Java Task – 2

1. Use Case: Electricity Bill Generation.

Write a code to take the service Number, previous reading, and current reading as input and then find the number of units consumed, then calculate the cost of electricity as follows.

If units consumed is <100 then cost @Rs.1/per unit.

Else if units consumed <150 and >=100 then cost @Rs.1.5 per unit.

Else if unit consumed <200 and >=150 then cost @Rs.2 per unit.

Else if >=200 units then cost @Rs.2.5 per unit.

Now calculate 5% CGST on the price and 15% SGST then find the total amount to pay.

Display all the details on the output screen.

Program 1:

```
import java.util.Scanner;
public class ElectricityBillGeneration {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter Service Number: ");
        String serviceNumber = scanner.nextLine();
        System.out.print("Enter Previous Reading: ");
        int previousReading = scanner.nextInt()
        System.out.print("Enter Current Reading: ");
        int currentReading = scanner.nextInt();
        int unitsConsumed = currentReading - previousReading;
        double cost;
        if (unitsConsumed < 100) {
            cost = unitsConsumed * 1.0;
        } else if (unitsConsumed < 150) {
            cost = unitsConsumed * 1.5;
        } else if (unitsConsumed < 200) {
            cost = unitsConsumed * 2.0;
        } else {
```

```
            cost = unitsConsumed * 2.5;
        }
        double cgst = cost * 0.05;
        double sgst = cost * 0.15;
        double totalAmount = cost + cgst + sgst;
        System.out.println("\nElectricity Bill Details:");
        System.out.println("Service Number: " + serviceNumber);
        System.out.println("Previous Reading: " + previousReading);
        System.out.println("Current Reading: " + currentReading);
        System.out.println("Units Consumed: " + unitsConsumed);
        System.out.println("Cost of Electricity: Rs. " + cost);
        System.out.println("CGST (5%): Rs. " + cgst);
        System.out.println("SGST (15%): Rs. " + sgst);
        System.out.println("Total Amount to Pay: Rs. " + totalAmount);
        scanner.close();
    }
}
```

```
Enter Service Number: 123456789
Enter Previous Reading: 300
Enter Current Reading: 500

Electricity Bill Details:
Service Number: 123456789
Previous Reading: 300
Current Reading: 500
Units Consumed: 200
Cost of Electricity: Rs. 500.0
CGST (5%): Rs. 25.0
SGST (15%): Rs. 75.0
Total Amount to Pay: Rs. 600.0

=== Code Execution Successful ===
```

2. Use Case: Bus Ticket Generation

Write a code to take boarding stop and destination stop values as integers, the number of adult

passengers, and a number of child passengers as integers. Then find the number of stops the

passengers are travelling. Calculate Rs.10 per adult passenger per stop. Calculate Rs.5 per child

passenger and then find the cost of the ticket(including adult ticket cost and child ticket cost).

No apply an offer as follows to the ticket.

If No.of adult passengers are >=5 then give a discount of 20% of cost of the ticket.

Else If No.of adult passengers are 4 then give a discount of 15% of cost of the ticket.

Else If No.of adult passengers are 3 then give a discount of 10% of cost of the ticket.

Else If No.of adult passengers are 2 then give a discount of 5% of cost of the ticket.

Else If No.of adult passengers are 1 then give a discount of 0% of cost of the ticket.

Then calculate a 10% tax on the cost of the ticket.

Then find the total price of the ticket including tax.

Then display all the data on the output screen.

Program 2:

```
import java.util.Scanner;
public class BusTicketGeneration {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter Boarding Stop: ");
        int boardingStop = scanner.nextInt();
        System.out.print("Enter Destination Stop: ");
        int destinationStop = scanner.nextInt();
        System.out.print("Enter Number of Adult Passengers: ");
        int adultPassengers = scanner.nextInt();
        System.out.print("Enter Number of Child Passengers: ");
        int childPassengers = scanner.nextInt();
        int stopsTraveled = Math.abs(destinationStop - boardingStop);
        int adultCostPerStop = 10;
        int childCostPerStop = 5;
        double adultTicketCost = adultPassengers * stopsTraveled * adultCostPerStop;
        double childTicketCost = childPassengers * stopsTraveled * childCostPerStop;
        double totalTicketCost = adultTicketCost + childTicketCost;
        double discount = 0.0;
        if (adultPassengers >= 5) {
            discount = 0.20;
        } else if (adultPassengers == 4) {
            discount = 0.15;
        } else if (adultPassengers == 3) {
            discount = 0.10;
        } else if (adultPassengers == 2) {
            discount = 0.05;
        }
```

```
        double discountAmount = totalTicketCost * discount;
        double discountedCost = totalTicketCost - discountAmount;
        double tax = discountedCost * 0.10;
        double totalPrice = discountedCost + tax;
        System.out.println("\nBus Ticket Details:");
        System.out.println("Boarding Stop: " + boardingStop);
        System.out.println("Destination Stop: " + destinationStop);
        System.out.println("Stops Traveled: " + stopsTraveled);
        System.out.println("Number of Adult Passengers: " + adultPassengers);
        System.out.println("Number of Child Passengers: " + childPassengers);
        System.out.println("Adult Ticket Cost: Rs. " + adultTicketCost);
        System.out.println("Child Ticket Cost: Rs. " + childTicketCost);
        System.out.println("Total Ticket Cost before Discount: Rs. " + totalTicketCost);
        System.out.println("Discount: Rs. " + discountAmount);
        System.out.println("Cost after Discount: Rs. " + discountedCost);
        System.out.println("Tax (10%): Rs. " + tax);
        System.out.println("Total Price of the Ticket: Rs. " + totalPrice);

        scanner.close();
    }
}
```

```
Enter Boarding Stop: 10
Enter Destination Stop: 5
Enter Number of Adult Passengers: 30
Enter Number of Child Passengers: 15

Bus Ticket Details:
Boarding Stop: 10
Destination Stop: 5
Stops Traveled: 5
Number of Adult Passengers: 30
Number of Child Passengers: 15
Adult Ticket Cost: Rs. 1500.0
Child Ticket Cost: Rs. 375.0
Total Ticket Cost before Discount: Rs. 1875.0
Discount: Rs. 375.0
Cost after Discount: Rs. 1500.0
Tax (10%): Rs. 150.0
Total Price of the Ticket: Rs. 1650.0

=== Code Execution Successful ===
```

3. Use Case: ATM Withdraw.

You are given an Initial account balance of 10000, an ATM pin as 1234, and Cash available in the

machine as 8000. Then Write a code to perform the withdraw operation.

Read the Pin from the use and verify whether it is correct or not.

If correct, read the amount ‘X’ to withdraw.

Check given X is a valid amount or not (multiple of 100).

If X is a valid amount, Check X<=balance.

If Yes, then check X amount is available in the ATM machine or not.

If cash is available then print the updated balance.

Else print Transaction failed.

Program 3:

```
import java.util.Scanner;
public class ATMWithdraw {
   public static void main(String[] args)
      int accountBalance = 10000;
      int atmPin = 1234;
      int atmCashAvailable = 8000;
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter ATM Pin: ");
      int enteredPin = scanner.nextInt();
      if (enteredPin != atmPin) {
         System.out.println("Incorrect Pin. Transaction failed.");
         return;
      }
      System.out.print("Enter amount to withdraw: ");
      int withdrawAmount = scanner.nextInt();
      if (withdrawAmount % 100 != 0) {
         System.out.println("Invalid amount. Transaction failed.");
         return;
      }
```

```
        if (withdrawAmount > accountBalance) {
            System.out.println("Insufficient balance. Transaction failed.");
            return;
        }
        if (withdrawAmount > atmCashAvailable) {
            System.out.println("ATM has insufficient cash. Transaction failed.");
            return;
        }
        accountBalance -= withdrawAmount;
        atmCashAvailable -= withdrawAmount;
        System.out.println("Transaction successful.");
        System.out.println("Updated Account Balance: " + accountBalance);
        System.out.println("Cash remaining in ATM: " + atmCashAvailable);

        scanner.close();
    }
}
```

```
Enter ATM Pin: 1234
Enter amount to withdraw: 2000
Transaction successful.
Updated Account Balance: 8000
Cash remaining in ATM: 6000

=== Code Execution Successful ===
```