DT Big Data Analysis Solution

Qu 1 :-

Write a Program to accept 10 Student Details Like..

1. StudentID - String - StudentID must start with S and followed by 3 digit.

2. StudentName - String

3. StudentAge - int - Value entered by user must not be greater than 100

4. StudentCity - String -

5. StudentCountry - String - Only India is allowed

Task :-

1. Store all the data inside txt file.

2. Read the data from txt file and count the number of students whose age is greater than 21.

3. Read the data from the file and display the details of all the students whose Names are starting with "A".

```
import java.util.*;
import java.io.*;
import java.util.regex.Pattern;

public class Qu1 {
 static ArrayList<Student> std = new ArrayList<Student>();
 static void enterData() {
  Scanner s = new Scanner(System.in);
  for(int i=0; i<10; i++) {
   System.out.println("Enter Student Id: ");
   String id = s.next();
   while(! Pattern.matches("s[0-9][0-9][0-9]", id))
   {
    System.out.println("Invalid Id. Enter Again: ");
    id = s.next();
   }
   System.out.println("Enter Student Name: ");
   String name = s.next();
   System.out.println("Enter Age: ");
   int age = s.nextInt();
   while(age > 100) {
    System.out.println("Invalid Age. Enter Again: ");
    age = s.nextInt();
   }
   System.out.println("Enter City: ");
   String city = s.next();
   String coun = "India";

   std.add(new Student(id, name, age, city, coun));
  }
  s.close();
 }

 static void writeData() {
  String text="";
```

```
   try {
    BufferedWriter bw = new BufferedWriter(new FileWriter(new File("D:/student.txt"), true));
    for(Student st: std) {
     text = st.sid + "," + st.sname + "," + st.age + "," + st.city + "," + st.coun;
     bw.write(text);
     bw.newLine();
    }

    bw.close();
   } catch (Exception e) {
   }
  }



  static void readData() {
   int count = 0;
   try (BufferedReader br = new BufferedReader(new FileReader("D:/student.txt"))) {
        String line;
        System.out.println("_____________________________________________________________\nDetails of
students whose name starts with 'A'\n_____________________________________________________________");
        System.out.println("Student ID\tName\tAge\tCity\tCountry\n");
        while ((line = br.readLine()) != null) {
         String[] linedata = line.split(",");
         if(Integer.parseInt(linedata[2]) > 21) {
           count ++;
          }
          if(Pattern.matches("a[a-z]*|A[a-z]*", linedata[1])) {
          System.out.println(linedata[0]+"\t\t"+linedata[1]+"\t"+linedata[2]+"\t"+linedata[3]+"
\t"+linedata[4]);
         }

        }
        System.out.println("_____________________________________________________________");
        System.out.println("No of students with age greater than 21 are "+count);
   }


   catch(IOException e) {
    System.out.println(e.getMessage());
   }


  }

  public static void main(String[] args) {
   enterData();
   writeData();
   readData();
  }

}
```

```java
class Student{
 String sid;
 String sname;
 int age;
 String city;
 String coun = "India";
 public Student(String sid, String sname, int age, String city, String coun) {
  this.sid = sid;
  this.sname = sname;
  this.age = age;
  this.city = city;
  this.coun = coun;
 }
}
```

Qu 2 :-

Write a Java program to accept movie details like..

1. MovieID

2. MovieName

3. MovieDirector

4. MovieActor

5. MovieRating

6. MovieType

7. MovieReleaseDate

1. Accept 50 details and store the data inside the file using proper delimeter.

2. Read the data from file and perform below analysis.

Display the list of all the movies released in year 2015.

Display the list of all the movies having type as "U".

Display the list of all the movies having rating greater than 4 star

Display the list of all the movies having Actor as "Khans"

```
import java.util.*;
import java.io.*;
import java.time.LocalDate;

public class Qu2 {
 static ArrayList<String> rating4p = new ArrayList<String>();
 static ArrayList<String> typeU = new ArrayList<String>();
 static ArrayList<String> actorKhans = new ArrayList<String>();
 static ArrayList<String> released2015 = new ArrayList<String>();

 static ArrayList<Movies> mvi = new ArrayList<Movies>();
 static void enterData() {
  String name, director, actor, type, date;
  int id, rating;
  Scanner s = new Scanner(System.in);
  for(int i=0; i<50; i++) {
   System.out.println("Enter Movie Id: ");
   id = s.nextInt();
   s.nextLine();
   System.out.println("Enter Movie Name: ");
   name = s.nextLine();
   System.out.println("Enter Movie Directors Name: ");
   director = s.nextLine();
   System.out.println("Enter Actors Name: ");
   actor = s.nextLine();
   System.out.println("Enter Rating For The Movie: ");
   rating = s.nextInt();
   while(rating > 10) {
    System.out.println("Invalid Rating. Enter Again: ");
    rating = s.nextInt();
   }
```

```
    System.out.println("Enter Movie Type(U / UA / A): ");
    type = s.next();
    System.out.println("Enter Movie Release Date(dd/mm/yyyy): ");
    date = s.next();

    mvi.add(new Movies(id, name, director, actor, rating, type, date));
   }
   s.close();
  }

  static void writeData() {
   String text="";
   try {
    BufferedWriter bw = new BufferedWriter(new FileWriter(new File("D:/Movies.txt")));
    for(Movies m: mvi) {
     text = m.id + ", " + m.name + ", "+m.director + ", "+m.actor + ", "+m.rating + ", "+m.type +
", "+m.releaseDate;
     bw.write(text);
     bw.newLine();
    }

    bw.close();
   }
   catch (Exception e) {
    e.printStackTrace();
   }
  }



  static void readData() {
   int count = 0;
   try (BufferedReader br = new BufferedReader(new FileReader("D:/Movies.txt"))) {
       String line;
       while ((line = br.readLine()) != null) {
        String[] linedata = line.split(", ");


        String[] d = linedata[6].split("-");
        LocalDate releaseDate = LocalDate.of(Integer.parseInt(d[0]), Integer.parseInt(d[1]),
Integer.parseInt(d[2]));
        if(releaseDate.getYear() == 2015) {
         released2015.add(line);
         }

        if(linedata[5].toLowerCase().equals("u")) {
         typeU.add(line);
        }
        if(Integer.parseInt(linedata[4])>4) {
         rating4p.add(line);
        }
        String[] actor = linedata[3].split(" ");
        if(actor[actor.length-1].toLowerCase().equals("khan")) {
         actorKhans.add(line);
        }

       }
   }


   catch(IOException e) {
```

```
   System.out.println(e.getMessage());
  }
  Iterator<String> itr = released2015.iterator();
  while(itr.hasNext()) {
   if (count==0) {

System.out.println("_____________________________________________________________________\n");
    System.out.println("Movies released in 2015:");
    count++;
   }
   System.out.println(itr.next());
  }

  count=0;
  itr = typeU.iterator();
  while(itr.hasNext()) {
   if (count==0) {

System.out.println("_____________________________________________________________________\n");
    System.out.println("Movies with U certificate:");
    count++;
   }
   System.out.println(itr.next());
  }

  count = 0;
  itr = rating4p.iterator();
  while(itr.hasNext()) {
   if (count==0) {

System.out.println("_____________________________________________________________________\n");
    System.out.println("Movies with rating 4+:");
    count++;
   }
   System.out.println(itr.next());
  }

  count = 0;
  itr = actorKhans.iterator();
  while(itr.hasNext()) {
   if (count==0) {

System.out.println("_____________________________________________________________________\n");
    System.out.println("Movies with khans:");
    count++;
   }
   System.out.println(itr.next());
  }
}
```

```
  public static void main(String[] args) {
    enterData();
    writeData();
    readData();
  }
}

class Movies{
  int id, rating;
  String name, director, actor, type;
  LocalDate releaseDate;
  public Movies(int id, String name, String director, String actor, int rating, String type,
String date) {
    this.id = id;
    this.name = name;
    this.director = director;
    this.actor = actor;
    this.rating = rating;
    this.type = type;
    try {
      String[] d = date.split("/");
      this.releaseDate = LocalDate.of(Integer.parseInt(d[2]), Integer.parseInt(d[1]),
Integer.parseInt(d[0]));
    }
    catch(Exception e) {
      e.printStackTrace();
    }
  }
}
```

Qu 3 :-

Write a Java program to accept Transaction details like..

1. CustomerID

2. ProductID

3. ProductPrice

4. Quantity

5. TotalPrice

6. DateOfPurchase

7. productType

8. DiscountPercentage

1. Accept 50 details and store the data inside the file using proper delimeter.

2. Read the data from file and perform below analysis.

Customer did highest purchase in a day

Total Purchase made by customer in one year.

Display the details of all the products where discount is >25%

```
import java.util.*;
import java.io.*;
import java.time.LocalDate;

public class Qu3 {
 static ArrayList<Transaction> trans = new ArrayList<Transaction>();
 static String[] tot;
 static ArrayList<String> disc = new ArrayList<String>();
 static HashMap<String, String[]> maxDay = new HashMap<String, String[]>();
 static HashMap<String[], String> totalYear = new HashMap<String[], String>();

 static void enterData() {
  String cid, pid, ptype, date;
  int quan;
  float price,  disc;

  Scanner s = new Scanner(System.in);
  for(int i=0; i<50; i++) {
   System.out.println("Enter Customer Id: ");
   cid = s.next();
   System.out.println("Enter Product Id: ");
   pid = s.next();
   System.out.println("Enter Product Price: ");
   price = s.nextFloat();
   System.out.println("Enter Quantity: ");
   quan = s.nextInt();
   System.out.println("Enter Transaction Date(dd/mm/yyyy): ");
   date = s.next();
   System.out.println("Enter Product Type: ");
   ptype = s.next();
   System.out.println("Enter Discount Percentage: ");
```

```
    disc = s.nextFloat();

    trans.add(new Transaction(cid, pid, price, quan, date, ptype, disc));
   }
   s.close();
  }

  static void writeData() {
   String text="";
   try {
    BufferedWriter bw = new BufferedWriter(new FileWriter(new File("D:/Transaction.txt")));
    for(Transaction t: trans) {
     text = t.cid + ", " + t.pid + ", "+ t.price + ", "+ t.quan + ", "+ t.total + ", "+ t.date +
", "+ t.ptype+", "+ t.disc;
     bw.write(text);
     bw.newLine();
    }

    bw.close();
   }
   catch (Exception e) {
    e.printStackTrace();
   }
  }

  static void readData() {
   int count = 0;
   try (BufferedReader br = new BufferedReader(new FileReader("D:/Transaction.txt"))) {
       String line;
       while ((line = br.readLine()) != null) {
        String[] linedata = line.split(", ");
        p1(line);
        p2(line);

        p3(line);
       }
   }
   catch(IOException e) {
    System.out.println(e.getMessage());
   }

  System.out.println("_____________________________________________________________________\nDatewi
se Maximum Sale\n_____________________________________________________________________");
   System.out.println("Date\t    Customer ID\t   Amount");
   for(Map.Entry c: maxDay.entrySet()){
    String[] val = (String[]) c.getValue();
    System.out.println(c.getKey() +"\t"+ val[0] +"\t    "+ val[1]);
   }


  System.out.println("_____________________________________________________________________\nYearwi
se Total Purchases\n_____________________________________________________________________");
   System.out.println("Year\t    Customer ID\t   Amount");
   for(Map.Entry c: totalYear.entrySet()){
    String[] val = (String[]) c.getKey();

    System.out.println(val[0] +"\t\t"+ val[1] +"\t\t    "+ c.getValue());
   }


  System.out.println("_____________________________________________________________________\nProduc
```

```
ts with discount greater than
25%\n_____");
   Iterator itr = disc.iterator();
   while(itr.hasNext()) {
    System.out.println(itr.next());
   }

  }

  static void p1(String line) {
     String date = line.split(", ")[5];
     if (dateExists(date)) {
      float oldPrice = Float.parseFloat(maxDay.get(date)[1]);
      float newPrice = Float.parseFloat(line.split(",")[4]);
      if(oldPrice < newPrice) {
       String[] val = new String[2];
       val[0] = line.split(", ")[0];
       val[1] = line.split(", ")[4];
       maxDay.put(date, val);
      }

     }
     else {
      String[] val = new String[2];
      val[0] = line.split(", ")[0];
      val[1] = line.split(", ")[4];
      maxDay.put(date, val);

     }
  }

  static void p2(String line) {
   String year = (line.split(", ")[5]).split("-")[0];
   String[] key = {year, line.split(", ")[0]};
   float newValue = Float.parseFloat(line.split(", ")[4]);
   addTotal(key, newValue);


  }

  static void p3(String line) {
   if(Float.parseFloat(line.split(", ")[7]) > 25.0) {
    disc.add(line);
   }
  }

  static boolean dateExists(String date) {
   for(Map.Entry<String, String[]> d: maxDay.entrySet()){
    if(date.equals(d.getKey())) {
     return true;
    }
   }
   return false;
  }

  static void addTotal(String[] key, float value) {
   int f = 0;
   for(Map.Entry<String[], String> k: totalYear.entrySet()){
    if(Arrays.equals(key, k.getKey())) {
     k.setValue(String.valueOf(value + Float.parseFloat(k.getValue())));
     f = 1;
```

```
    }
   }
   if(f==0) {
    totalYear.put(key, String.valueOf(value));
   }


  }

  public static void main(String[] args) {

   enterData();
   writeData();
   readData();
  }

}

class Transaction{

  String cid, pid, ptype;
  int quan;
  float price, total, disc;
  LocalDate date;


  public Transaction(String cid, String pid, float price, int quan, String date, String ptype,
float disc) {
   this.cid = cid;
   this.pid = pid;
   this.price = price;
   this.quan = quan;
   this.total = price * quan;
   try {
    String[] d = date.split("/");
    this.date = LocalDate.of(Integer.parseInt(d[2]), Integer.parseInt(d[1]),
Integer.parseInt(d[0]));
   }
   catch(Exception e) {
    e.printStackTrace();
   }
   this.ptype = ptype;
   this.disc = disc;
  }
}
```

Qu 4 :-

Write a java program to accept customer details like:-

1. CustomerId

2. CustomerName

3. CustomerGender

4. CustomerAge

5. CustomerAddress

6. CustomerCity

7. CustomerCountry

8. CustomerPin

9. CustomerPhone

10. CustomerDOB

1. Accept 50 details and store the data inside the file using proper delimeter.

2. Read the data from file and perform below analysis.

Display Total male and female customer

Display total customer country wise

Display total customer whose age is greater than 18

Display total customer who does not have phone number

```
import java.util.*;
import java.io.*;
import java.time.LocalDate;

public class Qu4 {
static ArrayList<Customer> cus = new ArrayList<Customer>();
static HashMap<String, Integer> countryCount = new HashMap<String, Integer>();

 static void enterData() {
  String cid, name, gender, address, city, coun, dob;
  int age, pin;
  long no;

  Scanner s = new Scanner(System.in);
  for(int i=0; i<50; i++) {
   System.out.println("Enter Customer Id: ");
   cid = s.next();
   System.out.println("Enter Customer Name: ");
   name = s.next();
   System.out.println("Enter Customers Gender: ");
   gender = s.next();
   System.out.println("Enter Age: ");
   age = s.nextInt();
   s.nextLine();
```

```
    System.out.println("Enter Address: ");
    address = s.nextLine();
    System.out.println("Enter City: ");
    city = s.nextLine();
    System.out.println("Enter Country: ");
    coun = s.nextLine();
    System.out.println("Enter Customer Pin: ");
    pin = s.nextInt();
    s.nextLine();
    System.out.println("Enter Customers Date of Birth(dd/mm/yyyy): ");
    dob = s.next();
    System.out.println("Enter Phone No: ");
    if(s.hasNextLong()) {
     no = s.nextLong();
    }
    else {
     no = 0;
    }




    cus.add(new Customer(cid, name, gender, age, address, city, coun, pin, no, dob));
   }
   s.close();
  }

  static void writeData() {
   String text="";
   try {
    BufferedWriter bw = new BufferedWriter(new FileWriter(new File("D:/Customer.txt"), true));
    for(Customer c: cus) {
     text = c.cid + ", " + c.name + ", " + c.gender+ ", " + c.age+ ", " + c.address + ", " +
c.city + ", " + c.coun + ", " +c.pin + ", " + c.no+ ", " + c.dob;
     bw.write(text);
     bw.newLine();
    }

    bw.close();
   }
   catch (Exception e) {
    e.printStackTrace();
   }
  }

  static void readData() {
   int cm=0, cf=0, ca=0, cnno=0;
   String country = "";
   try (BufferedReader br = new BufferedReader(new FileReader("D:/Customer.txt"))) {
       String line;
       while ((line = br.readLine()) != null) {
        String[] linedata = line.split(", ");

        if(linedata[2].toLowerCase().startsWith("m")) {
         cm++;
        }
        else if(linedata[2].toLowerCase().startsWith("f")) {
         cf++;
        }

        if(Integer.parseInt(linedata[3]) > 18) {
```

```
          ca++;
         }
         long no = Long.parseLong(linedata[8]);
         if(no == 0) {
          cnno++;
         }

         country = linedata[6].toLowerCase();


         if(!countryExists(country)) {
          countryCount.put(country, 1);

         }
         else {
          int count = countryCount.get(country);
          countryCount.replace(country, count, (count+1));
         }

        }
    }
    catch(IOException e) {
     System.out.println(e.getMessage());
    }
    System.out.println("No of male customers = "+cm);
    System.out.println("No of female customers = "+cf);
    System.out.println("No of customers with age greater than 18 = "+ca);
    System.out.println("No of customers with no phone no = "+cnno);
    System.out.println("Countrywise customer count: ");
      for(Map.Entry c: countryCount.entrySet()){
     System.out.println(c.getKey() +"\t"+ c.getValue());
    }


  }
  static boolean countryExists(String country) {
   for(Map.Entry c: countryCount.entrySet()){
    if(country.equals(c.getKey())) {
     return true;
    }
   }
   return false;
  }

  public static void main(String[] args) {
   enterData();
   writeData();
   readData();
  }
}

class Customer{
 String cid, name, gender, address, city, coun;
 int age, pin;
 long no;
 LocalDate dob;
 public Customer(String cid, String name,String gender, int age, String address, String city,
String coun, int pin, long no, String dob) {
   this.cid = cid;
   this.name = name;
   this.gender = gender;
```

```
   this.age = age;
   this.address = address;
   this.city = city;
   this.coun = coun;
   this.pin = pin;
   this.no = no;
   try {
    String[] d = dob.split("/");
    this.dob = LocalDate.of(Integer.parseInt(d[2]), Integer.parseInt(d[1]),
Integer.parseInt(d[0]));
   }
   catch(Exception e) {
    e.printStackTrace();
   }
  }
}
```

Qu 5 :-

Write a program to accept the customer details like :-

1. CustomerId -- String --
2. CustomerName -- String --
3. CustomerGender -- String --
4. CustomerAge -- int --
5. CustomerAddress -- String --
6. CustomerCity -- String --
7. CustomerCountry -- String --
8. CustomerPin -- String --
9. CustomerPhone -- String --
10. CustomerAreaOfInterest --String--
11. CustomerDOB -- Date --

Validate the data:-

1. CustomerId-must not be more than 4 character and must start with 'C'
2. CustomerName -- data must not be blank or null
3. CustomerGender -- Should allow only Male and Female
4. CustomerPhone should allow null value if user is not entering any data
5. CustomerAreaOfInterest -- user must enter 2 value
6. CustomerDOB --should not be more than 8 characters

Analysis

1. Add at least 10 records and store the data in file seperated by delimeter like comma and space
2. After storing the data kindly read the data from file and do below analysis
   1. Display total male and female customers
   2. Display the total customer country wise
   3. Display the total customer who like Traveling
   4. Display total customer who like reading
   5. Display total customer who is above 18 years of age
   6. Display total customer who do not have phone number.

```
import java.util.*;
import java.io.*;
import java.time.LocalDate;
import java.util.regex.Pattern;

public class Qu5 {
 static ArrayList<Customers> cust = new ArrayList<Customers>();
 private static HashMap<String, Integer> countryCount = new HashMap<String, Integer>();

 static void enterData() {
  Scanner s = new Scanner(System.in);
  String cid, cname, gender, cadd, ccity, ccoun, cpin, cphone, caoi1, caoi2, dob;
  int cage;
  for(int i=0; i<10; i++) {
   System.out.println("Enter Customer Id: ");
   cid = s.nextLine();
   while(! Pattern.matches("c...", cid))
   {
    System.out.println("Invalid Id. Enter Again: ");
    cid = s.nextLine();
   }
   System.out.println("Enter Customer Name: ");
   cname = s.nextLine();
   while(cname.equals(""))
   {
    System.out.println("Invalid Name. Enter Again: ");
    cname = s.nextLine();
   }
   System.out.println("Enter Customers Gender(Male/Female): ");
   gender = s.nextLine().toLowerCase();
   while(!(gender.equals("male") || gender.equals("female")))
   {
    System.out.println("Invalid Gender. Enter Again: ");
    gender = s.nextLine();
   }
   System.out.println("Enter Age: ");
   cage = s.nextInt();
   s.nextLine();
   System.out.println("Enter Address: ");
   cadd = s.nextLine();
   System.out.println("Enter City: ");
   ccity = s.nextLine();
   System.out.println("Enter Country: ");
   ccoun = s.nextLine().toLowerCase();
   System.out.println("Enter Customer Pin: ");
   cpin = s.nextLine();
   System.out.println("Enter Customers Date of Birth(dd/mm/yyyy): ");
   dob = s.nextLine();
   System.out.println("Enter Phone No: ");
   cphone = s.nextLine();
   if(cphone.equals("")) {
    cphone = "null";
   }
   System.out.println("Enter First Area of Interest: ");
   caoi1 = s.nextLine().toLowerCase();
   System.out.println("Enter Second Area of Interest: ");
   caoi2 = s.nextLine().toLowerCase();
```

```
    cust.add(new Customers(cid, cname, gender, cage, cadd, ccity, ccoun, cpin, cphone, caoi1,
caoi2, dob));
   }
   s.close();
  }

  static void writeData() {
   String text="";
   try {
    BufferedWriter bw = new BufferedWriter(new FileWriter(new File("D:/Customers.txt"), true));
    for(Customers c: cust) {
     text = c.cid + "; " + c.cname + "; " + c.gender+ "; " + c.cage+ "; " + c.cadd + "; " +
c.ccity + "; " + c.ccoun + "; " +c.cpin + "; " + c.cphone+ "; " + c.caoi1+ "; " + c.caoi2+ "; " +
c.dob;
     bw.write(text);
     bw.newLine();
    }

    bw.close();
   }
   catch (Exception e) {
    e.printStackTrace();
   }
  }

  static void readData() {
   int cm=0, cf=0, ct=0, cr=0, ca = 0, cnno=0;
   String country = "";
   try (BufferedReader br = new BufferedReader(new FileReader("D:/Customers.txt"))) {
       String line;
       while ((line = br.readLine()) != null) {
        String[] linedata = line.split("; ");
        if(linedata[2].equals("male")) {
         cm++;
        }
        else if(linedata[2].equals("female")) {
         cf++;
        }

        country = linedata[6].toLowerCase();

        if(!countryExists(country)) {
         countryCount.put(country, 1);

        }
        else {
         int count = countryCount.get(country);
         countryCount.replace(country, count, (count+1));
        }

        if(linedata[9].equals("travelling") || linedata[10].equals("travelling")) {
         ct++;
        }

        if(linedata[9].equals("reading") || linedata[10].equals("reading")) {
         cr++;
        }

        if(Integer.parseInt(linedata[3]) > 18) {
         ca++;
        }
```

```
        if(linedata[8].equals("null")) {
          cnno++;
        }

       }
    }
    catch(Exception e) {
     e.printStackTrace();
    }

    System.out.println("Total no of male customers: "+cm);
    System.out.println("Total no of female customers: "+cf);
    System.out.println("Countrywise customer count: ");
    for(Map.Entry c: countryCount.entrySet()){
       System.out.println(c.getKey() +"\t"+ c.getValue());
    }
    System.out.println("Total no of customers who like Traveling: "+ct);
    System.out.println("Total no of customers who like Reading: "+cr);
    System.out.println("No of customers with age 18 above: "+ca);
    System.out.println("Total no of customer who do not have phone number: "+cnno);

  }

  static boolean countryExists(String country) {
    for(Map.Entry c: countryCount.entrySet()){
      if(country.equals(c.getKey())) {
        return true;
      }
    }
    return false;
  }

  public static void main(String[] args) {
    enterData();
    writeData();
    readData();
  }

}

class Customers{
  String cid, cname, gender, cadd, ccity, ccoun, cpin, cphone, caoi1, caoi2;
  int cage;
  LocalDate dob;

  public Customers(String cid, String cname, String gender, int cage, String cadd, String ccity,
String ccoun, String cpin, String cphone, String caoi1, String caoi2, String dob) {
    this.cid = cid;
    this.cname = cname;
    this.gender = gender;
    this.cage = cage;
    this.cadd = cadd;
    this.ccity = ccity;
    this.ccoun = ccoun;
    this.cpin = cpin;
    this.cphone = cphone;
    this.caoi1 = caoi1;
    this.caoi2 = caoi2;
    try {
      String[] d = dob.split("/");
```

```
    this.dob = LocalDate.of(Integer.parseInt(d[2]), Integer.parseInt(d[1]),
Integer.parseInt(d[0]));
   }
   catch(Exception e) {
    e.printStackTrace();
   }
  }
}
```