--------------------------------DBHandler.java---------------------------------

import java.sql.Connection;

import java.io.FileInputStream;

import java.io.IOException;

import java.sql.\*;

import java.util.Properties;

import java.io.FileNotFoundException;

public class DBHandler {

public Connection establishConnection() throws ClassNotFoundException, SQLException, FileNotFoundException {

Connection con = null;

Properties props = new Properties();

// this try block reads the db Properties file and establishConnection.

try{

FileInputStream fis = new FileInputStream("src/db.properties");

props.load(fis);

Class.forName(props.getProperty("db.classname"));

con = DriverManager.getConnection(props.getProperty("db.url"),props.getProperty("db.username"),props.getProperty("db.password"));

}

catch(IOException e){

e.printStackTrace();

}

return con;

//fill code here

}

}

-----------------------------------ElectricityBill.java------------------------------

*public* *class* ElectricityBill {

*private* *String* consumerNumber;

*private* *String* consumerName;

*private* *String* consumerAddress;

*private* *int* unitsConsumed;

*private* *double* billAmount;

*public* *String* getConsumerNumber() {

        return consumerNumber;

    }

*public* *void* setConsumerNumber(*String* *consumerNumber*) {

*this*.consumerNumber = consumerNumber;

    }

*public* *String* getConsumerName() {

        return consumerName;

    }

*public* *void* setConsumerName(*String* *consumerName*) {

*this*.consumerName = consumerName;

    }

*public* *String* getConsumerAddress() {

        return consumerAddress;

    }

*public* *void* setConsumerAddress(*String* *consumerAddress*) {

*this*.consumerAddress = consumerAddress;

    }

*public* *int* getUnitsConsumed() {

        return unitsConsumed;

    }

*public* *void* setUnitsConsumed(*int* *unitsConsumed*) {

*this*.unitsConsumed = unitsConsumed;

    }

*public* *double* getBillAmount() {

        return billAmount;

    }

*public* *void* setBillAmount(*double* *billAmount*) {

*this*.billAmount = billAmount;

    }

*//Write the required business logic as expected in the question description*

*public* *void* calculateBillAmount() {

*// method for calaculating the bill amount.*

*int* units = unitsConsumed;

*double* bill = 0;

        if(units <= 100){

            bill = 0;

        }

        if(units > 100 && units <= 300){

            bill = (units-100) \* 1.5;

        }

        if(units > 300 && units <= 600){

            bill = 200 \* 1.5 + (units-300) \* 3.5;

        }

        if(units > 600 && units <= 1000){

            bill = 200 \* 1.5 + 300 \* 3.5 + (units-600) \* 5.5;

        }

        if(units > 1000){

            bill = 200 \* 1.5 + 300 \* 3.5 + 400 \* 5.5 + (units-1000) \* 7.5;

        }

        setBillAmount(bill);

*//fill the code*

    }

}

-----------------------------ElectricityBoard.java-------------------------------

import *java.util.List*;

import *java.util.\**;

import *java.io.FileReader*;

import *java.io.File*;

import *java.io.BufferedReader*;

import *java.io.FileNotFoundException*;

import *java.io.IOException*;

import *java.util.regex.Pattern*;

import *java.sql.SQLException*;

import *java.sql.Connection*;

import *java.sql.PreparedStatement*;

*public* *class* ElectricityBoard {

*//write the required business logic methods as expected in the question description*

*public* *void* addBill(*List*<*ElectricityBill*> *billList*) {

*DBHandler* db =  new DBHandler();

        try(*Connection* con = db.establishConnection()){

*PreparedStatement* stmt = con.prepareStatement("insert into ElectricityBill values(?,?,?,?,?);");

*// for loop to insert the values into the table*

            for(*ElectricityBill* obj : billList){

                stmt.setString(1,obj.getConsumerNumber());

                stmt.setString(2,obj.getConsumerName());

                stmt.setString(3,obj.getConsumerAddress());

                stmt.setInt(4,obj.getUnitsConsumed());

                stmt.setDouble(5,obj.getBillAmount());

                stmt.execute();

            }

        }

        catch(*ClassNotFoundException* *e*){

            e.printStackTrace();

        }

        catch(*FileNotFoundException* *e*){

            e.printStackTrace();

        }

        catch(*SQLException* *e*){

            e.printStackTrace();

        }

*//fill the code*

    }

*public* *List*<*ElectricityBill*> generateBill(*String* *filePath*) {

*List* <*ElectricityBill*> list = new *ArrayList*<>();

*File* f = new File (filePath);

*// this try block is for opening and reading the file*

        try(*BufferedReader* br = new BufferedReader(new FileReader(f)))

        {

*String* line = null;

            while((line = br.readLine())!= null)

            {

*String* records[] = null;

*String* consumerNumber = "";

*String* consumerName = "";

*String* consumerAddress = "";

*int* unitsConsumed = 0;

                records = line.split(",");

                consumerNumber = records[0];

                consumerName = records[1];

                consumerAddress = records[2];

                unitsConsumed = Integer.parseInt(records[3]);

*//this try block checks for the validated consumerNumber*

                try{

                    if(validate(consumerNumber)){

*ElectricityBill* obj = new ElectricityBill();

                        obj.setConsumerNumber(consumerNumber);

                        obj.setConsumerName(consumerName);

                        obj.setConsumerAddress(consumerAddress);

                        obj.setUnitsConsumed(unitsConsumed);

                        obj.calculateBillAmount();

                        list.add(obj);

                    }

                }

                catch(*InvalidConsumerNumberException* *e*){

                    System.out.println(e.getMessage());

                }

            }

        }

        catch(*FileNotFoundException* *e*){

            e.printStackTrace();

        }

        catch(*IOException* *e*){

            e.printStackTrace();

        }

    return list;

*//fill the code*

    }

*public* *boolean* validate(*String* *consumerNumber*) *throws* *InvalidConsumerNumberException* {

*// method for validating the consumerNumber*

*boolean* isValid = Pattern.matches("^[0][0-9]{9}" , consumerNumber);

            if(!isValid){

                throw new InvalidConsumerNumberException("Invalid Consumer Number");

            }

            return true;

*//fill the code*

    }

}

-------------------------------------------Main.java-----------------------------------

import *java.util.\**;

import *java.util.List*;

import *java.util.ArrayList*;

*public* *class* Main {

*public* *static* *void* main(*String*[] *args*) {

*Scanner* sc= new Scanner(System.in);

*String* filePath = "src/ElectricityBill.txt";

*List*<*ElectricityBill*> list = new *ArrayList*<>();

*ElectricityBoard* eb = new ElectricityBoard();

       list = eb.generateBill(filePath);

       for(*ElectricityBill* obj: list){

           System.out.println(obj.getConsumerNumber() + " " + obj.getConsumerName() + " " + obj.getBillAmount());

       }

       eb.addBill(list);

       System.out.println("Successfully Inserted");

       sc.close();

*//fill your code here*

   }

}

---------------------InvalidConsumerNumberException.java---------

*//make the required changes to this class so that InvalidConsumerNumberException is of type exception.*

*public* *class* InvalidConsumerNumberException *extends* *Exception*{

*public* InvalidConsumerNumberException(*String* *message*)

    {

*super*(message);

    }

*//fill the code*

}

---------------------------------------ElectricityBill.txt--------------------------------

0191919191,John,Chennai,650

0191919192,Peter,Mumbai,1100

1919191919,Rose,Mumbai,453

0191919193,Tom,Hyderabad,750

01919191945,Raj,Chennai,120

0191919194,Sam,Chennai,250

0191919195,Anya,Chennai,34

--------------------------------db.properties------------------------------

------------------------db.classname=com.mysql.jdbc.Driver

db.url=jdbc:mysql://localhost:3306/ElectricityBill

db.username=

db.password=