Group: Amar Korac & Pranesh Gandhi

There are various types of fitness tracking apps that utilize a vast database. Within such database exists few categories such as; nutrition, workout, and activity tracking. They play an important role in redefining fitness tracking technology. Our real life scenario will focus on nutrition. Users will moderate their weight loss or gain. The nutrition tracking will provide users with the data of calories consumed. The user can search the app's database for food, input the portion of the food and automatically output calories and macros for a meal. The importance in this is that people can finally understand what enters their bodies, discover the desired BMI for either weight loss or gain, and observe average calories ate on a timed basis.

2. On the second page, mention the type of data you used:

i. did you use real data (yes or no), which tables and attributes are covered with such real data set.

We used real data for table food attribute common name, table tracking attributes carbs, proteins,fat, calories, table user attribute height and weight.

ii. did you use java to generate random data, which tables and attributes are covered with such random data.

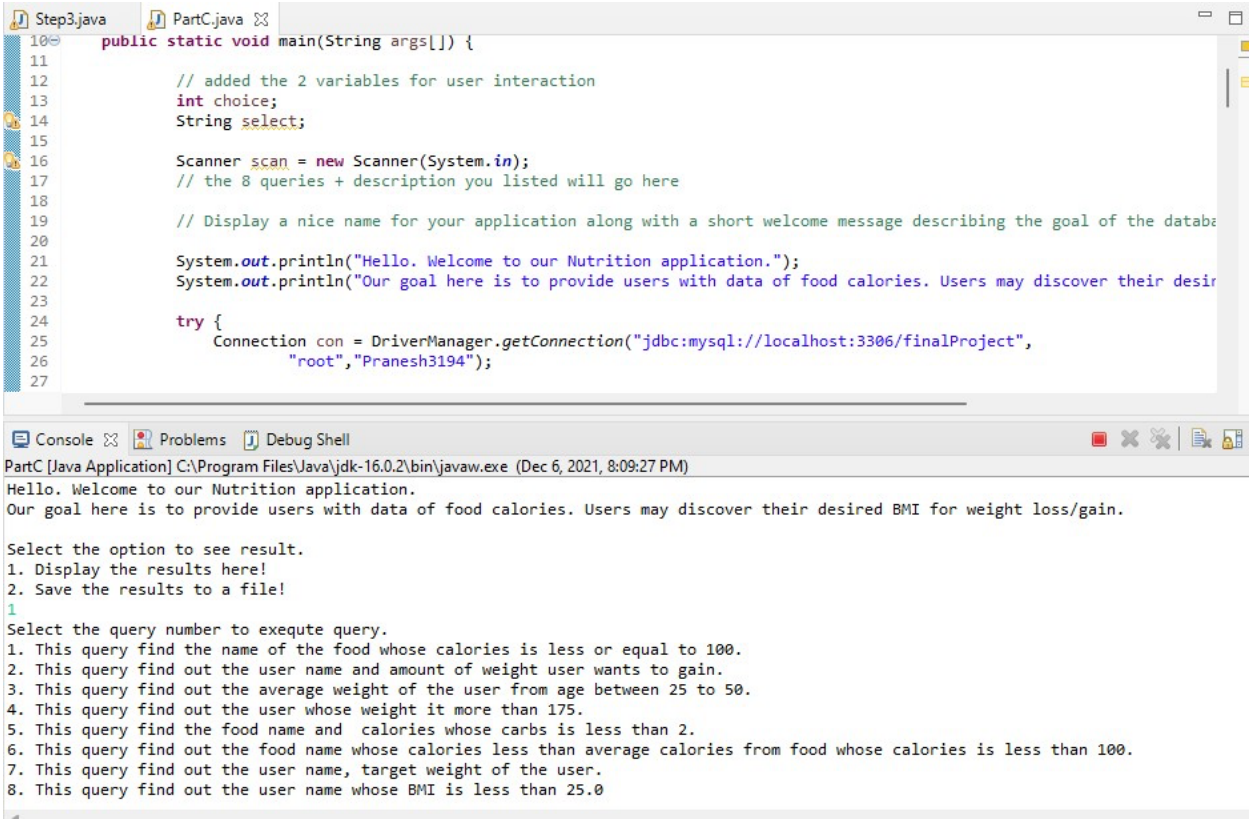
No, we did not generate random data using java.

iii. did you use online tools to generate random data, which tables and attributes are covered with such data
set

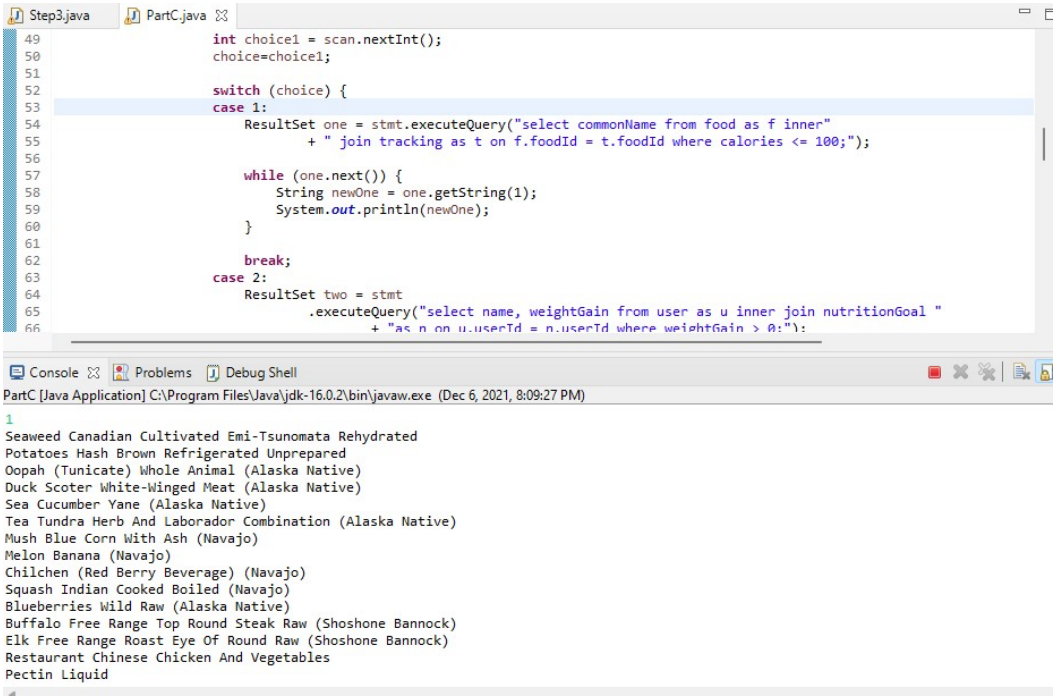
We generate random data for table user attribute name, age and table nutritiongoal attribute weightLoss, weightGain, athlete.

3. On the third page, one screenshot of your running application to show if you developed a terminal, GUI, or web application along with the menu of queries for user interaction. Do not take screenshots with a camera.

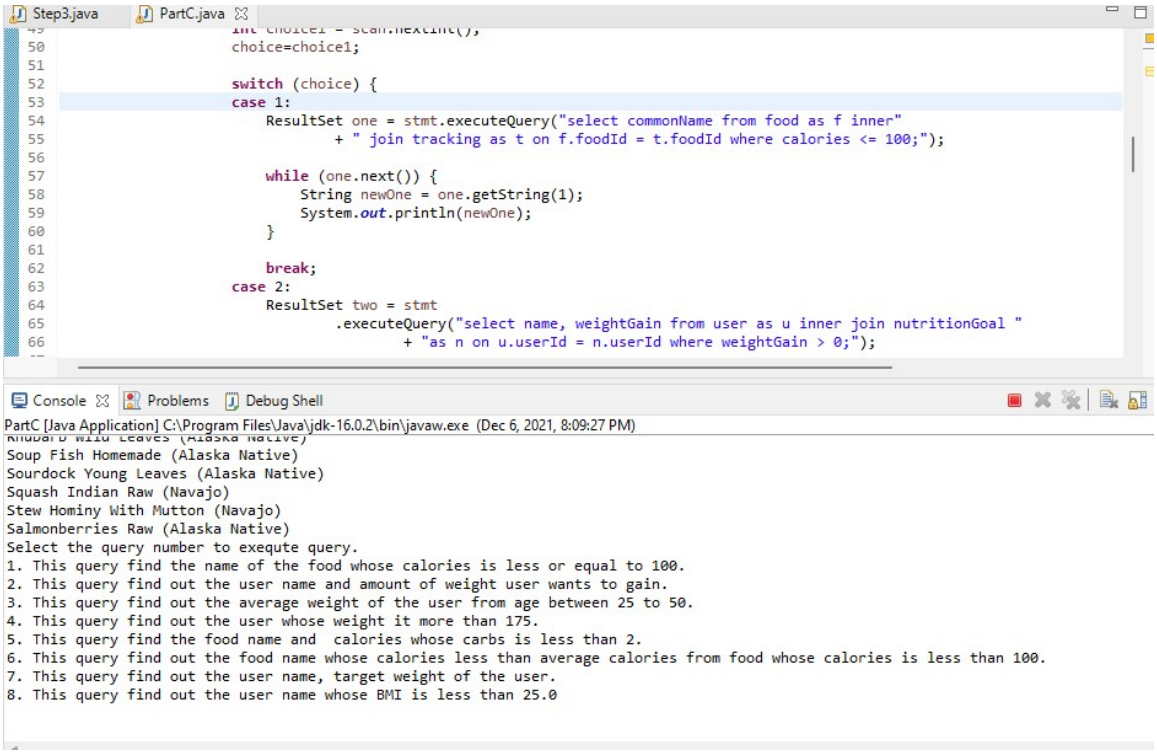
User Interaction



4**. On the following page(s), for each of the 8 queries, take screenshot(s) for the SQL query you developed as well as the output. Do not take screenshots with a camera.**
Case 1

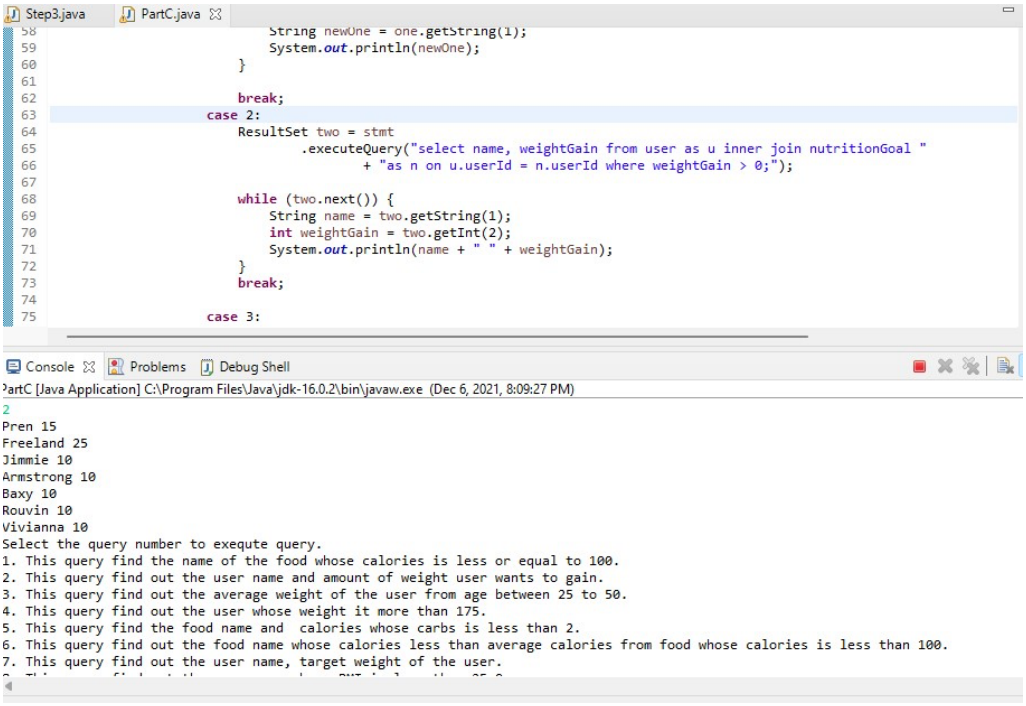


```
Step3.java  PartC.java

49                  int choice1 = scan.nextInt();
50                  choice=choice1;
51
52                  switch (choice) {
53                  case 1:
54                      ResultSet one = stmt.executeQuery("select commonName from food as f inner"
55                              + " join tracking as t on f.foodId = t.foodId where calories <= 100;");
56
57                      while (one.next()) {
58                          String newOne = one.getString(1);
59                          System.out.println(newOne);
60                      }
61
62                      break;
63                  case 2:
64                      ResultSet two = stmt
65                              .executeQuery("select name, weightGain from user as u inner join nutritionGoal "
66                                      + "as n on u.userId = n.userId where weightGain > 0;");

Console  Problems  Debug Shell
PartC [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe  (Dec 6, 2021, 8:09:27 PM)

1
Seaweed Canadian Cultivated Emi-Tsunomata Rehydrated
Potatoes Hash Brown Refrigerated Unprepared
Oopah (Tunicate) Whole Animal (Alaska Native)
Duck Scoter White-Winged Meat (Alaska Native)
Sea Cucumber Yane (Alaska Native)
Tea Tundra Herb And Laborador Combination (Alaska Native)
Mush Blue Corn With Ash (Navajo)
Melon Banana (Navajo)
Chilchen (Red Berry Beverage) (Navajo)
Squash Indian Cooked Boiled (Navajo)
Blueberries Wild Raw (Alaska Native)
Buffalo Free Range Top Round Steak Raw (Shoshone Bannock)
Elk Free Range Roast Eye Of Round Raw (Shoshone Bannock)
Restaurant Chinese Chicken And Vegetables
Pectin Liquid
```



```
Step3.java  PartC.java

50                  choice=choice1;
51
52                  switch (choice) {
53                  case 1:
54                      ResultSet one = stmt.executeQuery("select commonName from food as f inner"
55                              + " join tracking as t on f.foodId = t.foodId where calories <= 100;");
56
57                      while (one.next()) {
58                          String newOne = one.getString(1);
59                          System.out.println(newOne);
60                      }
61
62                      break;
63                  case 2:
64                      ResultSet two = stmt
65                              .executeQuery("select name, weightGain from user as u inner join nutritionGoal "
66                                      + "as n on u.userId = n.userId where weightGain > 0;");

Console  Problems  Debug Shell
PartC [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe  (Dec 6, 2021, 8:09:27 PM)
Soup Fish Homemade (Alaska Native)
Sourdock Young Leaves (Alaska Native)
Squash Indian Raw (Navajo)
Stew Hominy With Mutton (Navajo)
Salmonberries Raw (Alaska Native)
Select the query number to exequte query.
1. This query find the name of the food whose calories is less or equal to 100.
2. This query find out the user name and amount of weight user wants to gain.
3. This query find out the average weight of the user from age between 25 to 50.
4. This query find out the user whose weight it more than 175.
5. This query find the food name and  calories whose carbs is less than 2.
6. This query find out the food name whose calories less than average calories from food whose calories is less than 100.
7. This query find out the user name, target weight of the user.
8. This query find out the user name whose BMI is less than 25.0
```

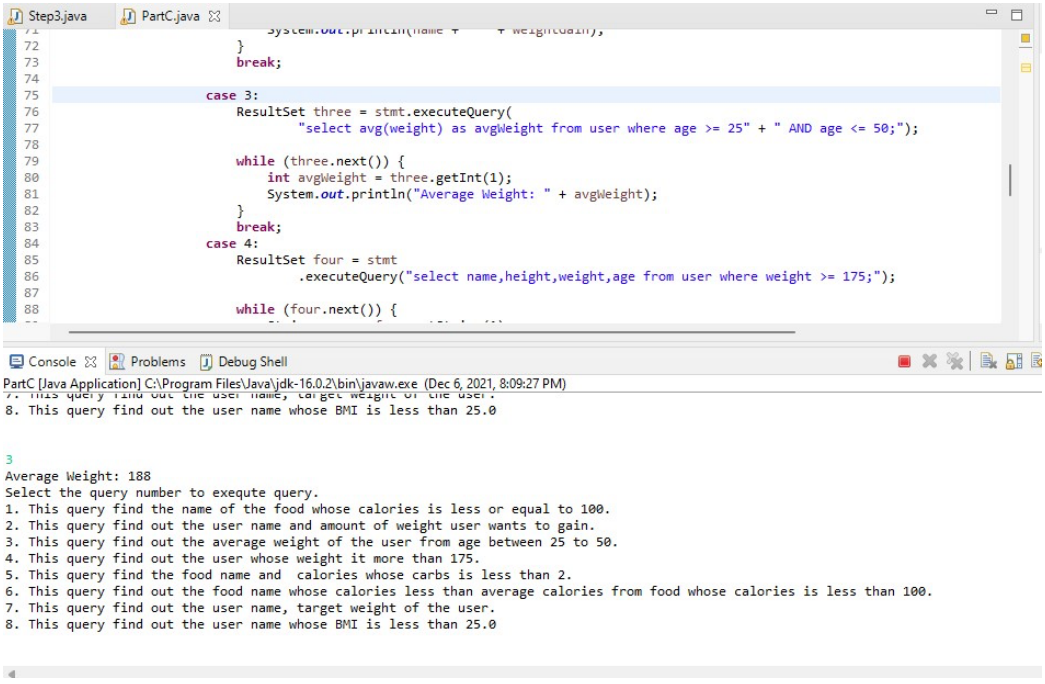
Case 2



```
                        String newOne = one.getString(1);
                        System.out.println(newOne);
                    }

                    break;
                case 2:
                    ResultSet two = stmt
                            .executeQuery("select name, weightGain from user as u inner join nutritionGoal "
                                    + "as n on u.userId = n.userId where weightGain > 0;");

                    while (two.next()) {
                        String name = two.getString(1);
                        int weightGain = two.getInt(2);
                        System.out.println(name + " " + weightGain);
                    }
                    break;

                case 3:
```

```
PartC [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe  (Dec 6, 2021, 8:09:27 PM)
2
Pren 15
Freeland 25
Jimmie 10
Armstrong 10
Baxy 10
Rouvin 10
Vivianna 10
Select the query number to exequte query.
1. This query find the name of the food whose calories is less or equal to 100.
2. This query find out the user name and amount of weight user wants to gain.
3. This query find out the average weight of the user from age between 25 to 50.
4. This query find out the user whose weight it more than 175.
5. This query find the food name and  calories whose carbs is less than 2.
6. This query find out the food name whose calories less than average calories from food whose calories is less than 100.
7. This query find out the user name, target weight of the user.
```

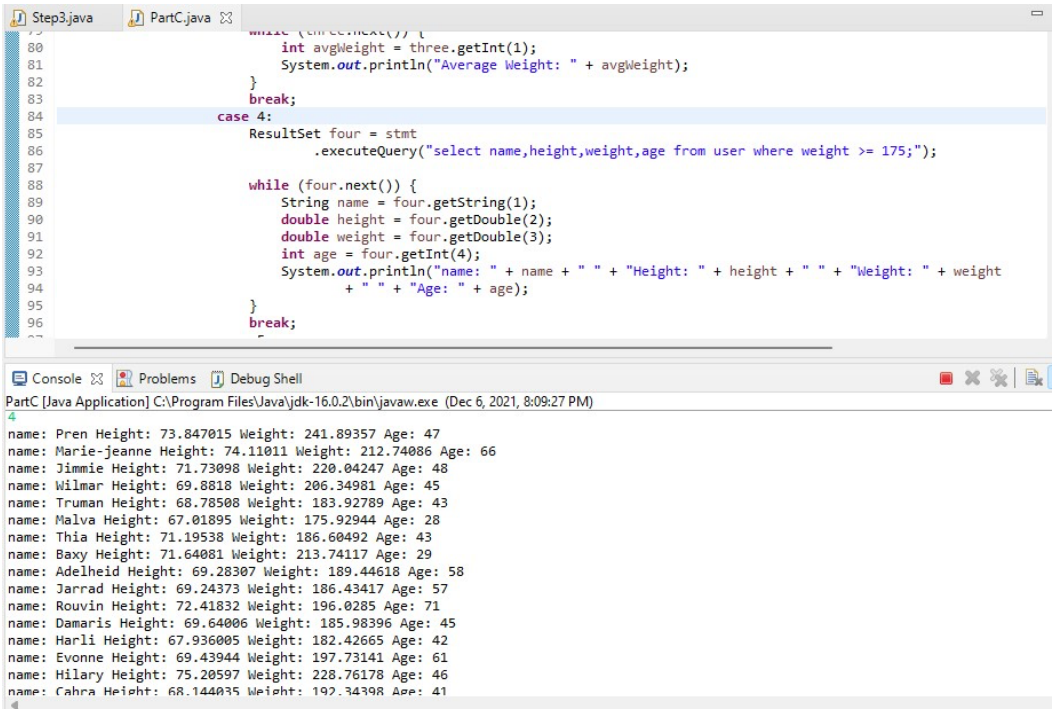
Case 3



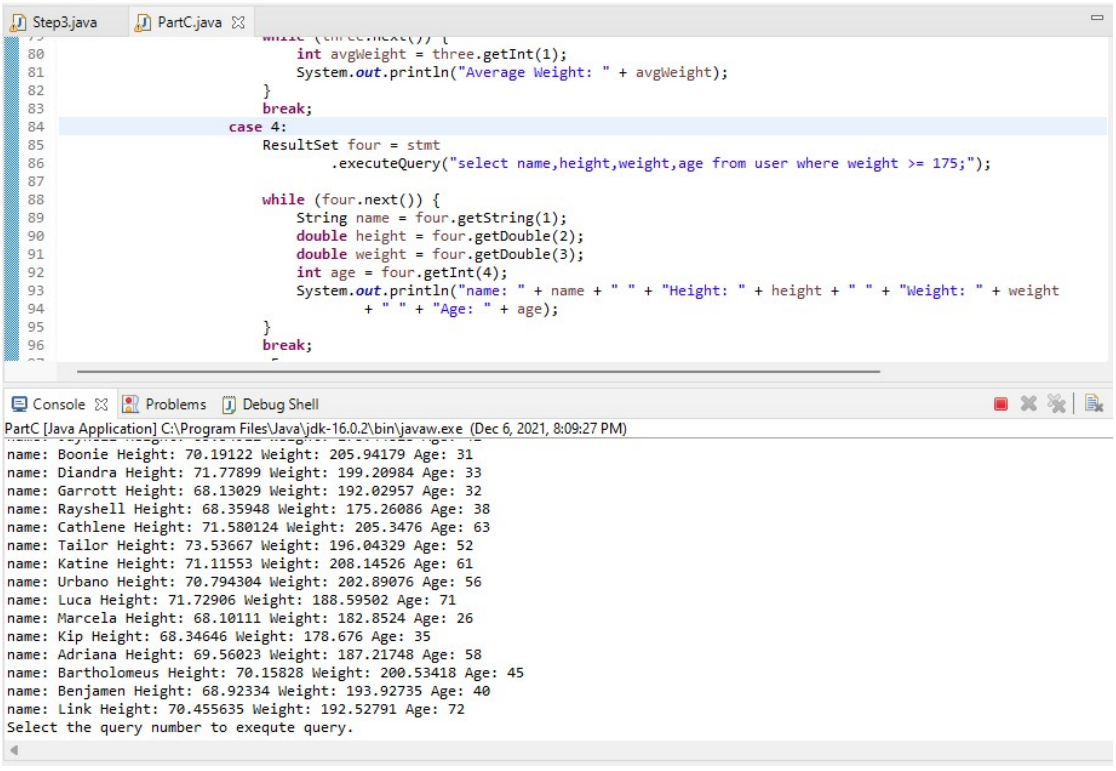
```
                    }
                    break;

                case 3:
                    ResultSet three = stmt.executeQuery(
                            "select avg(weight) as avgWeight from user where age >= 25" + " AND age <= 50;");

                    while (three.next()) {
                        int avgWeight = three.getInt(1);
                        System.out.println("Average Weight: " + avgWeight);
                    }
                    break;
                case 4:
                    ResultSet four = stmt
                            .executeQuery("select name,height,weight,age from user where weight >= 175;");

                    while (four.next()) {
```

```
PartC [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe  (Dec 6, 2021, 8:09:27 PM)
8. This query find out the user name whose BMI is less than 25.0

3
Average Weight: 188
Select the query number to exequte query.
1. This query find the name of the food whose calories is less or equal to 100.
2. This query find out the user name and amount of weight user wants to gain.
3. This query find out the average weight of the user from age between 25 to 50.
4. This query find out the user whose weight it more than 175.
5. This query find the food name and  calories whose carbs is less than 2.
6. This query find out the food name whose calories less than average calories from food whose calories is less than 100.
7. This query find out the user name, target weight of the user.
8. This query find out the user name whose BMI is less than 25.0
```

Case 4

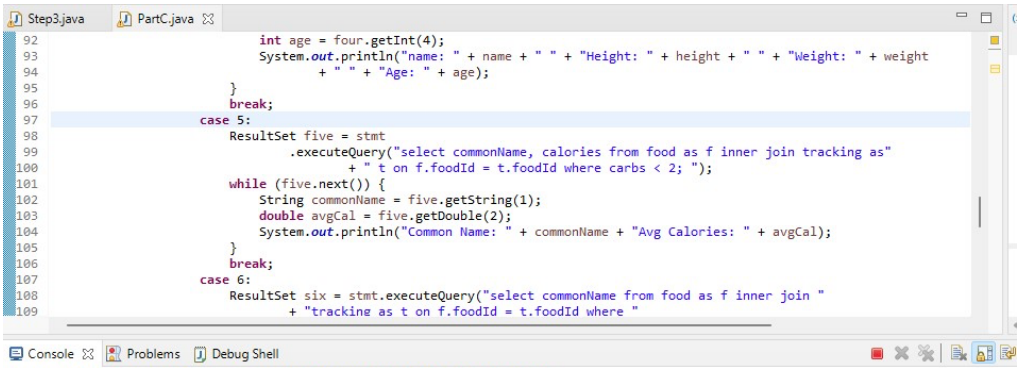


```
Step3.java  PartC.java
80                      int avgWeight = three.getInt(1);
81                      System.out.println("Average Weight: " + avgWeight);
82                  }
83                  break;
84              case 4:
85                  ResultSet four = stmt
86                          .executeQuery("select name,height,weight,age from user where weight >= 175;");
87
88                  while (four.next()) {
89                      String name = four.getString(1);
90                      double height = four.getDouble(2);
91                      double weight = four.getDouble(3);
92                      int age = four.getInt(4);
93                      System.out.println("name: " + name + " " + "Height: " + height + " " + "Weight: " + weight
94                              + " " + "Age: " + age);
95                  }
96                  break;

Console  Problems  Debug Shell
PartC [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe  (Dec 6, 2021, 8:09:27 PM)
4
name: Pren Height: 73.847015 Weight: 241.89357 Age: 47
name: Marie-jeanne Height: 74.11011 Weight: 212.74086 Age: 66
name: Jimmie Height: 71.73098 Weight: 220.04247 Age: 48
name: Wilmar Height: 69.8818 Weight: 206.34981 Age: 45
name: Truman Height: 68.78508 Weight: 183.92789 Age: 43
name: Malva Height: 67.01895 Weight: 175.92944 Age: 28
name: Thia Height: 71.19538 Weight: 186.60492 Age: 43
name: Baxy Height: 71.64081 Weight: 213.74117 Age: 29
name: Adelheid Height: 69.28307 Weight: 189.44618 Age: 58
name: Jarrad Height: 69.24373 Weight: 186.43417 Age: 57
name: Rouvin Height: 72.41832 Weight: 196.0285 Age: 71
name: Damaris Height: 69.64006 Weight: 185.98396 Age: 45
name: Harli Height: 67.936005 Weight: 182.42665 Age: 42
name: Evonne Height: 69.43944 Weight: 197.73141 Age: 61
name: Hilary Height: 75.20597 Weight: 228.76178 Age: 46
name: Cahra Height: 68.144035 Weight: 192.34398 Age: 41
```



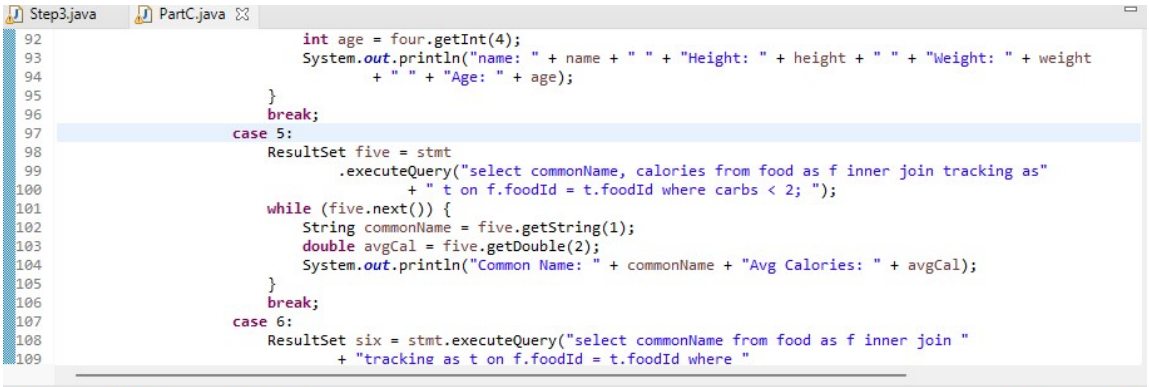
```
Step3.java  PartC.java
80                      int avgWeight = three.getInt(1);
81                      System.out.println("Average Weight: " + avgWeight);
82                  }
83                  break;
84              case 4:
85                  ResultSet four = stmt
86                          .executeQuery("select name,height,weight,age from user where weight >= 175;");
87
88                  while (four.next()) {
89                      String name = four.getString(1);
90                      double height = four.getDouble(2);
91                      double weight = four.getDouble(3);
92                      int age = four.getInt(4);
93                      System.out.println("name: " + name + " " + "Height: " + height + " " + "Weight: " + weight
94                              + " " + "Age: " + age);
95                  }
96                  break;

Console  Problems  Debug Shell
PartC [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe  (Dec 6, 2021, 8:09:27 PM)
name: Boonie Height: 70.19122 Weight: 205.94179 Age: 31
name: Diandra Height: 71.77899 Weight: 199.20984 Age: 33
name: Garrott Height: 68.13029 Weight: 192.02957 Age: 32
name: Rayshell Height: 68.35948 Weight: 175.26086 Age: 38
name: Cathlene Height: 71.580124 Weight: 205.3476 Age: 63
name: Tailor Height: 73.53667 Weight: 196.04329 Age: 52
name: Katine Height: 71.11553 Weight: 208.14526 Age: 61
name: Urbano Height: 70.794304 Weight: 202.89076 Age: 56
name: Luca Height: 71.72906 Weight: 188.59502 Age: 71
name: Marcela Height: 68.10111 Weight: 182.8524 Age: 26
name: Kip Height: 68.34646 Weight: 178.676 Age: 35
name: Adriana Height: 69.56023 Weight: 187.21748 Age: 58
name: Bartholomeus Height: 70.15828 Weight: 200.53418 Age: 45
name: Benjamen Height: 68.92334 Weight: 193.92735 Age: 40
name: Link Height: 70.455635 Weight: 192.52791 Age: 72
Select the query number to exequte query.
```

Case 5



```
                        int age = four.getInt(4);
                        System.out.println("name: " + name + " " + "Height: " + height + " " + "Weight: " + weight
                                + " " + "Age: " + age);
                    }
                    break;
                case 5:
                    ResultSet five = stmt
                            .executeQuery("select commonName, calories from food as f inner join tracking as"
                                    + " t on f.foodId = t.foodId where carbs < 2; ");
                    while (five.next()) {
                        String commonName = five.getString(1);
                        double avgCal = five.getDouble(2);
                        System.out.println("Common Name: " + commonName + "Avg Calories: " + avgCal);
                    }
                    break;
                case 6:
                    ResultSet six = stmt.executeQuery("select commonName from food as f inner join "
                            + "tracking as t on f.foodId = t.foodId where "
```

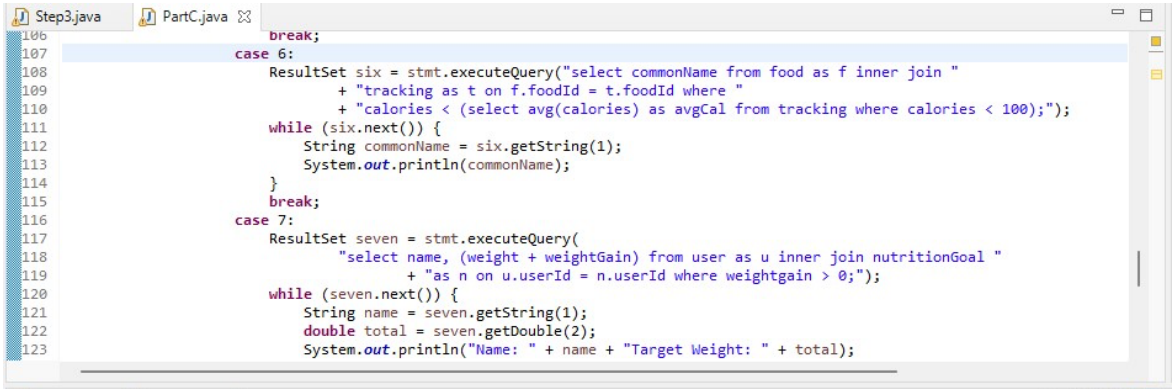
```
Console  Problems  Debug Shell
PartC [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe  (Dec 6, 2021, 8:09:27 PM)
5
Common Name: Moose Meat Raw (Alaska Native)Avg Calories: 103.0
Common Name: Seal Bearded (Oogruk) Meat Raw (Alaska Native)Avg Calories: 110.0
Common Name: Oil Bearded Seal (Oogruk) (Alaska Native)Avg Calories: 899.0
Common Name: Oopah (Tunicate) Whole Animal (Alaska Native)Avg Calories: 67.0
Common Name: Owl Horned Flesh Raw (Alaska Native)Avg Calories: 136.0
Common Name: Fish Salmon King Chinook Smoked And Canned (Alaska Native)Avg Calories: 150.0
Common Name: Fish Salmon King Chinook Smoked Brined (Alaska Native)Avg Calories: 430.0
Common Name: Duck Scoter White-Winged Meat (Alaska Native)Avg Calories: 84.0
Common Name: Sea Cucumber Yane (Alaska Native)Avg Calories: 56.0
Common Name: Squirrel Ground Meat (Alaska Native)Avg Calories: 111.0
Common Name: Tea Tundra Herb And Laborador Combination (Alaska Native)Avg Calories: 1.0
Common Name: Walrus Meat Dried (Alaska Native)Avg Calories: 251.0
Common Name: Deer (Venison) Sitka Raw (Alaska Native)Avg Calories: 116.0
Common Name: Whale Bowhead Subcutaneous Fat (Blubber) (Alaska Native)Avg Calories: 870.0
Common Name: Whale Bowhead Skin And Subcutaneous Fat (Muktuk) (Alaska Native)Avg Calories: 465.0
```



```
                        int age = four.getInt(4);
                        System.out.println("name: " + name + " " + "Height: " + height + " " + "Weight: " + weight
                                + " " + "Age: " + age);
                    }
                    break;
                case 5:
                    ResultSet five = stmt
                            .executeQuery("select commonName, calories from food as f inner join tracking as"
                                    + " t on f.foodId = t.foodId where carbs < 2; ");
                    while (five.next()) {
                        String commonName = five.getString(1);
                        double avgCal = five.getDouble(2);
                        System.out.println("Common Name: " + commonName + "Avg Calories: " + avgCal);
                    }
                    break;
                case 6:
                    ResultSet six = stmt.executeQuery("select commonName from food as f inner join "
                            + "tracking as t on f.foodId = t.foodId where "
```

```
Console  Problems  Debug Shell
PartC [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe  (Dec 6, 2021, 8:09:27 PM)
Common Name: Pork Cured Feet PickledAvg Calories: 140.0
Common Name: Lean Cured HamAvg Calories: 145.0
Common Name: Low Fat Cured HamAvg Calories: 178.0
Common Name: Pork Cured Ham Patties UnheatedAvg Calories: 315.0
Common Name: Pork Cured Ham Steak Boneless Extra Lean UnheatedAvg Calories: 122.0
Common Name: Pork Cured Ham Whole Separable Lean And Fat UnheatedAvg Calories: 246.0
Common Name: Pork Cured Ham Whole Separable Lean Only UnheatedAvg Calories: 147.0
Common Name: Pork Cured Separable Fat (From Ham And Arm Picnic) UnheatedAvg Calories: 579.0
Common Name: Pork Cured Separable Fat (From Ham And Arm Picnic) RoastedAvg Calories: 591.0
Common Name: Pork Cured Shoulder Arm Picnic Separable Lean And Fat RoastedAvg Calories: 280.0
Common Name: Pork Cured Shoulder Arm Picnic Separable Lean Only RoastedAvg Calories: 170.0
Common Name: Pork Fresh Variety Meats And By-Products Tail Cooked SimmeredAvg Calories: 396.0
Common Name: Pork Fresh Loin Center Loin (Chops) Bone-In Separable Lean Only Cooked Pan-FriedAvg Calories: 195.0
Common Name: Pork Fresh Loin Center Rib (Chops) Bone-In Separable Lean Only Cooked Pan-FriedAvg Calories: 211.0
Common Name: Pork Fresh Loin Blade (Chops) Bone-In Separable Lean And Fat Cooked Pan-FriedAvg Calories: 256.0
Common Name: Pork Cured Ham Extra Lean And Regular Canned UnheatedAvg Calories: 144.0
```

Case 6

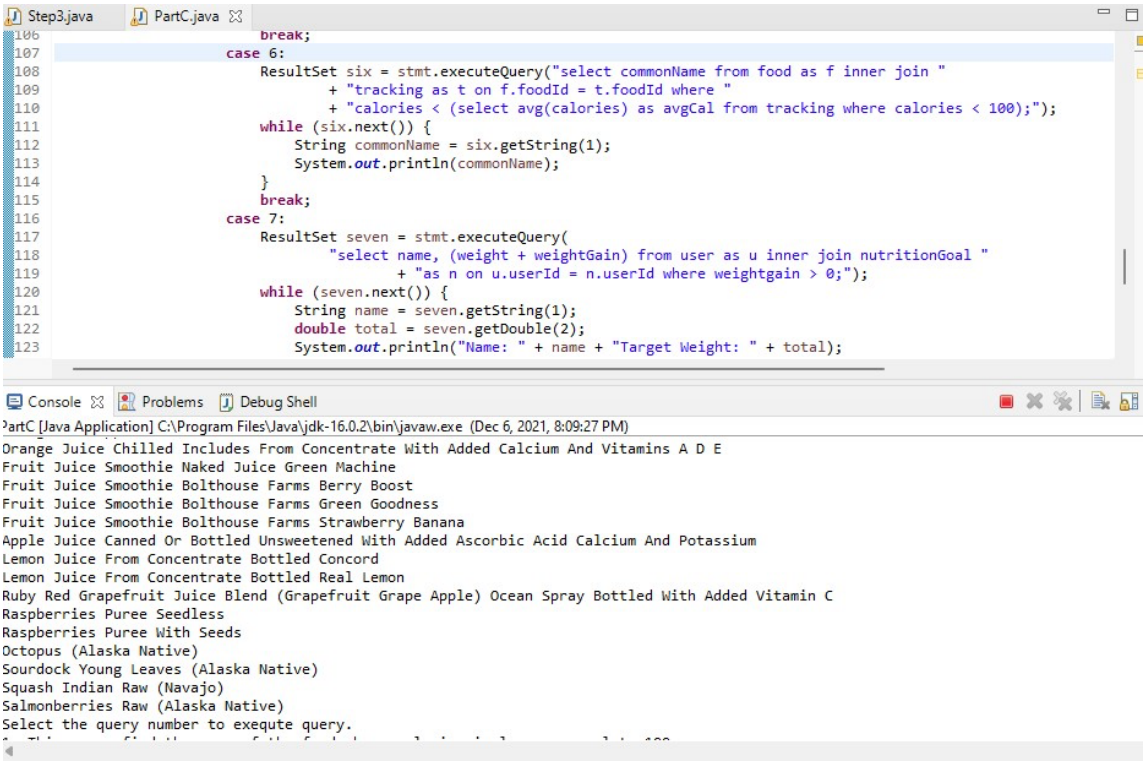


```
                    break;
                case 6:
                    ResultSet six = stmt.executeQuery("select commonName from food as f inner join "
                            + "tracking as t on f.foodId = t.foodId where "
                            + "calories < (select avg(calories) as avgCal from tracking where calories < 100);");
                    while (six.next()) {
                        String commonName = six.getString(1);
                        System.out.println(commonName);
                    }
                    break;
                case 7:
                    ResultSet seven = stmt.executeQuery(
                            "select name, (weight + weightGain) from user as u inner join nutritionGoal "
                                    + "as n on u.userId = n.userId where weightgain > 0;");
                    while (seven.next()) {
                        String name = seven.getString(1);
                        double total = seven.getDouble(2);
                        System.out.println("Name: " + name + "Target Weight: " + total);
```

Console | Problems | Debug Shell

PartC [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe (Dec 6, 2021, 8:09:27 PM)

```
6
Seaweed Canadian Cultivated Emi-Tsunomata Rehydrated
Sea Cucumber Yane (Alaska Native)
Tea Tundra Herb And Laborador Combination (Alaska Native)
Mush Blue Corn With Ash (Navajo)
Melon Banana (Navajo)
Chilchen (Red Berry Beverage) (Navajo)
Squash Indian Cooked Boiled (Navajo)
Pectin Liquid
Milk Buttermilk Fluid Cultured Reduced Fat
Tomato And Vegetable Juice Low Sodium
Chocolate-Flavored Drink Whey And Milk Based
Babyfood Juice Pear
Babyfood Juice Apple - Cherry
Milk Imitation Non-Soy
Lemons
```



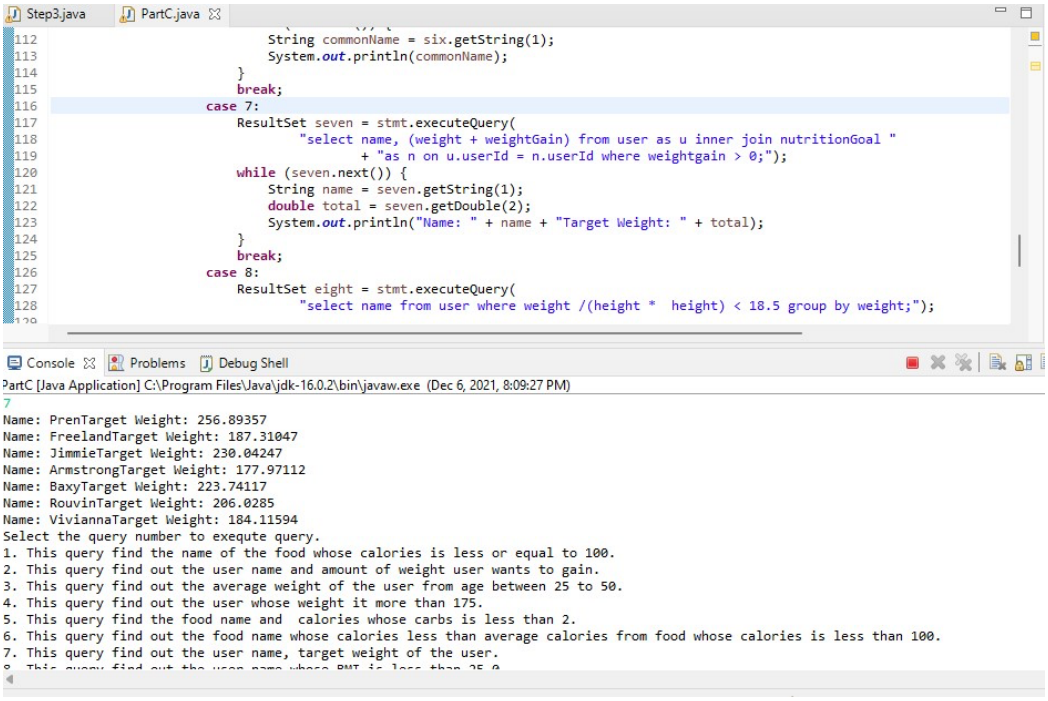
```
                    break;
                case 6:
                    ResultSet six = stmt.executeQuery("select commonName from food as f inner join "
                            + "tracking as t on f.foodId = t.foodId where "
                            + "calories < (select avg(calories) as avgCal from tracking where calories < 100);");
                    while (six.next()) {
                        String commonName = six.getString(1);
                        System.out.println(commonName);
                    }
                    break;
                case 7:
                    ResultSet seven = stmt.executeQuery(
                            "select name, (weight + weightGain) from user as u inner join nutritionGoal "
                                    + "as n on u.userId = n.userId where weightgain > 0;");
                    while (seven.next()) {
                        String name = seven.getString(1);
                        double total = seven.getDouble(2);
                        System.out.println("Name: " + name + "Target Weight: " + total);
```

Console | Problems | Debug Shell

PartC [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe (Dec 6, 2021, 8:09:27 PM)

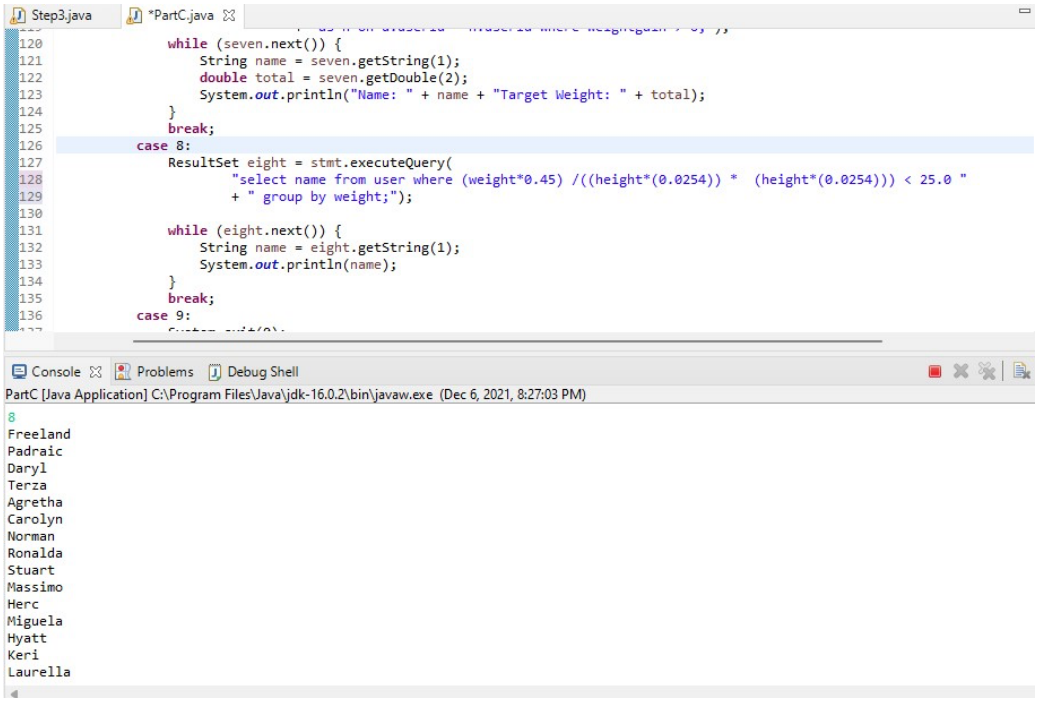
```
Orange Juice Chilled Includes From Concentrate With Added Calcium And Vitamins A D E
Fruit Juice Smoothie Naked Juice Green Machine
Fruit Juice Smoothie Bolthouse Farms Berry Boost
Fruit Juice Smoothie Bolthouse Farms Green Goodness
Fruit Juice Smoothie Bolthouse Farms Strawberry Banana
Apple Juice Canned Or Bottled Unsweetened With Added Ascorbic Acid Calcium And Potassium
Lemon Juice From Concentrate Bottled Concord
Lemon Juice From Concentrate Bottled Real Lemon
Ruby Red Grapefruit Juice Blend (Grapefruit Grape Apple) Ocean Spray Bottled With Added Vitamin C
Raspberries Puree Seedless
Raspberries Puree With Seeds
Octopus (Alaska Native)
Sourdock Young Leaves (Alaska Native)
Squash Indian Raw (Navajo)
Salmonberries Raw (Alaska Native)
Select the query number to exequte query.
```

Case 7

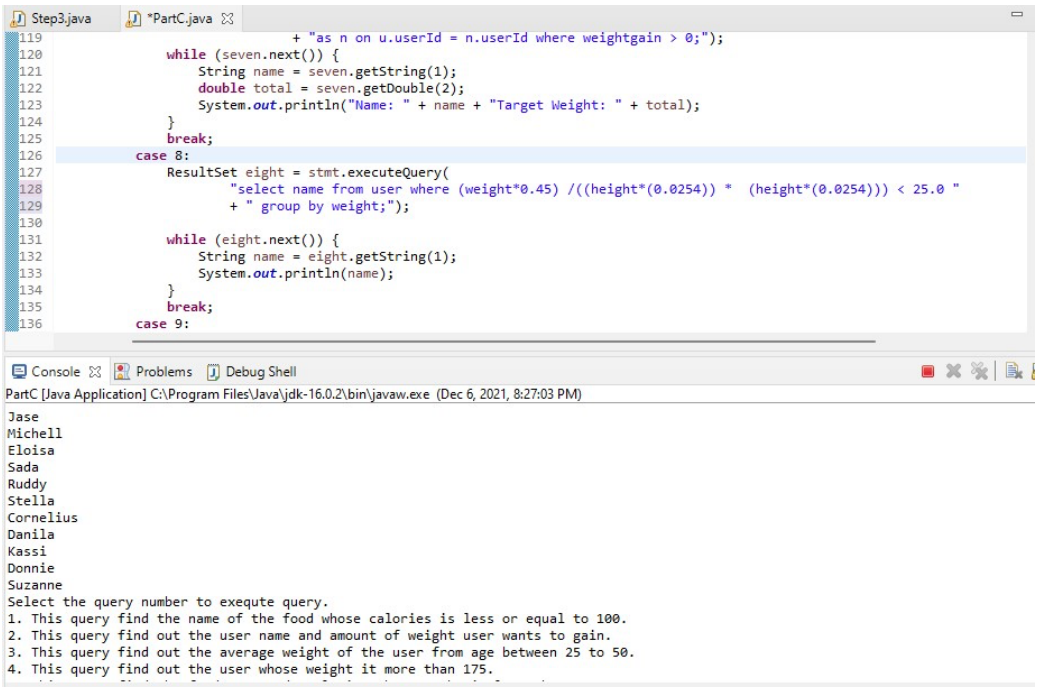


```
Step3.java  PartC.java
                            String commonName = six.getString(1);
                            System.out.println(commonName);
                        }
                        break;
                    case 7:
                        ResultSet seven = stmt.executeQuery(
                                "select name, (weight + weightGain) from user as u inner join nutritionGoal "
                                        + "as n on u.userId = n.userId where weightgain > 0;");
                        while (seven.next()) {
                            String name = seven.getString(1);
                            double total = seven.getDouble(2);
                            System.out.println("Name: " + name + "Target Weight: " + total);
                        }
                        break;
                    case 8:
                        ResultSet eight = stmt.executeQuery(
                                "select name from user where weight /(height *  height) < 18.5 group by weight;");

Console  Problems  Debug Shell
PartC [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe  (Dec 6, 2021, 8:09:27 PM)
7
Name: PrenTarget Weight: 256.89357
Name: FreelandTarget Weight: 187.31047
Name: JimmieTarget Weight: 230.04247
Name: ArmstrongTarget Weight: 177.97112
Name: BaxyTarget Weight: 223.74117
Name: RouvinTarget Weight: 206.0285
Name: ViviannaTarget Weight: 184.11594
Select the query number to exequte query.
1. This query find the name of the food whose calories is less or equal to 100.
2. This query find out the user name and amount of weight user wants to gain.
3. This query find out the average weight of the user from age between 25 to 50.
4. This query find out the user whose weight it more than 175.
5. This query find the food name and  calories whose carbs is less than 2.
6. This query find out the food name whose calories less than average calories from food whose calories is less than 100.
7. This query find out the user name, target weight of the user.
```

Case 8



```
Step3.java  *PartC.java
                while (seven.next()) {
                    String name = seven.getString(1);
                    double total = seven.getDouble(2);
                    System.out.println("Name: " + name + "Target Weight: " + total);
                }
                break;
            case 8:
                ResultSet eight = stmt.executeQuery(
                        "select name from user where (weight*0.45) /((height*(0.0254)) *  (height*(0.0254))) < 25.0 "
                        + " group by weight;");

                while (eight.next()) {
                    String name = eight.getString(1);
                    System.out.println(name);
                }
                break;
            case 9:

Console  Problems  Debug Shell
PartC [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe  (Dec 6, 2021, 8:27:03 PM)
8
Freeland
Padraic
Daryl
Terza
Agretha
Carolyn
Norman
Ronalda
Stuart
Massimo
Herc
Miguela
Hyatt
Keri
Laurella
```



Notes:

1. In many cases, there are multiple lines record available. To display here, we need to take multiple screen shots which make document complicated. That's why we just took maximum two screen shots of the case.
2. We uploaded two java files.in which step3.java is used to insert the queries from csv files to database where as PartC.java is used to generate output from the queries.

5. On the last page, list the edits/modifications/adjustments you made (if any) to the ER, Relational Schemas,
or Queries you submitted in step2.

We changed case number 8 because we put the data from web and they does not have any BMI which less than or equal to 18.5. Therefore we increased to 25. Also, we modified the query to convert weight lb to kg and height inch to meter.