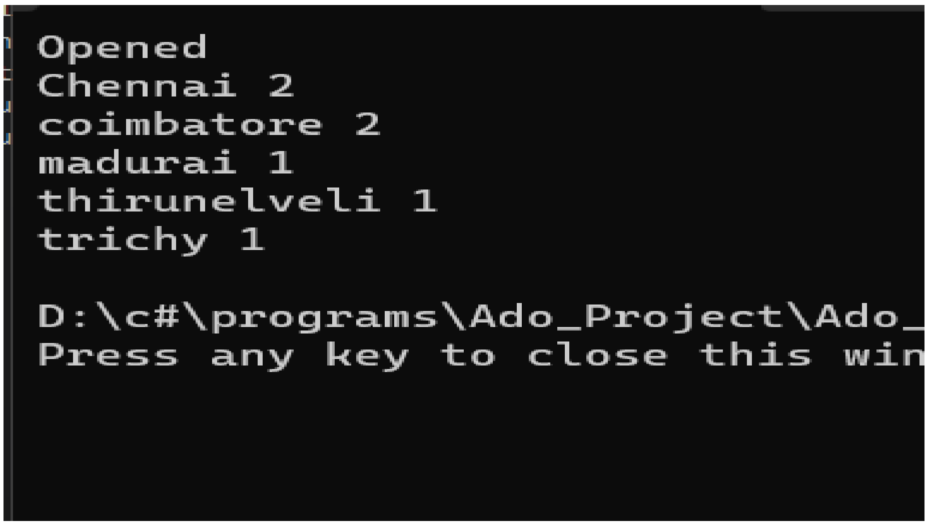
ADO.Net Assessment

1. Display the number of nodes per region

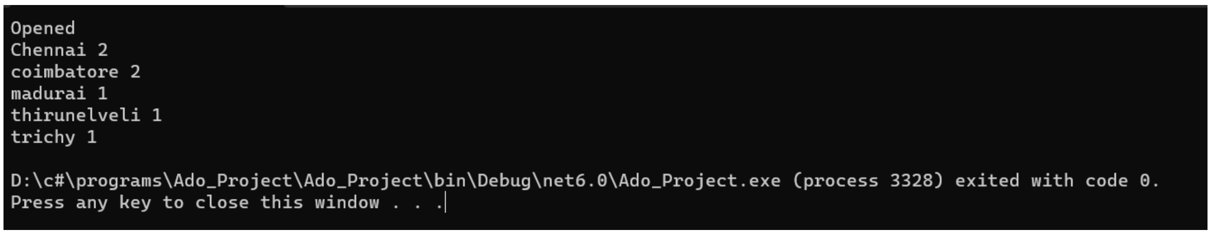
```
public void OpenConnection()
      {
          conn = new SqlConnection("data source =
SPARK\\SQLEXPRESS; " + "database = CustBank; " + "integrated
security = SSPI");
          try
          {
              conn.Open();
              Console.WriteLine("Opened");
          }
          catch (SqlException e)
          {
              Console.WriteLine(e.Message);
              Console.WriteLine(e.StackTrace);
          }

      }


      public void Region()
      {

          SqlCommand cmd = new SqlCommand("create table
branch(region_id  int PRIMARY KEY not null, region nvarchar(20))",
conn);
          if (conn != null)
          {
              cmd.ExecuteNonQuery();
              Console.WriteLine("Region Table  Created");
          }

      }
      public void Customer_Nodes()
      {
          SqlCommand cmd = new SqlCommand("create table
customernodes(region_id int FOREIGN KEY references branch,cust_id
int PRIMARY KEY not null, node_name nvarchar(20))", conn);
          if (conn != null)
          {
              cmd.ExecuteNonQuery();
              Console.WriteLine("Customer Nodes Table Created");
          }

      }
      public void Customer_Transaction()
      {

          SqlCommand cmd = new SqlCommand("create table
customer_transaction(cust_id int FOREIGN KEY references
customernodes, balance int, date_of_transactions date, transaction_amt
int, transaction_mode varchar(10))", conn);
          if (conn != null)
          {
              cmd.ExecuteNonQuery();
              Console.WriteLine("Customer Transaction Table Created");
          }

      }

      public void Insert_Region()
      {
          Console.Write("Enter Region ID : ");
          int r_id = int.Parse(Console.ReadLine());
          Console.Write("Enter Branch Name : ");
          string region = Console.ReadLine();
          SqlCommand cmd = new SqlCommand();
          cmd.CommandType = System.Data.CommandType.Text;
          cmd.Connection = conn;
          cmd.CommandText = "insert into
branch(region_id,region)values(@regionid, @region)";
          cmd.Parameters.AddWithValue("@regionid", r_id);
          cmd.Parameters.AddWithValue("@region", region);

          int _rows = cmd.ExecuteNonQuery();
          if (_rows > 0)
          {
              Console.WriteLine("Values Inserted");
          }
          else
          {
              Console.WriteLine("Failed to Insert");
          }
      }

      public void insert_Customer_Nodes()
      {
          Console.Write("Enter Region ID : ");
          int r_id = int.Parse(Console.ReadLine());
          Console.Write("Enter Customer ID : ");
          int cust_id = int.Parse(Console.ReadLine());
          Console.Write("Enter Region Name : ");
          string node = Console.ReadLine();
          SqlCommand cmd = new SqlCommand();
          cmd.CommandType = System.Data.CommandType.Text;
          cmd.Connection = conn;
          cmd.CommandText = "insert into
customernodes(region_id,cust_id,node_name)values(@regionid,
@custid ,@node)";
          cmd.Parameters.AddWithValue("@regionid", r_id);
          cmd.Parameters.AddWithValue("@custid", cust_id);
          cmd.Parameters.AddWithValue("@node", node);

          int _rows = cmd.ExecuteNonQuery();
          if (_rows > 0)
          {
              Console.WriteLine("Values Inserted");
          }
          else
          {
              Console.WriteLine("Failed to Insert");
          }
      }

public void insert_Customer_transaction()
      {

          Console.Write("Enter Customer ID : ");
          int cust_id = int.Parse(Console.ReadLine());
          Console.Write("Enter Balance : ");
          int bal = int.Parse(Console.ReadLine());
          Console.Write("Enter Date of Transaction: ");
          int date = int.Parse(Console.ReadLine());
          Console.Write("Enter Transaction Amount: ");
          int trans_amt = int.Parse(Console.ReadLine());
          Console.Write("Enter Transaction Mode :");
          string trans_mode = Console.ReadLine();

          SqlCommand cmd = new SqlCommand();
          cmd.CommandType = System.Data.CommandType.Text;
          cmd.Connection = conn;
          cmd.CommandText = "insert into
customer_transaction(cust_id,balance,node_name)values(@custid,
@bal ,@date, @transdate,@transamt,@transmode)";
          cmd.Parameters.AddWithValue("@balance", bal);
          cmd.Parameters.AddWithValue("@custid", cust_id);
          cmd.Parameters.AddWithValue("@date", date);
          cmd.Parameters.AddWithValue("@transamt", trans_amt);
          cmd.Parameters.AddWithValue("@transmode", trans_mode);

          int _rows = cmd.ExecuteNonQuery();
          if (_rows > 0)
          {
              Console.WriteLine("Values Inserted");
          }
          else
          {
              Console.WriteLine("Failed to Insert");
          }

           Connect connect = new Connect();
           connect.OpenConnection();
           connect.CreateTable();
           connect.InsertValues();
           connect.qtn3();

public void qtn1()
      {
          SqlCommand cmd = new SqlCommand();
          cmd = new SqlCommand("select r.region, count( distinct branch)
node_counts from Customer_nodes c " +
            "inner join region r on c.region=r.region group by
r.region",conn);

         SqlDataReader r=cmd.ExecuteReader();
         while (r.Read())
         {
            Console.WriteLine(r[0] + " " + r[1]);
         }
                }
```



```
Opened
Chennai 2
coimbatore 2
madurai 1
thirunelveli 1
trichy 1

D:\c#\programs\Ado_Project\Ado_
Press any key to close this win
```

2. Display the number of customers allocated to each region

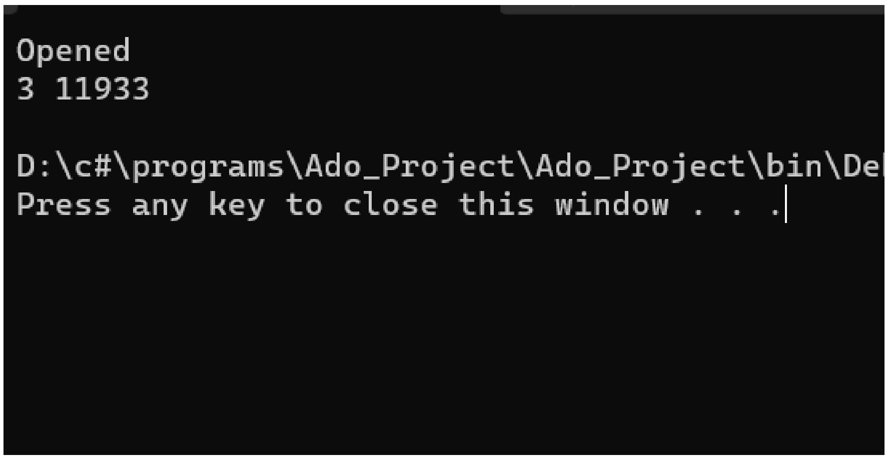
```
public void qtn2()

public void qtn2()
      {
         SqlCommand cmd = new SqlCommand();
         cmd = new SqlCommand("select r.region, count(distinct
c.Customer_id) customer_counts from Customer_nodes c " +
            "inner join region r on c.region=r.region group by
r.region",conn);
         SqlDataReader r = cmd.ExecuteReader();
         while (r.Read())
         {
            Console.WriteLine(r[0] + " " + r[1]);
         }
      }
```



```
Opened
Chennai 2
coimbatore 2
madurai 1
thirunelveli 1
trichy 1

D:\c#\programs\Ado_Project\Ado_Project\bin\Debug\net6.0\Ado_Project.exe (process 3328) exited with code 0.
Press any key to close this window . . .
```

3. Display the total count and average amount of deposits for all the customers

```
Connect.qtn3()

public void qtn3()
        {
           SqlCommand cmd = new SqlCommand();
           cmd = new SqlCommand("select count(*) Total_count , AVG(balance) Average_amount from " +
              "Customer_transaction where Type_of_Transaction='debit'", conn);

           SqlDataReader r = cmd.ExecuteReader();

           while (r.Read())
           {
              Console.WriteLine(r[0] + " " + r[1]);
           }
           r.Close();
           cmd = new SqlCommand("UPDATE Customer_Transaction  SET transaction_amount =
transaction_amount - balance  WHERE Type_of_Transaction='debit'", conn);
           cmd = new SqlCommand("UPDATE Customer_Transaction  SET transaction_amount =
transaction_amount + balance WHERE Type_of_Transaction='credit'", conn);

        }
```

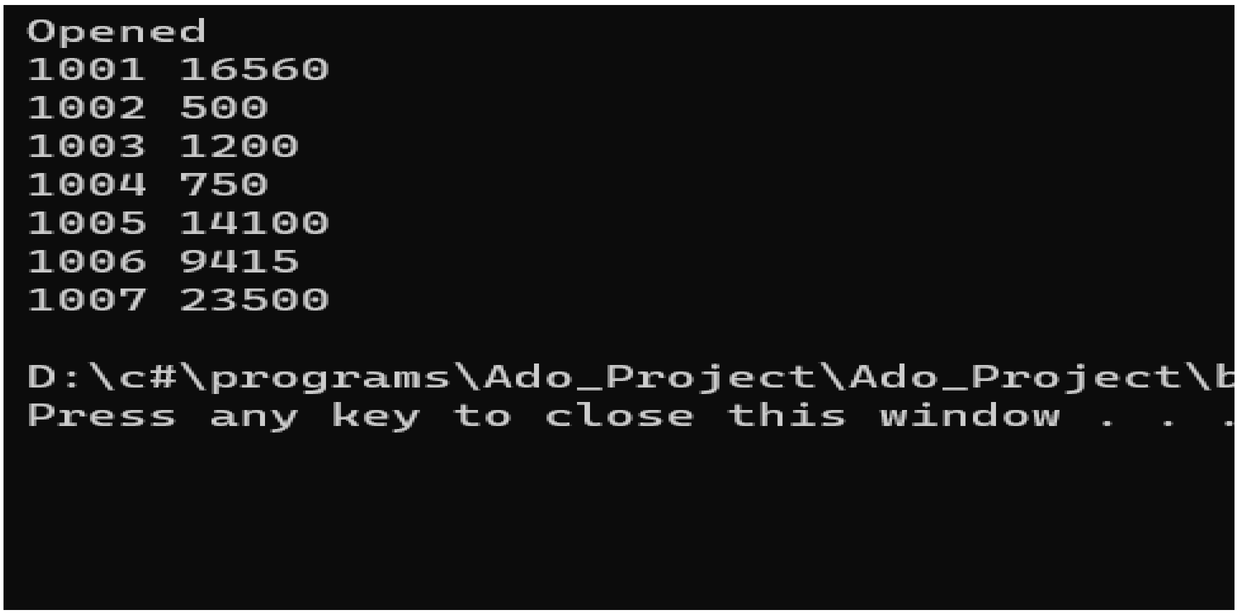


```
Opened
3 11933

D:\c#\programs\Ado_Project\Ado_Project\bin\De
Press any key to close this window . . .
```

4. Display the closing balance for each customer at the end of the month

```
public void qtn4()
        {
           SqlCommand cmd = new SqlCommand();
```

```
            cmd = new SqlCommand("select customer_id,transaction_amount from Customer_Transaction
where Month(Date_of_Transaction)=01", conn);
            SqlDataReader r = cmd.ExecuteReader();

            while (r.Read())
            {
                Console.WriteLine(r[0] + " " + r[1]);
            }
            r.Close();
        }
```



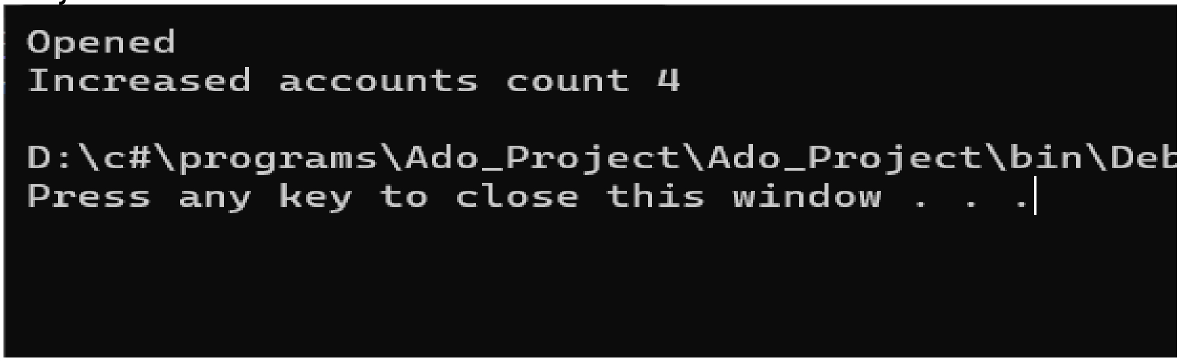
```
Opened
1001 16560
1002 500
1003 1200
1004 750
1005 14100
1006 9415
1007 23500

D:\c#\programs\Ado_Project\Ado_Project\b
Press any key to close this window . . .
```

5. Display the number of customers who have increased their closing balance compared to the

previous month.

Connect.qtn5;

```
public void qtn5()
        {
            SqlCommand cmd = new SqlCommand();
            cmd = new SqlCommand(" select   n.Customer_id
,count(n.Customer_id) count_of_Increased_acc from
Customer_Transaction t  inner join Customer_nodes n on
t.Customer_id=n.Customer_id  where t.balance>
n.Cust_balance”,conn);
            SqlDataReader r = cmd.ExecuteReader();
            while (r.Read())
            {
                Console.WriteLine("Increased accounts count "+r[0] );
            }
            r.Close();
        }
```



```
Opened
Increased accounts count 4

D:\c#\programs\Ado_Project\Ado_Project\bin\Deb
Press any key to close this window . . .
```