 1:
In you IntelliJ Project create a new directory name Data, see images below.

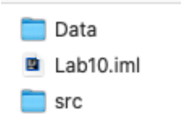
Important: Notice, I selected the COP2210_Lab10 folder not the src folder.



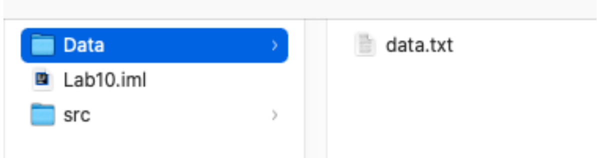


Data

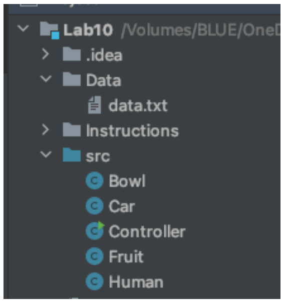
This will create a new folder named Data within your Lab10 Project on your hard drive.



Now move a copy of the data.txt file that you downloaded from the Lab10 assignment canvas page into the Data folder.



Back in IntelliJ your Project should look like this. If it does not ask someone to help you.



Step 2:
In the Controller class main() write the following code:

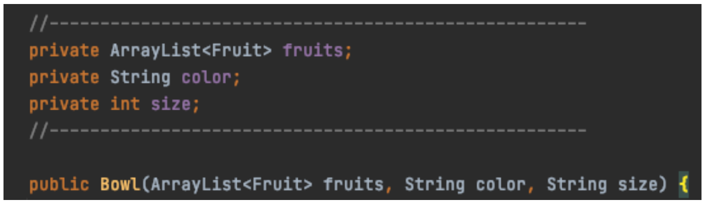
Notice the throws Exceptions and try catch blocks.
This code reads the data.txt file and outputs its content to the console.
Look at the fileLocation variable this is important if you are using a mac or windows os.

```java
public static void main(String[] args) throws Exception{

    //Step 0
    yourInfoHeader();


    //--------------------------
    // Step 1
    //--------------------------
    System.out.println();

    System.out.println();
    System.out.println("=========================================");
    System.out.println("Data.txt output");
    System.out.println("=========================================");

    try{

        // use this if you are on a mac
        String fileLocation = "./Data/data.txt";

        // use this if you are on a windows
        // String fileLocation = ".\Data\data.txt";
        BufferedReader br = new BufferedReader(new FileReader("./data/data.txt"));
        String line;

        String strCurrentLine;

        while((strCurrentLine = br.readLine())!=null) {
            System.out.println(strCurrentLine);
        }//end while
    }catch(IOException e){

        e.printStackTrace();

    }

    //-----------------------------------------



}//end main
```

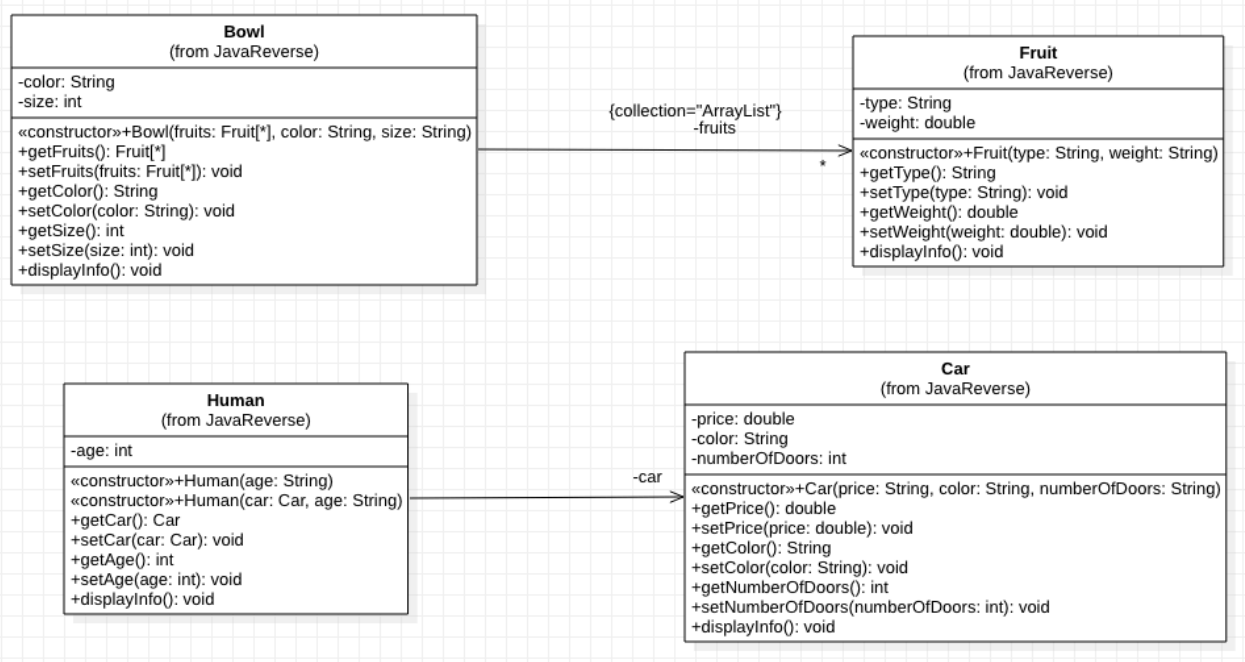
Console output:

```
=========================================
Data.txt output
=========================================
Fruit,Pear,67.6
Fruit,Apple,126.5
Fruit,Mango,87.1
Human,25,Car->YES,Car,51907.68,White,4
Fruit,Orange,266.3
Bowl,Red,12
Car,48533.78,Blue,2
Fruit,Orange,416.6
Car,50172.71,Red,2
Fruit,Orange,232.5
Bowl,Green,10
Bowl,Red,13,Fruit->YES,Fruit,Orange,408.0,Fruit,Pear,293.9,Fruit,Orange,140.7,Fruit,Mango,207.7
Fruit,Mango,115.3
Fruit,Orange,198.1
Bowl,Red,13
Bowl,Blue,12,Fruit->YES,Fruit,Orange,287.6,Fruit,Mango,190.8,Fruit,Apple,385.8,Fruit,Pear,183.1
Human,48
Bowl,White,10
Human,12
Fruit,Apple,115.8
Bowl,Blue,11,Fruit->YES,Fruit,Orange,129.0
Bowl,Blue,13,Fruit->YES,Fruit,Pear,68.4,Fruit,Apple,357.2,Fruit,Pear,361.5,Fruit,Apple,249.9
Car,56287.36,White,4
Human,45
Human,17
Bowl,Green,12
Bowl,White,11,Fruit->YES,Fruit,Mango,73.9,Fruit,Mango,360.6,Fruit,Mango,413.1,Fruit,Pear,203.9
Car,41774.36,White,2
Car,51720.78,Blue,2
Car,63641.93,Black,4
Fruit,Orange,309.4
Human,51,Car->YES,Car,31717.69,Blue,2
Bowl,Blue,12,Fruit->YES,Fruit,Pear,363.0
Bowl,Black,12,Fruit->YES,Fruit,Apple,384.5,Fruit,Pear,368.4,Fruit,Orange,184.7
Fruit,Pear,91.2
Fruit,Pear,394.0
Car,48310.36,Black,4
Human,22,Car->YES,Car,26886.42,Green,2
Fruit,Orange,337.5
Bowl,Green,12
Car,40748.31,Black,2
Human,67
Human,23,Car->YES,Car,29767.56,White,4
Bowl,Blue,12,Fruit->YES,Fruit,Orange,401.4
Car,39897.81,Black,4
Car,41360.77,White,2
Human,69,Car->YES,Car,63158.19,Red,2
Car,34347.53,Red,2
Human,49,Car->YES,Car,52668.61,White,2
Human,33
```

Step 3:
Write all the shell code for the classes **Bowl**, **Car**, **Fruit** and **Human** classes. Be careful a class constructor sometimes may take a different input type than what the instance variable are, see below. Hint, look at the UML carefully.



```java
//--------------------------------------------------
private ArrayList<Fruit> fruits;
private String color;
private int size;
//--------------------------------------------------

public Bowl(ArrayList<Fruit> fruits, String color, String size) {
```



Step 4:
In the Controller class main() under the code you wrote in step 2.

Create the following arraylists:

- **cars** which is an arraylist that holds only Car objects
- **humans** which is an arraylist that holds only Human objects
- **fruits** which is an arraylist that holds only Fruit objects
- **bowls** which is an arraylist that holds only Bowl objects

Under these arraylists write the following code:

```
try {

    // for mac os
    BufferedReader br = new BufferedReader(new FileReader(“./data/data.txt”));

    // for window os
    //BufferedReader br = new BufferedReader(new FileReader(".\data\data.txt"));


    String line;

    String strCurrentLine;

    while((strCurrentLine = br.readLine())!=null) {

// YOU WRITE CODE HERE
      HERE WRITE THE CODE THE PARSE ALL THE CARS FROM THE data.txt
      AND CREATES A NEW CAR INSTANCE WITH THE PARSE DATA
      THEN ADD THE NEW CAR INSTANCE TO THE cars ARRAYLIST


    }
}catch(IOException e){

    e.printStackTrace();

}
```

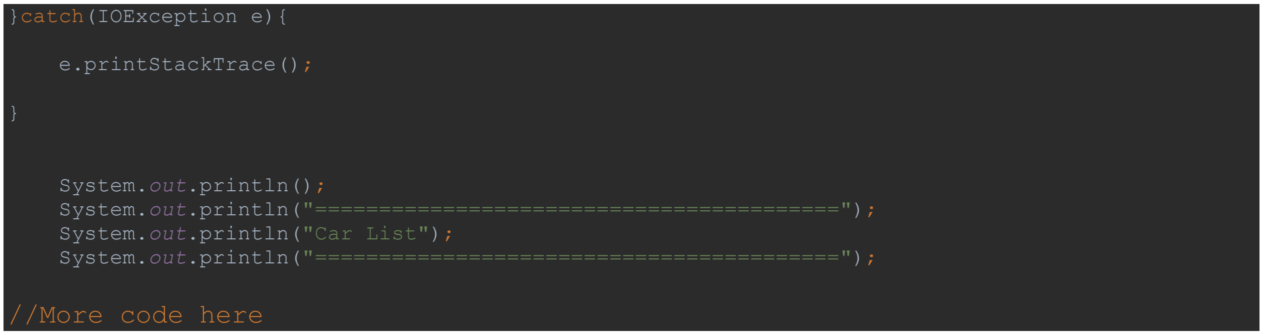
Hint: use the String split function and split on “,”. Store the results in a String[] named tokens

Look at the format of a Car line of txt:

Type	Price	Color	Number of Doors
Car	50172.71	Red	2

Hint: use an **if** statement with **equals()** check if the type is a “Car”

Step 5:
In the Controller class main() under the code you wrote in step 4, write the code that calls the displayInfo() for each car instance within the cars arraylist. You must use an **enhance** for loop for this.



```
}catch(IOException e){

    e.printStackTrace();

}


    System.out.println();
    System.out.println("=========================================");
    System.out.println("Car List");
    System.out.println("=========================================");

//More code here
```

Note you need to complete the displayInfo() of the Car class so it matches the output shown below.

Hint: use printf() for this

Console Output:

```
=========================================
Car List
=========================================
Type->Car-> Price: $48533.78     Color:Blue           Doors:2
Type->Car-> Price: $50172.71     Color:Red            Doors:2
Type->Car-> Price: $56287.36     Color:White          Doors:4
Type->Car-> Price: $41774.36     Color:White          Doors:2
Type->Car-> Price: $51720.78     Color:Blue           Doors:2
Type->Car-> Price: $63641.93     Color:Black          Doors:4
Type->Car-> Price: $48310.36     Color:Black          Doors:4
Type->Car-> Price: $40748.31     Color:Black          Doors:2
Type->Car-> Price: $39897.81     Color:Black          Doors:4
Type->Car-> Price: $41360.77     Color:White          Doors:2
Type->Car-> Price: $34347.53     Color:Red            Doors:2
```

Step 6:
In the Controller class main() under the code you wrote in step 4 continue the if statement with if else condition that parses all the humans out of the data.txt file and create a human instance and stores the instance within the **humans** arraylist.

Important, note a human may have a car so you will need to deal with this.
If the human does have a car you must do the following:

- Create a car instance with the car information
- Attached this car to the human instance. Hint look at the human constructor in the UML diagram.
- Store the car instance in the **cars** arraylist

data.txt format (Human without a car)

```
Human,67
```

data.txt format (Human with a car)

```
Human,23,Car->YES,Car,29767.56,White,4
```

Note, a human can only have one car assigned to it.

Step 7:
In the Controller class main() under the code you wrote in step 5, write the code that calls the displayInfo() for each human instance within the **humans** arraylist. You must use an **enhance** for loop for this.

Note you need to complete the displayInfo() of the Human class so it matches the output shown below.

Hint: use printf() for this

Console Output:

```
=========================================
Human List
=========================================
Type->Human-> Age: 25
        Type->Car-> Price: $51907.68         Color:White     Doors:4
Type->Human-> Age: 48
Type->Human-> Age: 12
Type->Human-> Age: 45
Type->Human-> Age: 17
Type->Human-> Age: 51
        Type->Car-> Price: $31717.69         Color:Blue      Doors:2
Type->Human-> Age: 22
        Type->Car-> Price: $26886.42         Color:Green     Doors:2
Type->Human-> Age: 67
Type->Human-> Age: 23
        Type->Car-> Price: $29767.56         Color:White     Doors:4
Type->Human-> Age: 69
        Type->Car-> Price: $63158.19         Color:Red       Doors:2
Type->Human-> Age: 49
        Type->Car-> Price: $52668.61         Color:White     Doors:2
Type->Human-> Age: 33
```

Step 8:
In the Controller class main() under the code you wrote in step 6 continue the if else statement with if else condition that parses all the fruits out of the data.txt file and create a fruit instance and stores the instance within the **fruits** arraylist.

data.txt format of Fruit

```
Fruit,Apple,126.5
```

Step 9:
In the Controller class main() under the code you wrote in step 7, write the code that calls the displayInfo() for each fruit instance within the **fruits** arraylist. You must use an **enhance** for loop for this.

Note you need to complete the displayInfo() of the Fruit class so it matches the output shown below.

Hint: use printf() for this

Console Output:

```
=========================================
Fruit List
=========================================
Type->Fruit-> Type: Pear              Weight:67.6
Type->Fruit-> Type: Apple             Weight:126.5
Type->Fruit-> Type: Mango             Weight:87.1
Type->Fruit-> Type: Orange            Weight:266.3
Type->Fruit-> Type: Orange            Weight:416.6
Type->Fruit-> Type: Orange            Weight:232.5
Type->Fruit-> Type: Mango             Weight:115.3
Type->Fruit-> Type: Orange            Weight:198.1
Type->Fruit-> Type: Apple             Weight:115.8
Type->Fruit-> Type: Orange            Weight:309.4
Type->Fruit-> Type: Pear              Weight:91.2
Type->Fruit-> Type: Pear              Weight:394.0
Type->Fruit-> Type: Orange            Weight:337.5
```

Step 10:
In the Controller class main() under the code you wrote in step 8 continue the if else statement with if else condition that parses all the bowls out of the data.txt file and create a bowl instance and stores the instance within the **bowl** arraylist.

Note a bowl may have 0-5 fruits within it.

data.txt format (Bowl without fruit)

```
Bowl,Green,10
```

data.txt format (Bowl with fruit)

```
Bowl,Blue,13,Fruit->YES,Fruit,Pear,68.4,Fruit,Apple,357.2,Fruit,Pear,361.5,Fruit,Apple,249.9
```

Important, note a bowl may have 0-5 fruits so you will need to deal with this.
If the bowl does have fruits you must do the following:

This is tricky and similar to step 6 but different.

- Create a fruit instance for each fruit within the bowl. Hint use a for loop … this is tricky.
- Attached this each fruit instance to the bowl instance . Hint look at the bowl constructor in the UML diagram look for an arraylist.
- Store each fruit instance in the **fruits** arraylist

Step 11:
In the Controller class main() under the code you wrote in step 9, write the code that calls the displayInfo() for each bowl instance within the **bowls** arraylist. You must use an **enhance** for loop for this.

Note you need to complete the displayInfo() of the Bowl class so it matches the output shown below.

Hint: use printf() for this

Console Output:

```
=========================================
Bowl List
=========================================
Type->Bowl-> Color: Red               size:12
Type->Bowl-> Color: Green             size:10
Type->Bowl-> Color: Red               size:13
               Type->Fruit-> Type: Orange            Weight:408.0
               Type->Fruit-> Type: Pear              Weight:293.9
               Type->Fruit-> Type: Orange            Weight:140.7
               Type->Fruit-> Type: Mango             Weight:207.7
Type->Bowl-> Color: Red               size:13
Type->Bowl-> Color: Blue              size:12
               Type->Fruit-> Type: Orange            Weight:287.6
               Type->Fruit-> Type: Mango             Weight:190.8
               Type->Fruit-> Type: Apple             Weight:385.8
               Type->Fruit-> Type: Pear              Weight:183.1
Type->Bowl-> Color: White             size:10
Type->Bowl-> Color: Blue              size:11
               Type->Fruit-> Type: Orange            Weight:129.0
Type->Bowl-> Color: Blue              size:13
               Type->Fruit-> Type: Pear              Weight:68.4
               Type->Fruit-> Type: Apple             Weight:357.2
               Type->Fruit-> Type: Pear              Weight:361.5
               Type->Fruit-> Type: Apple             Weight:249.9
Type->Bowl-> Color: Green             size:12
Type->Bowl-> Color: White             size:11
               Type->Fruit-> Type: Mango             Weight:73.9
               Type->Fruit-> Type: Mango             Weight:360.6
               Type->Fruit-> Type: Mango             Weight:413.1
               Type->Fruit-> Type: Pear              Weight:203.9
Type->Bowl-> Color: Blue              size:12
               Type->Fruit-> Type: Pear              Weight:363.0
Type->Bowl-> Color: Black             size:12
               Type->Fruit-> Type: Apple             Weight:384.5
               Type->Fruit-> Type: Pear              Weight:368.4
               Type->Fruit-> Type: Orange            Weight:184.7
Type->Bowl-> Color: Green             size:12
Type->Bowl-> Color: Blue              size:12
               Type->Fruit-> Type: Orange            Weight:401.4
```

Step 12:
In the Controller class below the main() write the code for the following method:

```
public static Human findYoungestHumanWithoutCar(ArrayList<Human> humans){
```

This method returns the youngest human without a car in the **humans** arraylist.

Step 13:
In the Controller class main() under the code you wrote in step 11, write the code that calls the findYoungestHumanWithoutCar(ArrayList<Human> humans) and calls the displayInfo() on the returned human instance

Hint code:

```
System.out.println();
System.out.println("=========================================");
System.out.println("Youngest Human without Car");
System.out.println("=========================================");

// NEED SOME CODE HERE
System.out.println();
```

Console Output:

```
=========================================
Youngest Human without Car
=========================================
Type->Human-> Age: 12
```

Step 14:
In the Controller class below the findYoungestHumanWithoutCar(ArrayList<Human> humans) write the code for the following method:

```
public static Human findOldestHumanWithCar(ArrayList<Human> humans){
```

This method returns the oldest human with a car in the **humans** arraylist.

Step 15:
In the Controller class main() under the code you wrote in step 14, write the code that calls the findOldestHumanWithCar(ArrayList<Human> humans) and calls the displayInfo() on the returned human instance

Console Output:

```
=========================================
Oldest Human with Car
=========================================
Type->Human-> Age: 69
               Type->Car-> Price: $63158.19         Color:Red       Doors:2
```

Step 16:
In the Controller class below the findOldestHumanWithCar(ArrayList<Human> humans) write the code for the following method:

```
public static Bowl bowlWithMostFruitWeight(ArrayList<Bowl> bowls){
```

This method returns the bowl which contains the most fruit measured by total weight of all the fruit within the bowl. This is tricky.

Step 15:
In the Controller class main() under the code you wrote in step 15, write the code that calls the bowlWithMostFruitWeight(ArrayList<Bowl> bowls)and calls the displayInfo() on the returned bowl instance.

Console Output:

```
=========================================
Bowl with the most fruit by weight
=========================================
Type->Bowl-> Color: White            size:11
               Type->Fruit-> Type: Mango            Weight:73.9
               Type->Fruit-> Type: Mango            Weight:360.6
               Type->Fruit-> Type: Mango            Weight:413.1
               Type->Fruit-> Type: Pear             Weight:203.9
```

Step 19:
If it is working zip the project and upload the zipped project file to canvas using the assignment link.

YOUR DONE ☺