Smt. Chandaben Mohanbhai Patel Institute of Computer Applications

BSc.IT SEMESTER 5

CA602 APPLICATION DEVELOPMENT USING .NET FRAMEWORK

CASE STUDY TOPIC – INVENTORY MANAGEMENT SYSTEM

Submitted To - DHATRI RAVAL

Submitted By – 20BSIT056 ANJALI BIJU

20BSIT071 MIT VYAS

20BSIT094 NIDHI KHATRI

Inventory Management

Inventory management refers to the process of ordering, storing, using, and selling a company's inventory. This includes the management of raw materials, components, and finished products, as well as warehousing and processing of such items.

Database: Inventory_database.mdf

Table Name: User_detail

Column Name	Data type	Purpose
Id	int	Id of the row (Primary key) auto incremented
Username	varchar(50)	It stores the username which the admin is required to input
Password	varchar(50)	It stores the username which the admin is required to input
Firstname	varchar(50)	First name of the admin
Lastname	varchar(50)	Last name of the admin
Email	varchar(50)	Email of the admin

Table Name: Units

Column Name	Data type	Purpose
Id	int	Id of the row (Primary key) auto incremented
unit	varchar(50)	Stores unit like Kg, grams, etc.

Table Name: Dealer_info

Column Name	Data type	Purpose
Id	int	Id of the row (Primary key) auto incremented
Dealer_name	varchar(50)	Stores the dealers' names
Dealer_company_name	varchar(50)	Stores dealers' company's names
Contact	varchar(50)	Stores contact number of dealer
Address	varchar(MAX)	Stores address of dealer
City	varchar(50)	Stores dealers' city

Table Name: Product_name

Column Name	Data type	Purpose
Id	int	Id of the row (Primary key) auto incremented
Product_name	varchar(50)	Stores the names of product that the owner sells

Unit	varchar(50)	Stores the unit (from Units table) of the product

Table Name: Purchase_master

Column Name	Data type	Purpose
Id	int	Id of the row (Primary key) auto incremented
Product_name	varchar(50)	Stores the product name(from Product_name table) bought from the dealer
Product_qty	varchar(50)	Stores the quantity of product bought
Product_unit	varchar(50)	Stores the unit (from Product_name table)
Product_price	varchar(50)	Stores the price of product per unit
Product_total	varchar(50)	Stores the total price base on the qty and price per unit
Purchase_date	varchar(50)	Stores the date on which it was purchased from the dealer
Purchace_partyname	varchar(50)	Stores the name of the dealer(from Dealer_info table) from whom product was bought
Purchase_type	varchar(50)	Stores the payment method(Cash, UPI, debit card, etc.)
Expiry_date	varchar(50)	Stores the expiry date of the product purchased
Profit	varchar(50)	Stores the profit gained from the purchase

Table Name: Stock

Column Name	Data type	Purpose
Id	int	Id of the row (Primary key) auto incremented
Product_name	varchar(50)	Stores the name of product (from Product_name table) in stock
Product_qty	int	Stores the quantity of product that is available in stock (updates every time when the products are purchased from dealer or sold to consumer)

Product_unit	varchar(50)	Stores the unit (from Product_name table) of product

Table Name: Order_user

Column Name	Data type	Purpose
Id	int	Id of the row (Primary key) auto incremented
First_name	varchar(50)	Stores the first name of the consumer
Last_name	varchar(50)	Stores the last name of the consumer
Bill_type	varchar(50)	Stores the payment method used by the consumer
Purchase_date	varchar(50)	Stores the date on which the consumer bought the product(s)

Table Name: Order_Item

Column Name	Data type	Purpose
Id	int	Id of the row (Primary key) auto incremented
Order_id	varchar(50)	Stores the Id from Order_user table
Product	varchar(50)	Stores product name bought by consumer
Price	varchar(50)	Stores the price (from Purchase_master table) of the product per unit
Quantity	varchar(50)	Stores the quantity (from Stock table) of product bought by consumer
Total	varchar(50)	Stores the total price to paid for that product bought

Code:

Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace ADNF_casestudy
{
   static class Program
   {
      /// <summary>
      /// The main entry point for the application.
      /// </summary>
      [STAThread]
      static void Main()
      {
         Application.EnableVisualStyles();
         Application.SetCompatibleTextRenderingDefault(false);
         Application.Run(new Login_Form());

      }
   }
```

```
}
```

Login_form.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace ADNF_casestudy
{
   public partial class Login_Form : Form
   {
      SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Welcome\Docume
nts\Inventory_Database.mdf;Integrated Security=True;Connect Timeout=30");
      public Login_Form()
      {

         InitializeComponent();
```

```
    details_fetch();
    textBox1.Text = null;
    textBox2.Text = null;
}

void details_fetch()
{
    String q = "select Username,Password from User_detail";
    SqlCommand cmd = new SqlCommand(q, con);
    con.Open();
    SqlDataReader sdr = cmd.ExecuteReader();
    while (sdr.Read())
    {
        name = sdr[0].ToString();
        pw = sdr[1].ToString();
    }
    con.Close();
}
String name,pw;
private void button1_Click(object sender, EventArgs e)
{

    if(textBox1.Text!="" && textBox2.Text!="")
    {
```

```
    if (textBox1.Text != name && textBox2.Text != pw)
    {
        MessageBox.Show("Username and Password Invalid");
        textBox1.Clear();
        textBox2.Clear();
    }
    else if (textBox1.Text!=name)
    {
        MessageBox.Show("Username Invalid");
        textBox1.Clear();
    }
    else if(textBox2.Text != pw)
    {
        MessageBox.Show("Password Invalid");
        textBox2.Clear();
    }
    else
    {
        this.Hide();
        Home h = new Home();
        h.Show();
    }
}
else
```

```
        {
            MessageBox.Show("Enter credentials");
        }
    }
  }
}
```

Home.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace ADNF_casestudy
{
  public partial class Home : Form
  {
    private int childFormNumber = 0;
    public Home()
```

```
        {
            InitializeComponent();
        }

        private void ShowNewForm(object sender, EventArgs e)
        {
            Form childForm = new Form();
            childForm.MdiParent = this;
            childForm.Text = "Window " + childFormNumber++;
            childForm.Show();
        }

        private void OpenFile(object sender, EventArgs e)
        {
            OpenFileDialog openFileDialog = new OpenFileDialog();
            openFileDialog.InitialDirectory =
Environment.GetFolderPath(Environment.SpecialFolder.Personal);
            openFileDialog.Filter = "Text Files (*.txt)|*.txt|All Files (*.*)|*.*";
            if (openFileDialog.ShowDialog(this) == DialogResult.OK)
            {
                string FileName = openFileDialog.FileName;
            }
        }

        private void SaveAsToolStripMenuItem_Click(object sender, EventArgs e)
```

```
        {
            SaveFileDialog saveFileDialog = new SaveFileDialog();
            saveFileDialog.InitialDirectory =
Environment.GetFolderPath(Environment.SpecialFolder.Personal);
            saveFileDialog.Filter = "Text Files (*.txt)|*.txt|All Files (*.*)|*.*";
            if (saveFileDialog.ShowDialog(this) == DialogResult.OK)
            {
                string FileName = saveFileDialog.FileName;
            }
        }

        private void ExitToolsStripMenuItem_Click(object sender, EventArgs e)
        {
            this.Close();
        }

        private void CutToolStripMenuItem_Click(object sender, EventArgs e)
        {
        }

        private void CopyToolStripMenuItem_Click(object sender, EventArgs e)
        {
        }

        private void PasteToolStripMenuItem_Click(object sender, EventArgs e)
```

```
        {
        }

        private void ToolBarToolStripMenuItem_Click(object sender, EventArgs e)
        {
            //toolStrip.Visible = toolBarToolStripMenuItem.Checked;
        }

        private void StatusBarToolStripMenuItem_Click(object sender, EventArgs e)
        {
            //statusStrip.Visible = statusBarToolStripMenuItem.Checked;
        }

        private void CascadeToolStripMenuItem_Click(object sender, EventArgs e)
        {
            LayoutMdi(MdiLayout.Cascade);
        }

        private void TileVerticalToolStripMenuItem_Click(object sender, EventArgs e)
        {
            LayoutMdi(MdiLayout.TileVertical);
        }

        private void TileHorizontalToolStripMenuItem_Click(object sender, EventArgs
e)
```

```
        {
            LayoutMdi(MdiLayout.TileHorizontal);
        }

        private void ArrangeIconsToolStripMenuItem_Click(object sender, EventArgs
e)
        {
            LayoutMdi(MdiLayout.ArrangeIcons);
        }

        private void CloseAllToolStripMenuItem_Click(object sender, EventArgs e)
        {
            foreach (Form childForm in MdiChildren)
            {
                childForm.Close();
            }
        }

        private void addNewUnitsToolStripMenuItem_Click(object sender, EventArgs
e)
        {
            Unit un = new Unit();
            un.Show();
        }
```

```
    private void aboutUserToolStripMenuItem_Click(object sender, EventArgs e)
    {
        User_info us = new User_info();
        us.Show();
    }

    private void addProductsToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        Modify_Products ap = new Modify_Products();
        ap.Show();
    }

    private void modifyDealersToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        Dealer_info di = new Dealer_info();
        di.Show();
    }

    private void purchaseProductsToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        Purchase_product pur = new Purchase_product();
        pur.Show();
```

```
    }

    private void salesProductsToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        Sales sale = new Sales();
        sale.Show();
    }

    private void purchaseReportToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        Purchase_report pr = new Purchase_report();
        pr.Show();
    }

    private void stocksReportToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        Stock_report sr = new Stock_report();
        sr.Show();
    }

    private void orderDetailReportToolStripMenuItem_Click(object sender,
EventArgs e)
    {
```

```
            Order_detail_report odr = new Order_detail_report();
            odr.Show();
        }
    }
}
```

User_info.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace ADNF_casestudy
{
    public partial class User_info : Form
    {
        public User_info()
        {
```

```
        InitializeComponent();
    }
    SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Welcome\Docume
nts\Inventory_Database.mdf;Integrated Security=True;Connect Timeout=30");

    private void User_info_Load(object sender, EventArgs e)
    {
        String q1 = "select Firstname,Lastname,Email from User_Detail";

        SqlCommand cmd1 = new SqlCommand(q1, con);

        con.Open();
        SqlDataReader sdr1 = cmd1.ExecuteReader();

        while (sdr1.Read())
        {
            label4.Text = sdr1[0].ToString();
            label5.Text = sdr1[1].ToString();
            label6.Text = sdr1[2].ToString();
        }

        con.Close();
    }
}
```

```
}
```

Modify_Products.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace ADNF_casestudy
{
   public partial class Modify_Products : Form
   {
      public Modify_Products()
      {
         InitializeComponent();
         fill_cbox();
         display();
```

```
    }
    SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Welcome\Docume
nts\Inventory_Database.mdf;Integrated Security=True;Connect Timeout=30");
    public static bool Validations(String str)
    {
      //Only alphabets are allowed here
      bool output = true;
      String regex1 = "[^a-zA-Z ]+";

      Regex rgex1 = new Regex(regex1);

      if (str == null)
      {
        output = false;
        return output;
      }
      MatchCollection matchedAuthors1 = rgex1.Matches(str);

      if (matchedAuthors1.Count != 0)
      {
        output = false;
        return output;
      }
      else
```

```
    {
        return output;
    }
}
void fill_cbox()
{
    String q = "select unit from Units";
    SqlCommand cmd = new SqlCommand(q, con);
    con.Open();
    SqlDataReader sdr = cmd.ExecuteReader();
    while (sdr.Read())
    {
        comboBox1.Items.Add(sdr[0].ToString());
        comboBox2.Items.Add(sdr[0].ToString());

    }
    con.Close();
}

public void display()
{
    String q = "select * from Product_name";
    SqlDataAdapter sda = new SqlDataAdapter(q, con);
    DataSet ds = new DataSet();
```

```
    sda.Fill(ds);
    dataGridView1.DataSource = ds.Tables[0];
}



private void insert_btn_Click(object sender, EventArgs e)
{
    if(textBox1.Text != "" && comboBox1.Text != "")
    {
        if (Validations(textBox1.Text))
        {
            String q = "insert into Product_name values(@product,@unit)";
            SqlCommand cmd = new SqlCommand(q, con);
            cmd.Parameters.AddWithValue("@product", textBox1.Text);
            cmd.Parameters.AddWithValue("@unit", comboBox1.SelectedItem);
            con.Open();
            cmd.ExecuteNonQuery();
            con.Close();
            MessageBox.Show("Data inserted");
            textBox1.Clear();
            comboBox1.ResetText();
            display();
        }
        else
```

```
            {
                MessageBox.Show("Enter Product name properly");
                textBox1.Clear();
            }
        }
        else
        {
            MessageBox.Show("Enter all product details");
        }



    }
    int i;
    private void dataGridView1_CellClick(object sender,
DataGridViewCellEventArgs e)
    {
        i = Convert.ToInt32(dataGridView1.SelectedCells[0].Value.ToString());



        String q1 = "select * from Product_name where id = '"+i+"'";
        SqlDataAdapter sda1 = new SqlDataAdapter(q1, con);
        DataTable dt1 = new DataTable();
        sda1 .Fill(dt1);
        foreach (DataRow dr1 in dt1.Rows)
        {
```

```
            textBox2.Text = dr1["Product_name"].ToString();
            comboBox2.SelectedItem = dr1["Unit"].ToString();
        }
    }

    private void update_btn_Click(object sender, EventArgs e)
    {
        if(textBox2.Text != "" && comboBox2.Text != "")
        {
            if (Validations(textBox2.Text))
            {
                String q = "update Product_name set Product_name = @product,Unit
= @unit where id = '" + i + "'";
                SqlCommand cmd = new SqlCommand(q, con);
                cmd.Parameters.AddWithValue("@product", textBox2.Text);
                cmd.Parameters.AddWithValue("@unit", comboBox2.SelectedItem);
                con.Open();
                cmd.ExecuteNonQuery();
                con.Close();
                MessageBox.Show("Data updated");
                textBox2.Clear();
                comboBox2.ResetText();
                display();
            }
            else
```

```
            {
                MessageBox.Show("Enter Product name properly");
                textBox2.Clear();
            }
        }
            else
            {
                MessageBox.Show("Enter all product details");
            }
}
   }
}
```

Purchase_product.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Threading.Tasks;
```

```
using System.Windows.Forms;


namespace ADNF_casestudy
{
   public partial class Purchase_product : Form
   {
      public Purchase_product()
      {
         InitializeComponent();
         fill_cb1();
      }
      SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Welcome\Docume
nts\Inventory_Database.mdf;Integrated Security=True;Connect Timeout=30");
      public void fill_cb1()
      {
         String q1 = "select Product_name from Product_name";
         String q2 = "select Dealer_name from Dealer_info";
         SqlCommand cmd1 = new SqlCommand(q1, con);
         SqlCommand cmd2 = new SqlCommand(q2, con);
         con.Open();
         SqlDataReader sdr1 = cmd1.ExecuteReader();


         while (sdr1.Read())
         {
```

```
            comboBox1.Items.Add(sdr1[0].ToString());
        }

        con.Close();

        con.Open();
        SqlDataReader sdr2 = cmd2.ExecuteReader();

        while (sdr2.Read())
        {
            comboBox2.Items.Add(sdr2[0].ToString());
        }

        con.Close();
    }
    private void pur_item_btn_Click(object sender, EventArgs e)
    {
        if(textBox1.Text!="" && textBox2.Text != "" && textBox3.Text != "" &&
textBox4.Text != "" && comboBox1.Text != "" && comboBox2.Text != "" &&
comboBox3.Text != "")
        {
            if(Validations2(textBox1.Text) && Validations2(textBox2.Text) &&
Validations2(textBox3.Text) && Validations2(textBox4.Text))
            {
                int i = 0;
```

```
            decimal qty = Convert.ToDecimal(textBox1.Text);
            String q = "select Product_qty from stock where Product_name=@pn";
            SqlCommand cmd = new SqlCommand(q, con);
            cmd.Parameters.AddWithValue("@pn", comboBox1.SelectedItem);
            con.Open();
            SqlDataReader sdr = cmd.ExecuteReader();
            while (sdr.Read())
            {
               i++;
               qty = Convert.ToDecimal((sdr[0]));
            }
            con.Close();

            if (i == 0)
            {
               String q1 = "insert into Purchase_master
values(@pro_n,@pro_q,@pro_u,@pro_p,@pro_t,@pur_d,@pur_p,@pur_t,@exp
_d,@profit)";
               SqlCommand cmd1 = new SqlCommand(q1, con);
               cmd1.Parameters.AddWithValue("@pro_n",
comboBox1.SelectedItem);
               cmd1.Parameters.AddWithValue("@pro_q", textBox1.Text);
               cmd1.Parameters.AddWithValue("@pro_u", label10.Text);
               cmd1.Parameters.AddWithValue("@pro_p", textBox2.Text);
               cmd1.Parameters.AddWithValue("@pro_t", textBox3.Text);
```

```
            cmd1.Parameters.AddWithValue("@pur_d",
dateTimePicker1.Value.ToString("dd-MM-yyyy"));
            cmd1.Parameters.AddWithValue("@pur_p",
comboBox2.SelectedItem);
            cmd1.Parameters.AddWithValue("@pur_t",
comboBox3.SelectedItem);
            cmd1.Parameters.AddWithValue("@exp_d",
dateTimePicker2.Value.ToString("dd-MM-yyyy"));
            cmd1.Parameters.AddWithValue("@profit", textBox4.Text);


            con.Open();
            cmd1.ExecuteNonQuery();
            con.Close();


            String q2 = "insert into Stock values(@pro_n,@pro_q,@pro_u)";
            SqlCommand cmd2 = new SqlCommand(q2, con);
            cmd2.Parameters.AddWithValue("@pro_n",
comboBox1.SelectedItem);
            cmd2.Parameters.AddWithValue("@pro_q", textBox1.Text);
            cmd2.Parameters.AddWithValue("@pro_u", label10.Text);


            con.Open();
            cmd2.ExecuteNonQuery();
            con.Close();
            MessageBox.Show("Data inserted");
        }
```

```
            else
            {
                String q3 = "insert into Purchase_master
values(@pro_n,@pro_q,@pro_u,@pro_p,@pro_t,@pur_d,@pur_p,@pur_t,@exp
_d,@profit)";
                SqlCommand cmd3 = new SqlCommand(q3, con);
                cmd3.Parameters.AddWithValue("@pro_n",
comboBox1.SelectedItem);
                cmd3.Parameters.AddWithValue("@pro_q", textBox1.Text);
                cmd3.Parameters.AddWithValue("@pro_u", label10.Text);
                cmd3.Parameters.AddWithValue("@pro_p", textBox2.Text);
                cmd3.Parameters.AddWithValue("@pro_t", textBox3.Text);
                cmd3.Parameters.AddWithValue("@pur_d",
dateTimePicker1.Value.ToString("dd-MM-yyyy"));
                cmd3.Parameters.AddWithValue("@pur_p",
comboBox2.SelectedItem);
                cmd3.Parameters.AddWithValue("@pur_t",
comboBox3.SelectedItem);
                cmd3.Parameters.AddWithValue("@exp_d",
dateTimePicker2.Value.ToString("dd-MM-yyyy"));
                cmd3.Parameters.AddWithValue("@profit", textBox4.Text);

                con.Open();
                cmd3.ExecuteNonQuery();
                con.Close();
```

```
                String q4 = "update Stock set Product_qty=" + (qty +
Convert.ToDecimal(textBox1.Text)) + " where Product_name=@pro_n";
                SqlCommand cmd4 = new SqlCommand(q4, con);
                cmd4.Parameters.AddWithValue("@pro_n",
comboBox1.SelectedItem);
                cmd4.Parameters.AddWithValue("@pro_q", textBox1.Text);


                con.Open();
                cmd4.ExecuteNonQuery();
                con.Close();
                MessageBox.Show("Data inserted");
                textBox1.Clear();
                textBox2.Clear();
                textBox3.Clear();
                textBox4.Clear();
                comboBox1.ResetText();
                comboBox2.ResetText();
                comboBox3.ResetText();
                dateTimePicker1.ResetText();
                dateTimePicker2.ResetText();
            }
        }
        else
        {
            MessageBox.Show("Enter proper data ");
```

```
        }
    }
    else
    {
        MessageBox.Show("Enter all data ");
    }

}

public static bool Validations2(String str)
{
    //Only numbers are allowed here
    bool output = true;

    String regex2 = "[^0-9.]+";

    Regex rgex2 = new Regex(regex2);
    if (str == null)
    {
        output = false;
        return output;
    }

    MatchCollection matchedAuthors2 = rgex2.Matches(str);
```

```
        if (matchedAuthors2.Count != 0)
        {
            output = false;
            return output;
        }
        else
        {
            return output;
        }
    }

    private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
    {
        String q = "select Unit from Product_name where
Product_name=@p_name";
        SqlCommand cmd = new SqlCommand(q, con);
        cmd.Parameters.AddWithValue("@p_name", comboBox1.SelectedItem);
        con.Open();
        SqlDataReader sdr = cmd.ExecuteReader();
        while (sdr.Read())
        {
            label10.Text = (sdr[0]).ToString();
        }
        con.Close();
```

```
    }

    private void textBox2_Leave(object sender, EventArgs e)
    {
      if (Validations2(textBox1.Text) && Validations2(textBox2.Text))
      {

        textBox3.Text = (Convert.ToDecimal(textBox1.Text) *
Convert.ToDecimal(textBox2.Text)).ToString();

      }
      else
      {
        MessageBox.Show("Enter Product details properly");
        textBox1.Clear();
        textBox2.Clear();
      }
    }

}
}
```

Sales.cs

```
using System;
using System.Collections.Generic;
```

```
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace ADNF_casestudy
{
   public partial class Sales : Form
   {
      public Sales()
      {
         InitializeComponent();
      }
      SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Welcome\Docume
nts\Inventory_Database.mdf;Integrated Security=True;Connect Timeout=30");
      DataTable dt = new DataTable();
      decimal tot = 0;
      private void Sales_Load(object sender, EventArgs e)
      {
```

```
            dt.Clear();
            dt.Columns.Add("Product");
            dt.Columns.Add("Price");
            dt.Columns.Add("Quantity");
            dt.Columns.Add("Total");
        }

        private void textBox4_KeyUp(object sender, KeyEventArgs e)
        {
            listBox1.Visible = true;

            listBox1.Items.Clear();
            String q = "select Product_name from Stock where Product_name like ('"+
textBox4.Text + "%')";
            SqlCommand cmd = new SqlCommand(q,con);
            con.Open();
            SqlDataReader sdr = cmd.ExecuteReader();
            while (sdr.Read())
            {
                listBox1.Items.Add(sdr[0].ToString());
            }
            con.Close();
        }

        private void textBox4_KeyDown(object sender, KeyEventArgs e)
```

```
{
   if(e.KeyCode==Keys.Down)
   {
      listBox1.Focus();
      listBox1.SelectedIndex = 0;
   }

}

private void listBox1_KeyDown(object sender, KeyEventArgs e)
{
   try
   {
      if(e.KeyCode == Keys.Down)
      {
         this.listBox1.SelectedItem = this.listBox1.SelectedIndex + 1;
      }
      if(e.KeyCode == Keys.Up)
      {
         this.listBox1.SelectedItem = this.listBox1.SelectedIndex - 1;
      }
      if(e.KeyCode == Keys.Enter)
      {
         textBox4.Text = listBox1.SelectedItem.ToString();
```

```
            listBox1.Visible = false;
            textBox3.Focus();
          }
        }
        catch(Exception ex)
        {

        }
      }

      private void textBox3_Enter(object sender, EventArgs e)
      {
        String q = "select top 1 Product_price from Purchase_master where
Product_name=@pn order by Id desc";
        SqlCommand cmd = new SqlCommand(q, con);
        cmd.Parameters.AddWithValue("@pn", textBox4.Text);
        con.Open();
        SqlDataReader sdr = cmd.ExecuteReader();
        while (sdr.Read())
        {
          textBox3.Text= (sdr[0].ToString());
        }
        con.Close();
      }
```

```
    private void textBox5_Leave(object sender, EventArgs e)
    {
        try
        {
            textBox6.Text = (Convert.ToDecimal(textBox3.Text) *
Convert.ToDecimal(textBox5.Text)).ToString();
        }
        catch(Exception ex)
        {


        }
    }

    public static bool Validations(String str)
    {
        //Only alphabets are allowed here
        bool output = true;
        String regex1 = "[^a-zA-Z ]+";

        Regex rgex1 = new Regex(regex1);

        if (str == null)
        {
            output = false;
            return output;
```

```
    }
    MatchCollection matchedAuthors1 = rgex1.Matches(str);

    if (matchedAuthors1.Count != 0)
    {
        output = false;
        return output;
    }
    else
    {
        return output;
    }
}

public static bool Validations2(String str)
{
    //Only numbers are allowed here
    bool output = true;

    String regex2 = "[^0-9.]+";

    Regex rgex2 = new Regex(regex2);
    if (str == null)
    {
```

```
            output = false;
            return output;
        }

        MatchCollection matchedAuthors2 = rgex2.Matches(str);

        if (matchedAuthors2.Count != 0)
        {
            output = false;
            return output;
        }
        else
        {
            return output;
        }
    }

    private void add_btn_Click(object sender, EventArgs e)
    {
        if(textBox1.Text != "" && textBox2.Text != "" && textBox3.Text != "" &&
textBox4.Text != "" && textBox5.Text != "" && textBox6.Text != "" &&
comboBox1.Text != "" )
        {
            if(Validations(textBox1.Text) && Validations(textBox2.Text) &&
Validations(textBox4.Text) && Validations2(textBox3.Text) &&
Validations2(textBox5.Text) && Validations2(textBox6.Text))
```

```
            {
                decimal stock = 0;
                String q = "select Product_qty from Stock where Product_name =
@pn";
                SqlCommand cmd = new SqlCommand(q, con);
                cmd.Parameters.AddWithValue("@pn", textBox4.Text);
                con.Open();
                SqlDataReader sdr = cmd.ExecuteReader();
                while (sdr.Read())
                {
                    stock = Convert.ToDecimal(sdr[0].ToString());
                }
                con.Close();

                if (Convert.ToDecimal(textBox5.Text) > stock)
                {
                    MessageBox.Show("This much stock is not available");
                }
                else
                {
                    DataRow dr = dt.NewRow();
                    dr["Product"] = textBox4.Text;
                    dr["Price"] = textBox3.Text;
                    dr["Quantity"] = textBox5.Text;
                    dr["Total"] = textBox6.Text;
```

```
            dt.Rows.Add(dr);
            dataGridView1.DataSource = dt;

            tot = tot + Convert.ToDecimal(dr["Total"].ToString());
            label10.Text = tot.ToString();

        }
        textBox3.Clear();
        textBox4.Clear();
        textBox5.Clear();
        textBox6.Clear();
      }
      else
      {
        MessageBox.Show("Enter proper data");
      }
    }
    else
    {
      MessageBox.Show("Enter all the data");
    }

}
```

```
    private void del_btn_Click(object sender, EventArgs e)
    {
       try
       {
          tot = 0;

dt.Rows.RemoveAt(Convert.ToInt32(dataGridView1.CurrentCell.RowIndex.ToStrin
g()));
          foreach(DataRow dr1 in dt.Rows)
          {
             tot = tot + Convert.ToDecimal(dr1["Total"].ToString());
             label10.Text = tot.ToString();
          }
       }
       catch(Exception ex)
       {


       }
    }
    private void sav_print_btn_Click(object sender, EventArgs e)
    {
       if (textBox1.Text != "" && textBox2.Text != "" && comboBox1.Text != "")
       {
          string orderid = "";
          String q = "insert into Order_user values(@fn,@ln,@bt,@pd)";
```

```
            SqlCommand cmd = new SqlCommand(q, con);
            cmd.Parameters.AddWithValue("@fn", textBox1.Text);
            cmd.Parameters.AddWithValue("@ln", textBox2.Text);
            cmd.Parameters.AddWithValue("@bt", comboBox1.SelectedItem);
            cmd.Parameters.AddWithValue("@pd",
dateTimePicker1.Value.ToString("dd/MM/yyyy"));

            con.Open();
            cmd.ExecuteNonQuery();
            con.Close();


            String q1 = "select top 1 * from Order_user order by Id desc";
            SqlCommand cmd1 = new SqlCommand(q1, con);
            //cmd1.Parameters.AddWithValue("@pn", textBox4.Text);
            con.Open();
            SqlDataReader sdr = cmd1.ExecuteReader();
            while (sdr.Read())
            {
               orderid = sdr[0].ToString();
            }
            con.Close();

            foreach (DataRow dr in dt.Rows)
            {
```

```
            decimal qty = 0;
            string pname = "";
            String q2 = "insert into Order_item
values(@oid,@prod,@price,@qty,@tot)";
            SqlCommand cmd2 = new SqlCommand(q2, con);
            cmd2.Parameters.AddWithValue("@oid", orderid.ToString());
            cmd2.Parameters.AddWithValue("@prod", dr["Product"].ToString());
            cmd2.Parameters.AddWithValue("@price", dr["Price"].ToString());
            cmd2.Parameters.AddWithValue("@qty", dr["Quantity"].ToString());
            cmd2.Parameters.AddWithValue("@tot", dr["Total"].ToString());

            con.Open();
            cmd2.ExecuteNonQuery();
            con.Close();

            qty = Convert.ToDecimal(dr["Quantity"]);
            pname = dr["Product"].ToString();

            String q3 = "update Stock set Product_qty=Product_qty-" + qty + "
where Product_name = '" + pname.ToString() + "'";
            SqlCommand cmd3 = new SqlCommand(q3, con);

            con.Open();
            cmd3.ExecuteNonQuery();
            con.Close();
```

```
            }
            textBox1.Clear();
            textBox2.Clear();
            textBox3.Clear();
            textBox4.Clear();
            textBox5.Clear();
            textBox6.Clear();
            label10.Text = "";

            dt.Clear();
            dataGridView1.DataSource = dt;

            MessageBox.Show("Data inserted");
        }
        else
        {
            MessageBox.Show("Enter all data properly");
        }
    }

  }
}
```

Unit.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace ADNF_casestudy
{
   public partial class Unit : Form
   {
      public Unit()
      {
         InitializeComponent();
         display();
      }
      SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Welcome\Docume
nts\Inventory_Database.mdf;Integrated Security=True;Connect Timeout=30");
```

```
public static bool Validations(String str)
{
   //Only alphabets are allowed here
   bool output = true;
   String regex1 = "[^a-zA-Z ]+";

   Regex rgex1 = new Regex(regex1);
   if (str == null)
   {
      output = false;
      return output;
   }
   MatchCollection matchedAuthors1 = rgex1.Matches(str);

   if (matchedAuthors1.Count != 0)
   {
      output = false;
      return output;
   }
   else
   {
      return output;
   }
}
```

```
private void add_unit_btn_Click(object sender, EventArgs e)
{
   if (textBox1.Text != "")
   {
      if (Validations(textBox1.Text))
      {
         int count = 0;
         String q1 = "select * from Units where unit = '" + textBox1.Text + "'";
         SqlDataAdapter sda1 = new SqlDataAdapter(q1, con);
         DataTable ds1 = new DataTable();
         sda1.Fill(ds1);
         count = Convert.ToInt32(ds1.Rows.Count.ToString());

         if (count == 0)
         {
            String q = "insert into Units values(@unit)";
            SqlCommand cmd = new SqlCommand(q, con);
            cmd.Parameters.AddWithValue("@unit", textBox1.Text);
            con.Open();
            cmd.ExecuteNonQuery();
            con.Close();
            MessageBox.Show("Data inserted");
            display();
```

```
            textBox1.Clear();
        }
        else
        {
            MessageBox.Show("The unit already exists in database");
            textBox1.Clear();
        }
    }
    else
    {
        MessageBox.Show("Enter a proper unit");
        textBox1.Clear();
    }
  }
  else
  {
    MessageBox.Show("Enter unit");
  }

}

public void display()
{
  String q = "select * from Units";
```

```
            SqlDataAdapter sda = new SqlDataAdapter(q, con);
            DataSet ds = new DataSet();
            sda.Fill(ds);
            dataGridView1.DataSource = ds.Tables[0];
        }
        int id;
        private void del_unit_btn_Click(object sender, EventArgs e)
        {
            id = Convert.ToInt32(dataGridView1.SelectedCells[0].Value.ToString());
            String q = "delete from Units where Id = '"+id+"'";
            SqlCommand cmd = new SqlCommand(q, con);
            con.Open();
            cmd.ExecuteNonQuery();
            con.Close();
            MessageBox.Show("Data deleted");
            display();
            textBox1.Clear();
        }

    }
}
```

Dealer_info.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace ADNF_casestudy
{
   public partial class Dealer_info : Form
   {
      public Dealer_info()
      {
         InitializeComponent();
         display();
      }
      SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Welcome\Docume
nts\Inventory_Database.mdf;Integrated Security=True;Connect Timeout=30");
```

```
public static bool Validations(String str)
{
   //Only alphabets are allowed here
   bool output = true;
   String regex1 = "[^a-zA-Z ]+";

   Regex rgex1 = new Regex(regex1);

   if (str == null)
   {
      output = false;
      return output;
   }
   MatchCollection matchedAuthors1 = rgex1.Matches(str);

   if (matchedAuthors1.Count != 0)
   {
      output = false;
      return output;
   }
   else
   {
      return output;
```

```
    }
}

public static bool Validations2(String str)
{
    //Only numbers are allowed here
    bool output = true;

    String regex2 = "[^0-9]+";

    Regex rgex2 = new Regex(regex2);
    if (str == null)
    {
        output = false;
        return output;
    }

    MatchCollection matchedAuthors2 = rgex2.Matches(str);

    if (matchedAuthors2.Count != 0)
    {
        output = false;
        return output;
    }
```

```
        else
        {
            return output;
        }
    }
    private void insert_btn_Click(object sender, EventArgs e)
    {
        if(textBox1.Text!="" && textBox2.Text != "" && textBox3.Text != "" &&
textBox4.Text != "" && textBox5.Text != "")
        {
            if (Validations(textBox1.Text) && Validations2(textBox3.Text) &&
Validations(textBox5.Text))
            {
                String q = "insert into Dealer_info
values(@dn,@dc,@contact,@address,@city)";
                SqlCommand cmd = new SqlCommand(q, con);
                cmd.Parameters.AddWithValue("@dn", textBox1.Text);
                cmd.Parameters.AddWithValue("@dc", textBox2.Text);
                cmd.Parameters.AddWithValue("@contact", textBox3.Text);
                cmd.Parameters.AddWithValue("@address", textBox4.Text);
                cmd.Parameters.AddWithValue("@city", textBox5.Text);
                con.Open();
                cmd.ExecuteNonQuery();
                con.Close();
                MessageBox.Show("Data inserted");
```

```
            display();
            textBox1.Clear();
            textBox2.Clear();
            textBox3.Clear();
            textBox4.Clear();
            textBox5.Clear();
        }
        else
        {
            MessageBox.Show("Enter dealer details properly");
            textBox1.Clear();
            textBox3.Clear();
            textBox5.Clear();
        }
    }
    else
    {
        MessageBox.Show("Enter all the dealer details");
    }


}

public void display()
```

```
{
    String q = "select * from Dealer_info";
    SqlDataAdapter sda = new SqlDataAdapter(q, con);
    DataSet ds = new DataSet();
    sda.Fill(ds);
    dataGridView1.DataSource = ds.Tables[0];
}
int id;
private void delete_btn_Click(object sender, EventArgs e)
{
    id = Convert.ToInt32(dataGridView1.SelectedCells[0].Value.ToString());
    String q = "delete from Dealer_info where Id = " + id + "";
    SqlCommand cmd = new SqlCommand(q, con);
    con.Open();
    cmd.ExecuteNonQuery();
    con.Close();
    display();
}

private void update_sel_btn_Click(object sender, EventArgs e)
{
    id = Convert.ToInt32(dataGridView1.SelectedCells[0].Value.ToString());
    panel2.Visible = true;
```

```
        String q = "select * from Dealer_info where id=" + id + "";
        SqlDataAdapter sda = new SqlDataAdapter(q, con);
        DataTable dt = new DataTable();

        sda.Fill(dt);
        foreach (DataRow dr in dt.Rows)
        {
            textBox6.Text = dr["Dealer_name"].ToString();
            textBox7.Text = dr["Dealer_company_name"].ToString();
            textBox8.Text = dr["Contact"].ToString();
            textBox9.Text = dr["Address"].ToString();
            textBox10.Text = dr["City"].ToString();
        }
    }

    private void update_btn_Click(object sender, EventArgs e)
    {
        if (textBox6.Text != "" && textBox7.Text != "" && textBox8.Text != "" &&
textBox9.Text != "" && textBox10.Text != "")
        {

            if (Validations(textBox6.Text) && Validations2(textBox8.Text) &&
Validations(textBox10.Text))
            {
```

```
            String q = "update dealer_info set
Dealer_name=@dn,Dealer_company_name=@dc,Contact=@contact,Address=@a
ddress,City=@city where id=" + id + "";
            SqlCommand cmd = new SqlCommand(q, con);
            cmd.Parameters.AddWithValue("@dn", textBox6.Text);
            cmd.Parameters.AddWithValue("@dc", textBox7.Text);
            cmd.Parameters.AddWithValue("@contact", textBox8.Text);
            cmd.Parameters.AddWithValue("@address", textBox9.Text);
            cmd.Parameters.AddWithValue("@city", textBox10.Text);
            con.Open();
            cmd.ExecuteNonQuery();
            con.Close();
            MessageBox.Show("Dealer details updated");
            display();
            textBox6.Clear();
            textBox7.Clear();
            textBox8.Clear();
            textBox9.Clear();
            textBox10.Clear();
        }
        else
        {
            MessageBox.Show("Enter dealer details properly");

        }
```

```
        }
        else
        {
            MessageBox.Show("Enter all the dealer details");
        }
      }
   }
}
```

Purchase_report.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace ADNF_casestudy
{
   public partial class Purchase_report : Form
```

```
{
    public Purchase_report()
    {
        InitializeComponent();
    }
    SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Welcome\Docume
nts\Inventory_Database.mdf;Integrated Security=True;Connect Timeout=30");

    private void view_all_btn_Click(object sender, EventArgs e)
    {

        decimal i = 0;
        String q = "select * from Purchase_master";
        SqlDataAdapter sda = new SqlDataAdapter(q, con);
        DataSet ds = new DataSet();
        sda.Fill(ds);
        dataGridView1.DataSource = ds.Tables[0];

        String q1 = "select Product_total from Purchase_master";
        SqlCommand cmd = new SqlCommand(q1, con);

        con.Open();
        SqlDataReader sdr = cmd.ExecuteReader();
        while (sdr.Read())
```

```
        {
            i = i + Convert.ToDecimal(sdr[0]);
        }
        con.Close();
        label3.Text = i.ToString();
    }

    private void search_btn_Click(object sender, EventArgs e)
    {
        string start_date, end_date;

        start_date = dateTimePicker1.Value.ToString("dd/MM/yyyy");
        end_date = dateTimePicker2.Value.ToString("dd/MM/yyyy");
        decimal i = 0;
        String q = "select * from Purchase_master where
Purchase_date>='"+start_date+"' AND Purchase_date<='"+end_date+"'";
        SqlDataAdapter sda = new SqlDataAdapter(q, con);
        DataSet ds = new DataSet();
        sda.Fill(ds);
        dataGridView1.DataSource = ds.Tables[0];

        String q1 = "select Product_total from Purchase_master where
Purchase_date>='"+start_date +"' AND Purchase_date<='"+end_date+"'";
        SqlCommand cmd = new SqlCommand(q1, con);
```

```
            con.Open();
            SqlDataReader sdr = cmd.ExecuteReader();
            while (sdr.Read())
            {
               i = i + Convert.ToDecimal(sdr[0]);
            }
            con.Close();
            label3.Text = i.ToString();
        }
    }
}
```

Stock_report.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
```

```
namespace ADNF_casestudy
{
    public partial class Stock_report : Form
    {
        public Stock_report()
        {
            InitializeComponent();
        }

        SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Welcome\Docume
nts\Inventory_Database.mdf;Integrated Security=True;Connect Timeout=30");

        private void Stock_report_Load(object sender, EventArgs e)
        {
            String q = "select * from Stock";
            SqlDataAdapter sda = new SqlDataAdapter(q, con);
            DataSet ds = new DataSet();
            sda.Fill(ds);
            dataGridView1.DataSource = ds.Tables[0];
        }
    }
}
```

Order_detail_report.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace ADNF_casestudy
{
   public partial class Order_detail_report : Form
   {
      public Order_detail_report()
      {
         InitializeComponent();
      }

      SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Welcome\Docume
nts\Inventory_Database.mdf;Integrated Security=True;Connect Timeout=30");
```

```
    private void view_all_btn_Click(object sender, EventArgs e)
    {

        decimal i = 0;
        String q = "select * from Order_Item oi left join Order_user ou on
oi.Order_id = ou.Id";
        SqlDataAdapter sda = new SqlDataAdapter(q, con);
        DataSet ds = new DataSet();
        sda.Fill(ds);
        dataGridView1.DataSource = ds.Tables[0];

        String q1 = "select Total from Order_Item oi left join Order_user ou on
oi.Order_id = ou.Id";
        SqlCommand cmd = new SqlCommand(q1, con);

        con.Open();
        SqlDataReader sdr = cmd.ExecuteReader();
        while (sdr.Read())
        {
           i = i + Convert.ToDecimal(sdr[0]);
        }
        con.Close();
        label3.Text = i.ToString();

    }
```

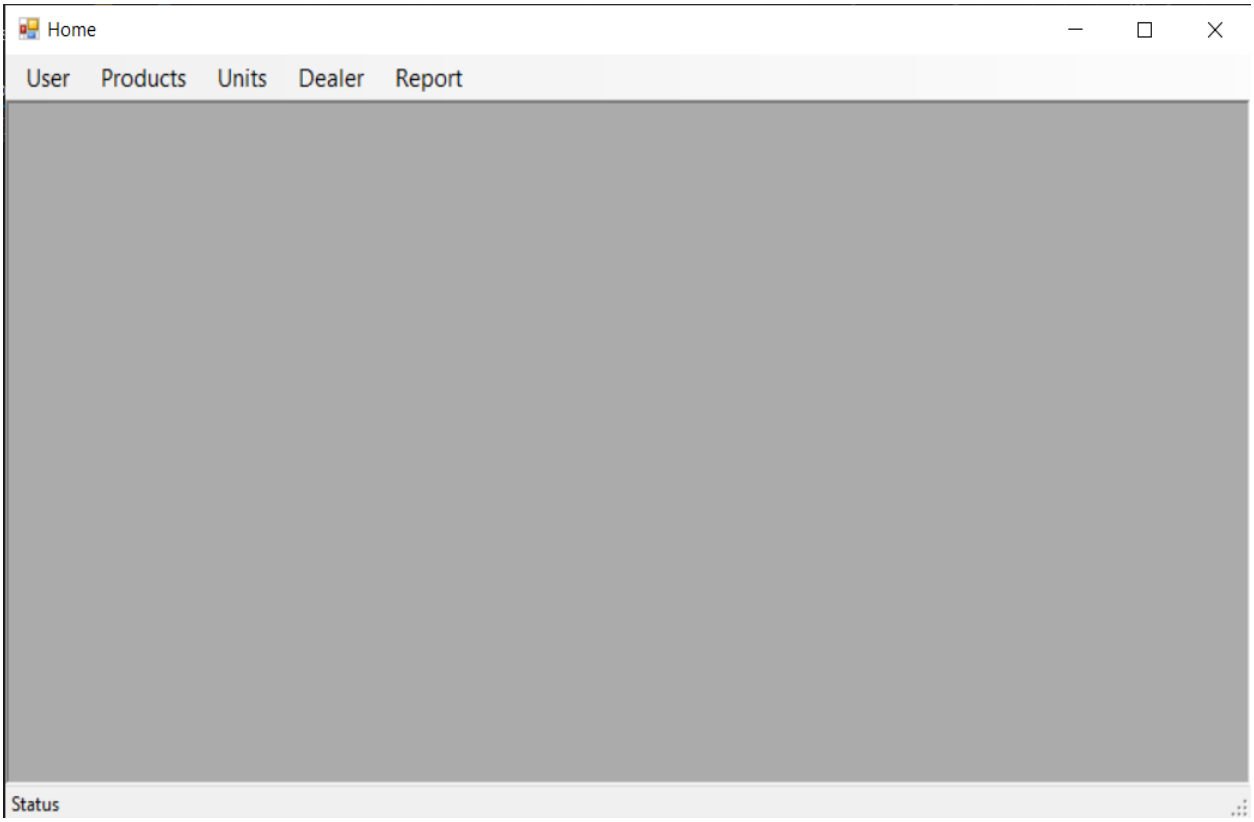
```
private void search_btn_Click(object sender, EventArgs e)
{
   if (comboBox1.Text != "")
   {
      decimal i = 0;
      String q = "select * from Order_Item i left join Order_user u on i.Order_id
= u.Id where u.Bill_type = '" + (comboBox1.SelectedItem).ToString() + "'";
      SqlDataAdapter sda = new SqlDataAdapter(q, con);
      DataSet ds = new DataSet();
      sda.Fill(ds);
      dataGridView1.DataSource = ds.Tables[0];

      String q1 = "select Total from Order_Item oi left join Order_user ou on
oi.Order_id = ou.Id";
      SqlCommand cmd = new SqlCommand(q1, con);

      con.Open();
      SqlDataReader sdr = cmd.ExecuteReader();
      while (sdr.Read())
      {
         i = i + Convert.ToDecimal(sdr[0]);
      }
      con.Close();
      label3.Text = i.ToString();
```

```
            }
            else
            {
                MessageBox.Show("Select Bill type");
            }
        }
    }
}
```

Description and Functionalities

User – Admin

- Primarily, the user who will interact with the system will be the administrator of the institution assigned to take care of all data transactions and insertion or update. It will have to go through an authorization process of login
- If the admin doesn't enter a value in any input space in the entire system it will display an error in the message box asking the admin to enter all the data
- If the admin enters numerical data in any input space where only alphabetical data is supposed to be supplied or vice versa then it will show error in the message box asking the admin to enter proper data



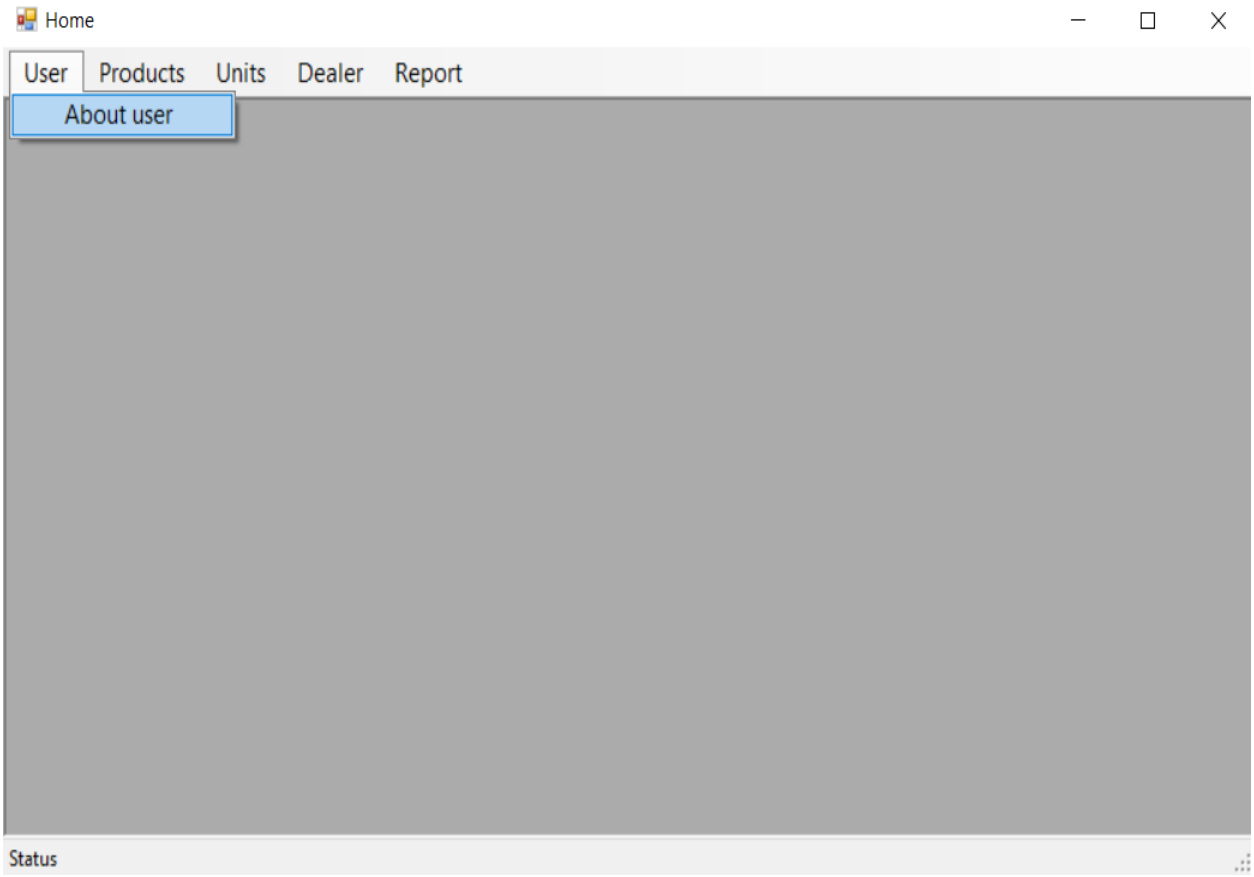
Home page:

- It is an MDI Parent form which is also a windows form
- Though this form, the user can access all the functionalities that this system provides.

Functionality 1:

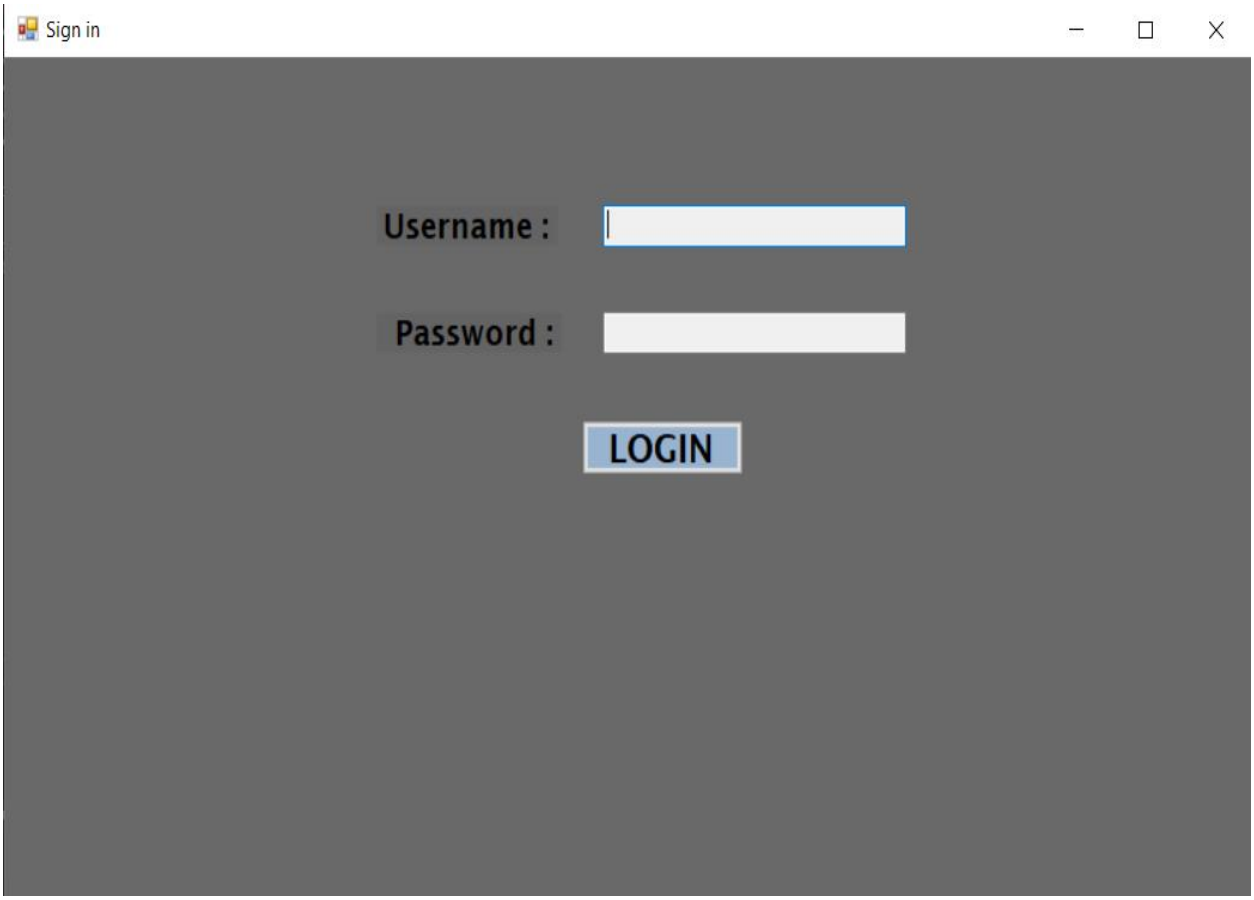
User-> About user

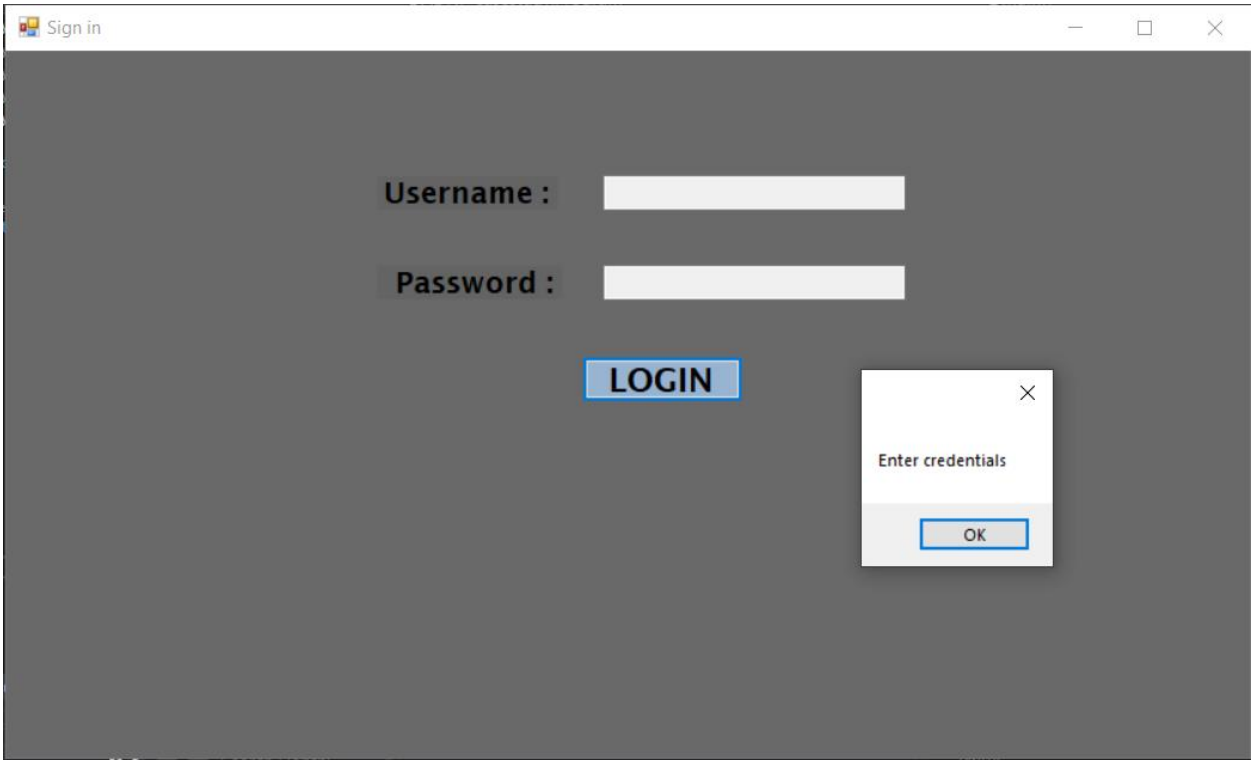
- It displays all the information of the admin that was already stored in “User_ detail” table in database

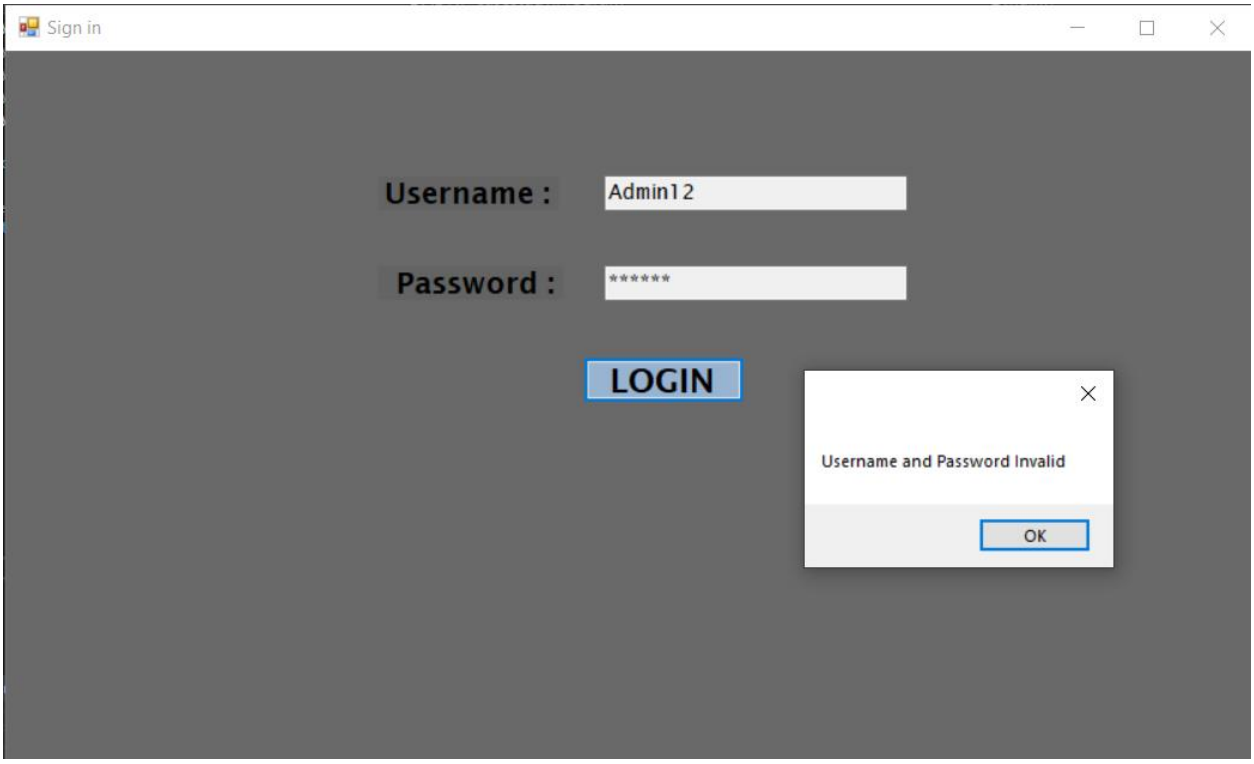


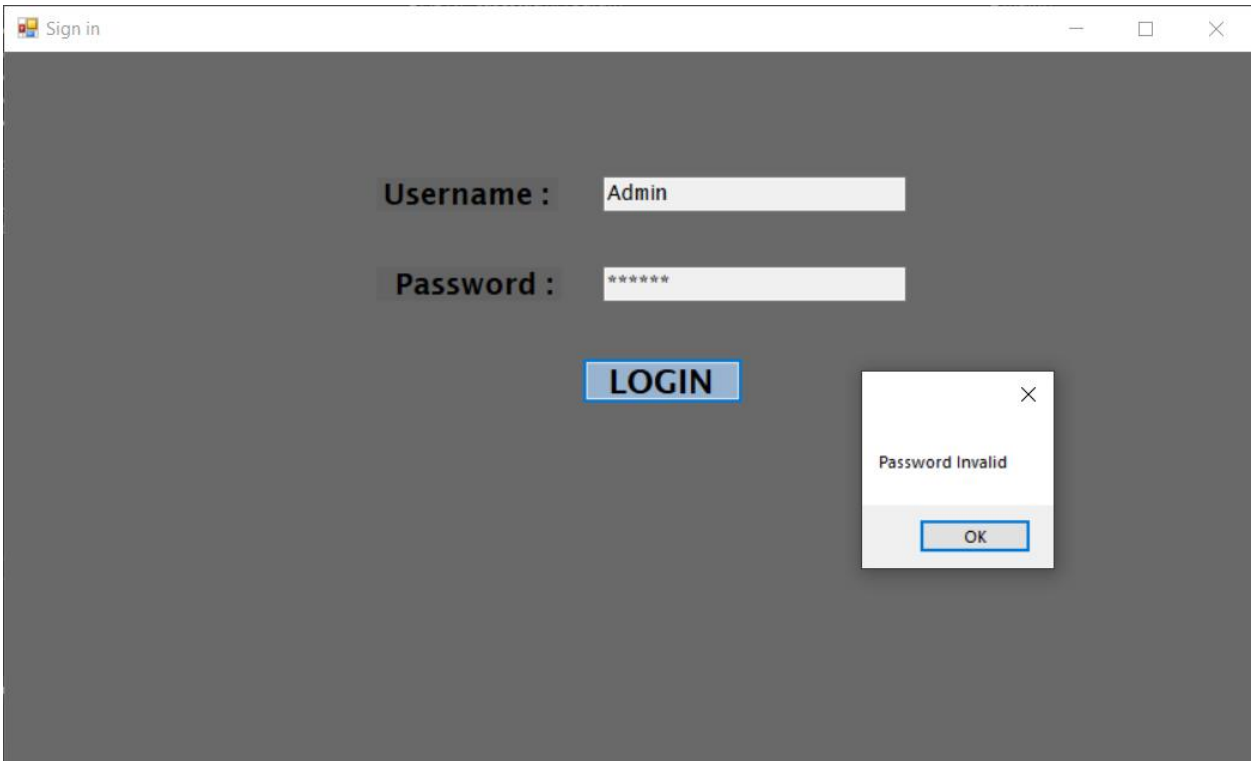
Login form:

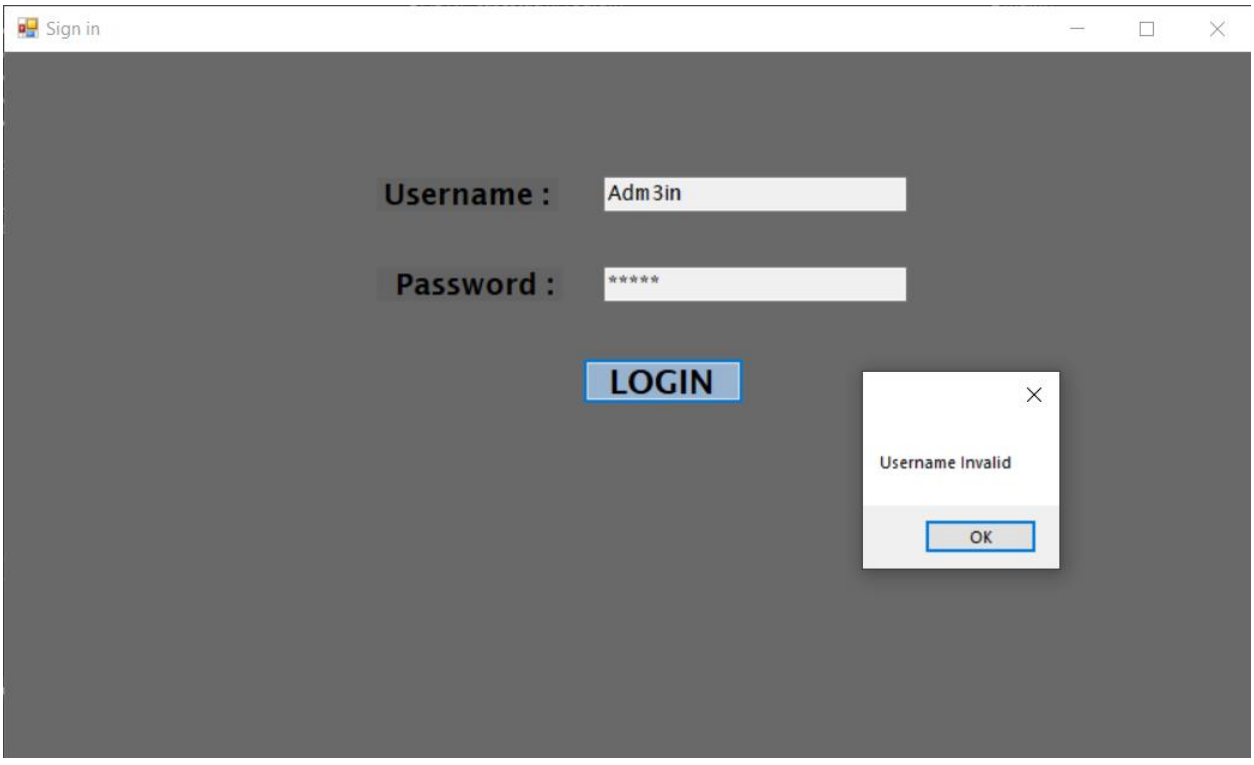
- Without being authenticated no user is allowed to view any other interfaces. After being authenticated user is authorized to perform certain work according to his/her profile.
- Admin has to enter correct username and password in order to enter next form.
- Username and password are saved in the database which is fetched and compared to the user input.
- Even one incorrect or empty input will show error message to user.




Sign in
Username :
Password :
LOGIN
Enter credentials
OK


Sign in
Username :
Admin12
Password :
******
LOGIN
Username and Password Invalid
OK

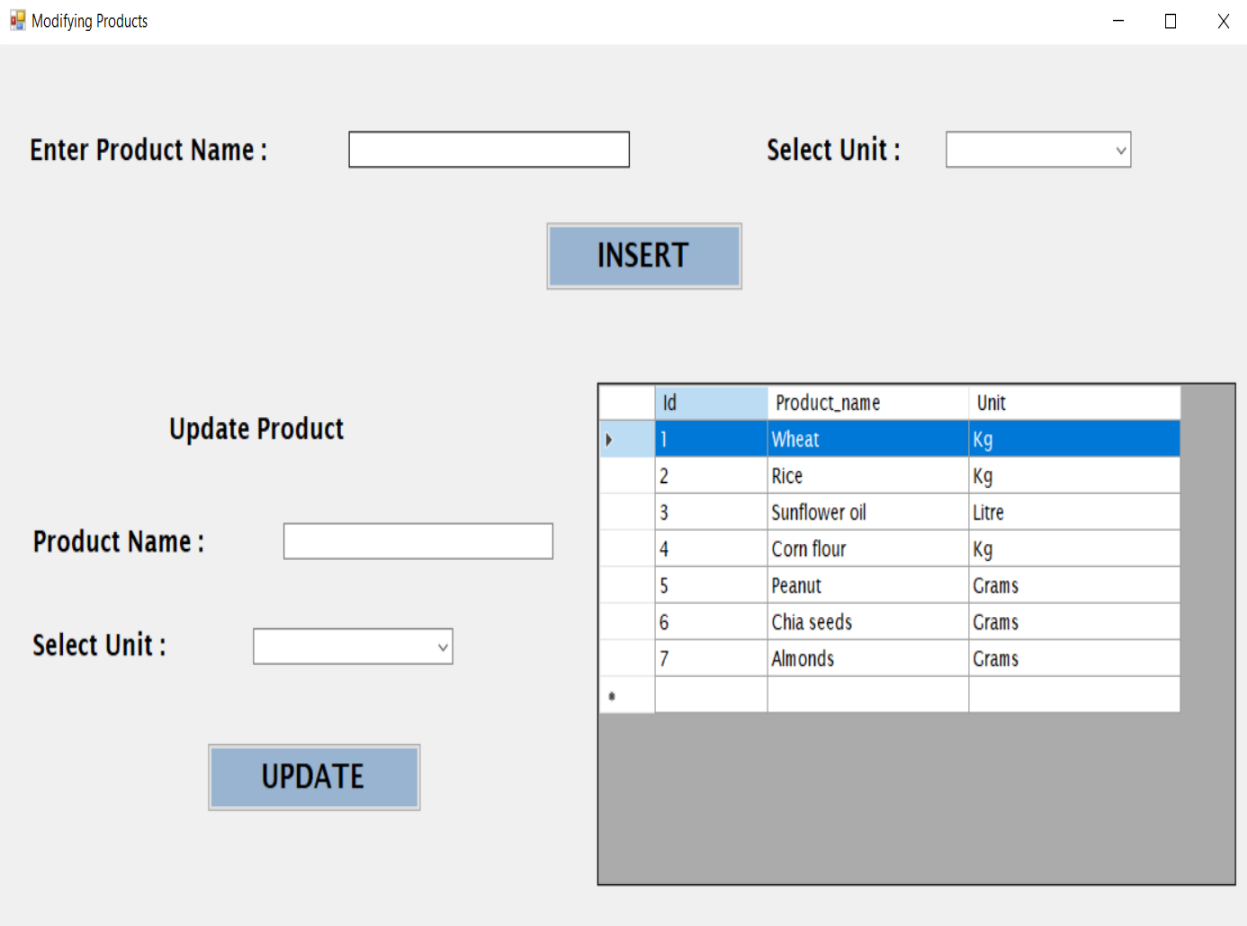

Sign in
Username :
Admin
Password :
******
LOGIN
Password Invalid
OK

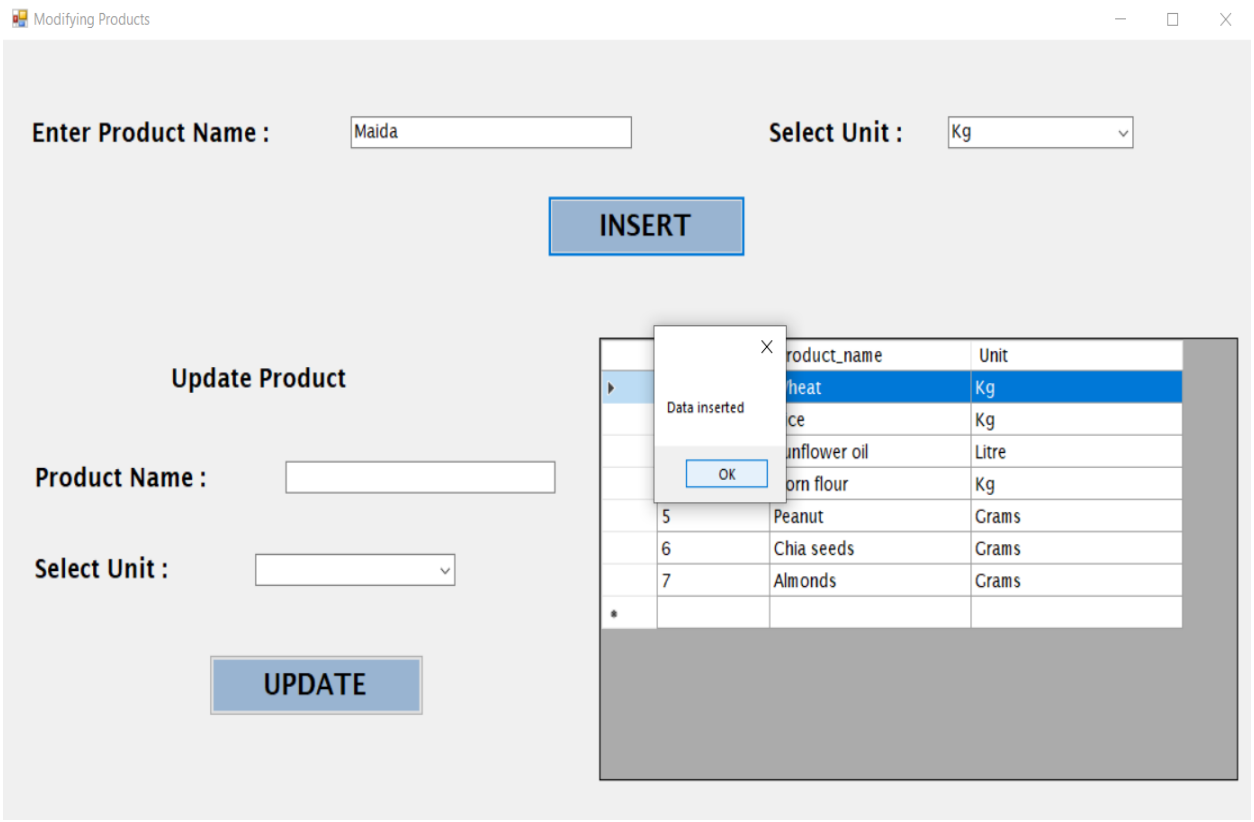

Sign in
Username :
Adm3in
Password :
*****
LOGIN
Username Invalid
OK

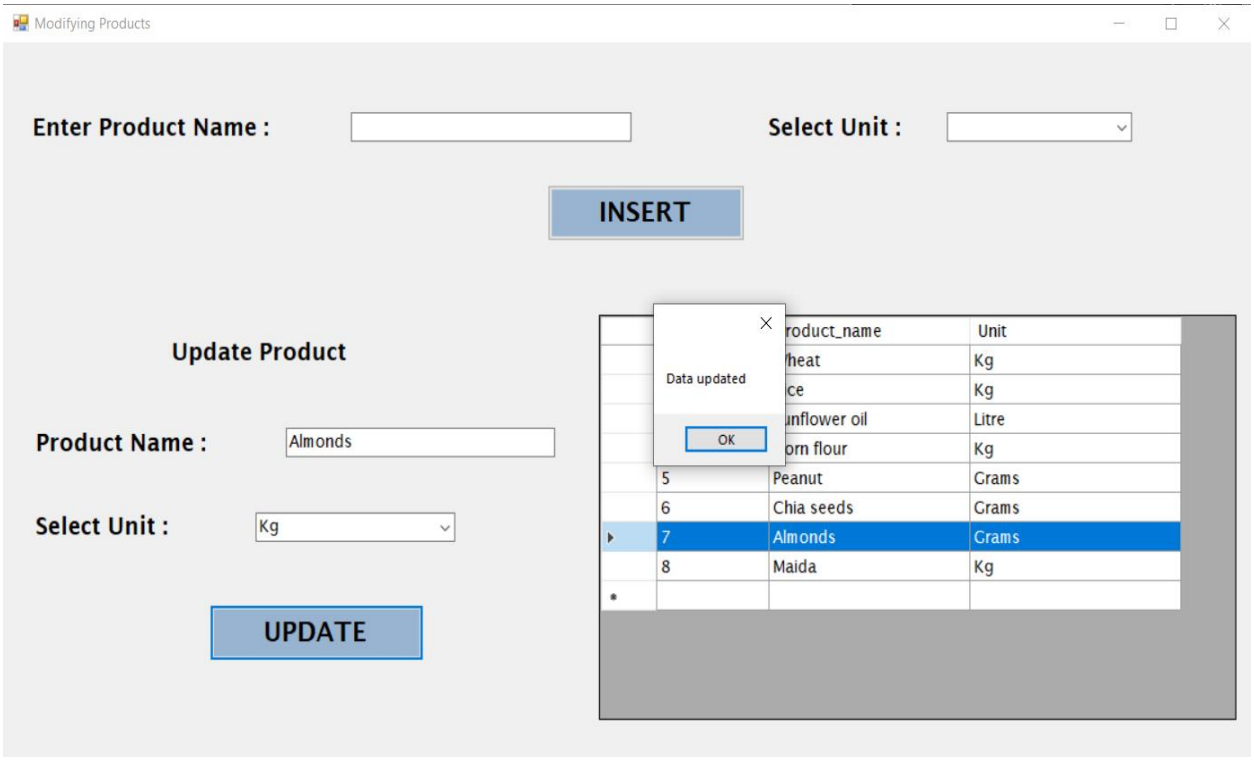
Functionality 2:

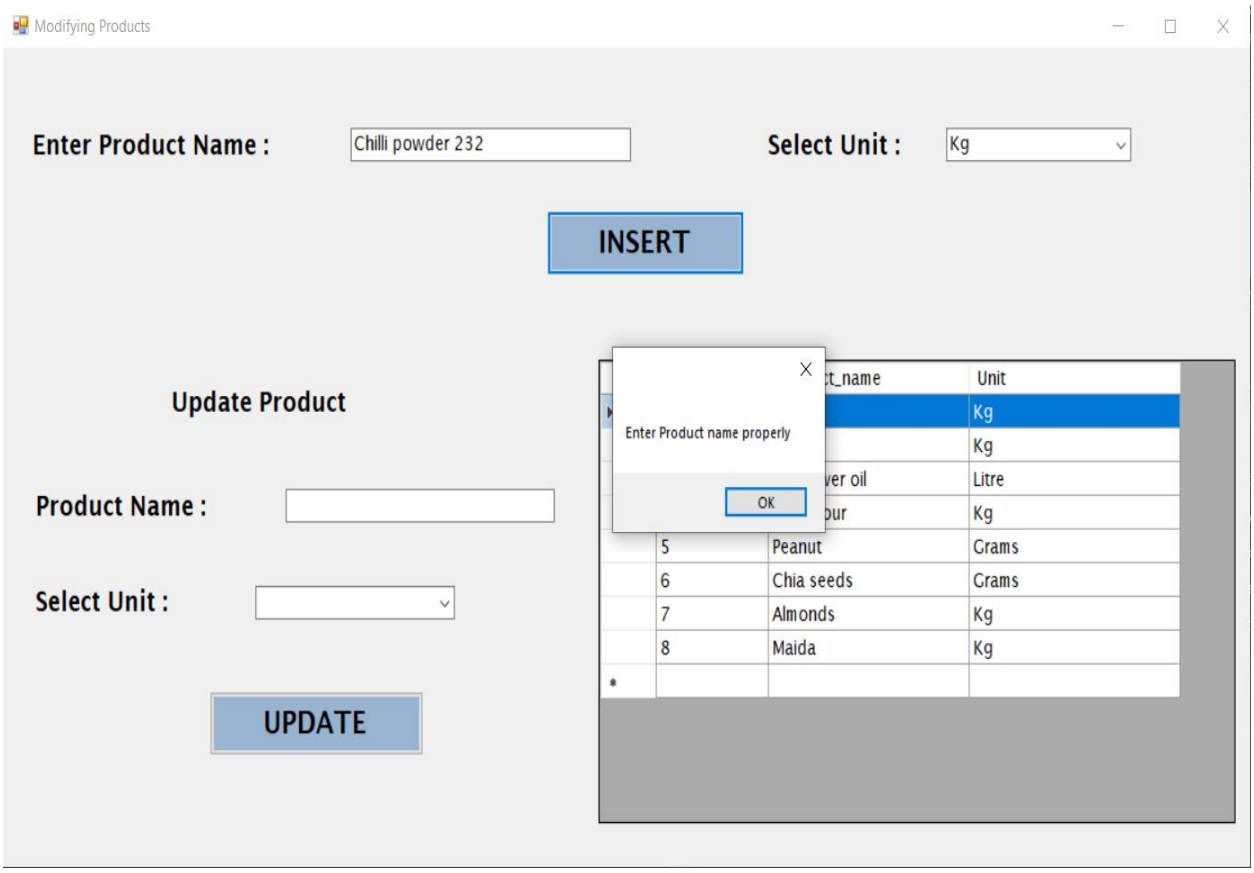
Products-> Modify products

- The user is allowed to insert any new product or update an existing one.
- When this form loads, all the products in the "Product_name" table in database is displayed in the dataGridView
- The user can input the product name and he/she can select the related unit from the combobox which stores all the units from the "Units" table in the database.
- The user can select a row from the datagridview and the data comes up in the textbox automatically and then the admin can make the necessary changes to it and click on update that will update the table and view it in the datagridview




Modifying Products
Enter Product Name :
Maida
Select Unit :
Kg
INSERT
Update Product
Product Name :
Select Unit :
UPDATE
Data inserted
OK
roduct_name
Unit
heat
Kg
ce
Kg
nflower oil
Litre
rn flour
Kg
5
Peanut
Grams
6
Chia seeds
Grams
7
Almonds
Grams

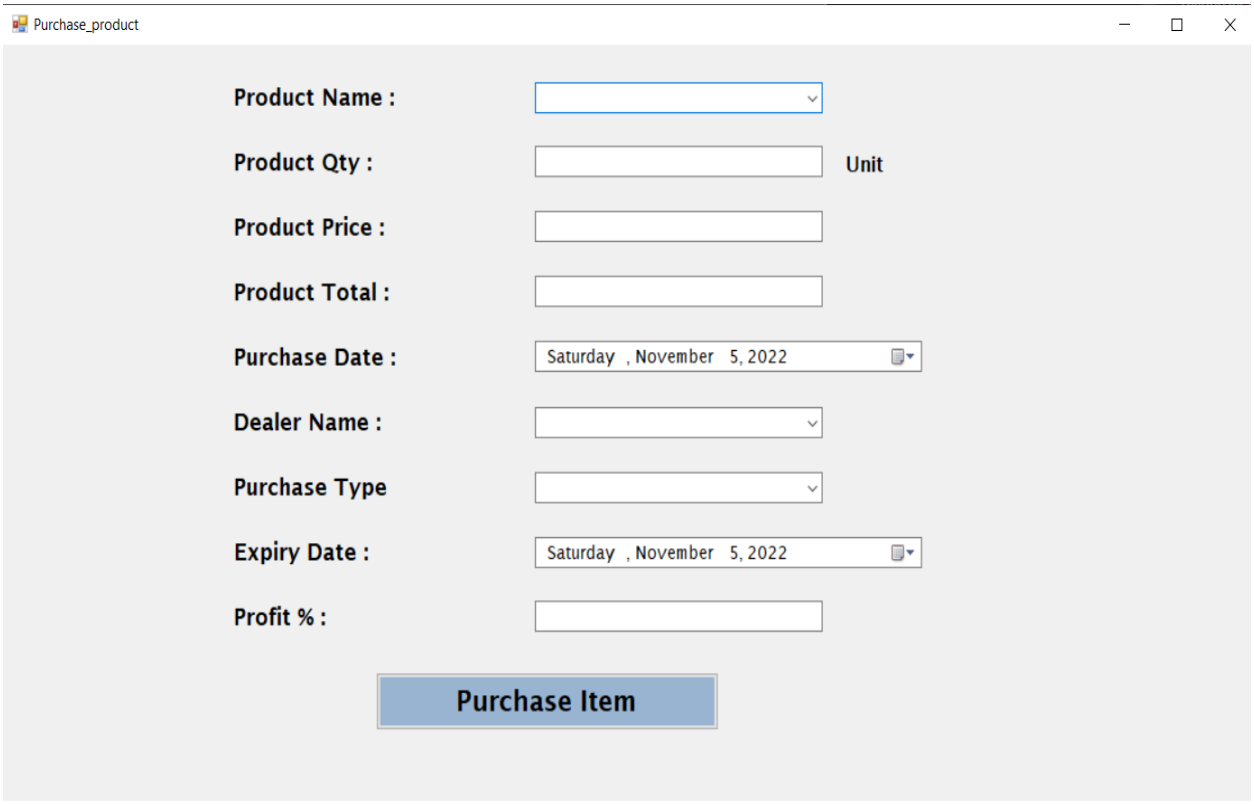

Modifying Products
Enter Product Name :
Select Unit :
INSERT
Update Product
Product Name :
Almonds
Select Unit :
Kg
UPDATE
Data updated
OK
roduct_name
Unit
heat
Kg
ce
Kg
nflower oil
Litre
rn flour
Kg
5
Peanut
Grams
6
Chia seeds
Grams
7
Almonds
Grams
8
Maida
Kg

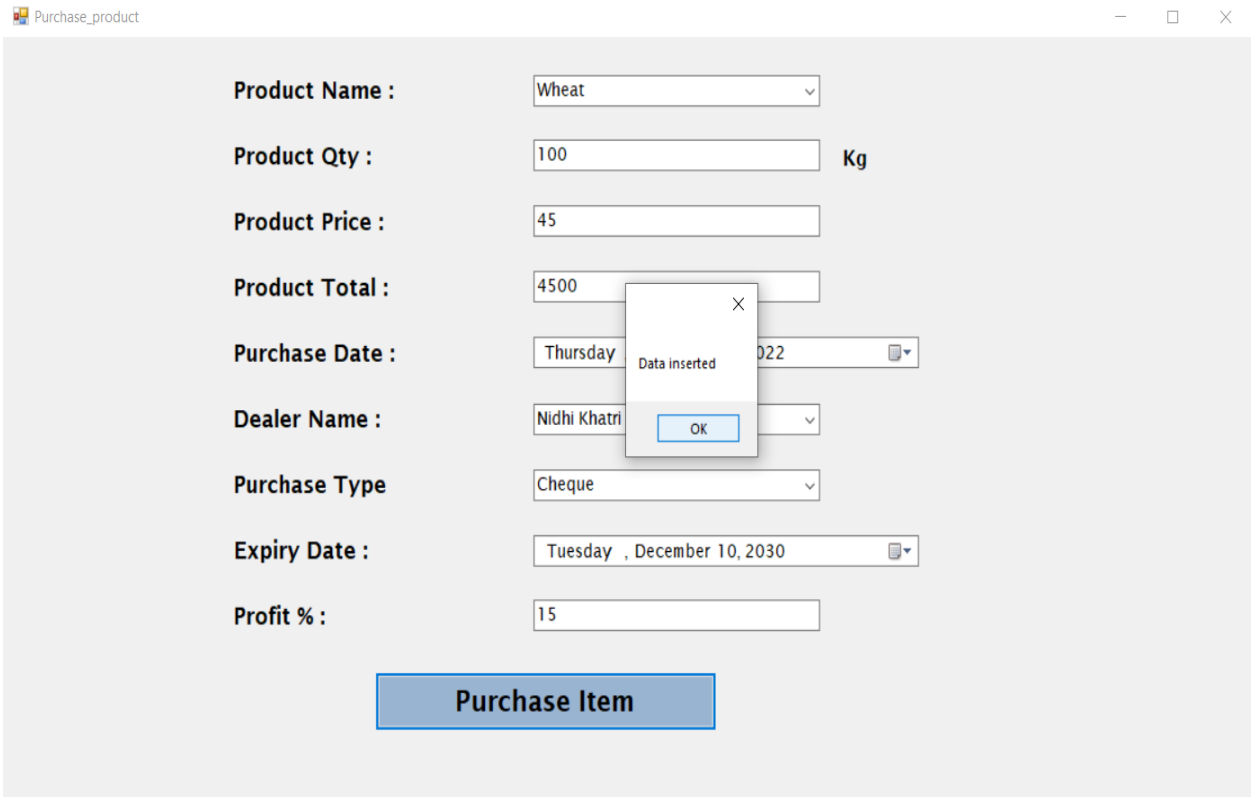

Modifying Products
Enter Product Name :
Chilli powder 232
Select Unit :
Kg
INSERT
Update Product
Product Name :
Select Unit :
UPDATE
Enter Product name properly
OK
t_name
Unit
Kg
Kg
ver oil
Litre
pur
Kg
5
Peanut
Grams
6
Chia seeds
Grams
7
Almonds
Kg
8
Maida
Kg

Functionality 3:

Products-> Purchase products

- This form is used when the admin has to enter the data of the products that he has purchased from the dealer.
- It inserts/updates the data in the “Purchase_master” and “Stock” table in database.
- Admin can :
- Enter the product name from the combobox which stores all the data from “Product_name” table depending on which, the unit will be displayed on its own.
- After adding quantity and price, the total cost will be calculated on its own
- Then add the purchase date from the dateTimePicker.
- Then choose the dealer’s name from the combobox which stores the dealer names from the table “Dealer_info” table in database
- Choose the purchase type (cash/upi/debit card etc.) from combobox
- Enter the expiry date of product using dateTimePicker.
- Enter the profit gained in the purchase
- Click on the Purchase button.


Purchase_product
Product Name :
Product Qty :
Unit
Product Price :
Product Total :
Purchase Date :
Saturday , November 5, 2022
Dealer Name :
Purchase Type
Expiry Date :
Saturday , November 5, 2022
Profit % :
Purchase Item

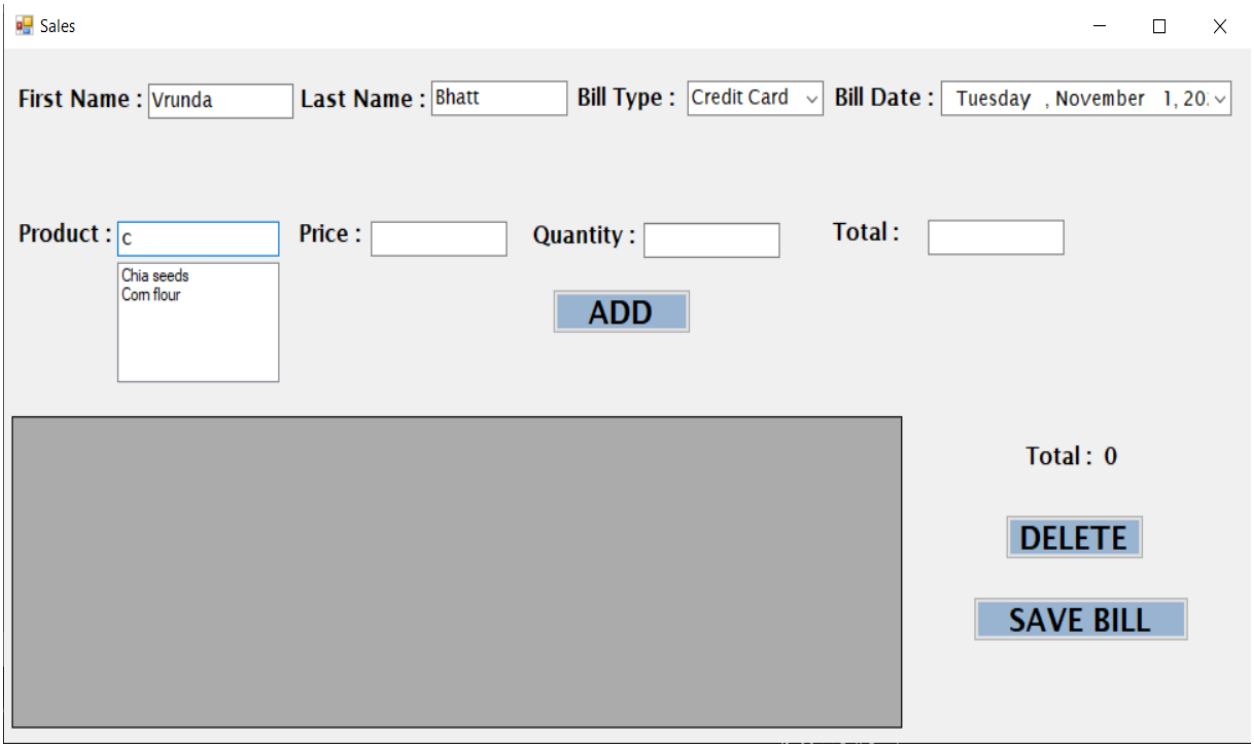

Purchase_product
Product Name :
Wheat
Product Qty :
100
Kg
Product Price :
45
Product Total :
4500
Purchase Date :
Thursday
022
Dealer Name :
Nidhi Khatri
Purchase Type
Cheque
Expiry Date :
Tuesday , December 10, 2030
Profit % :
15
Purchase Item
Data inserted
OK

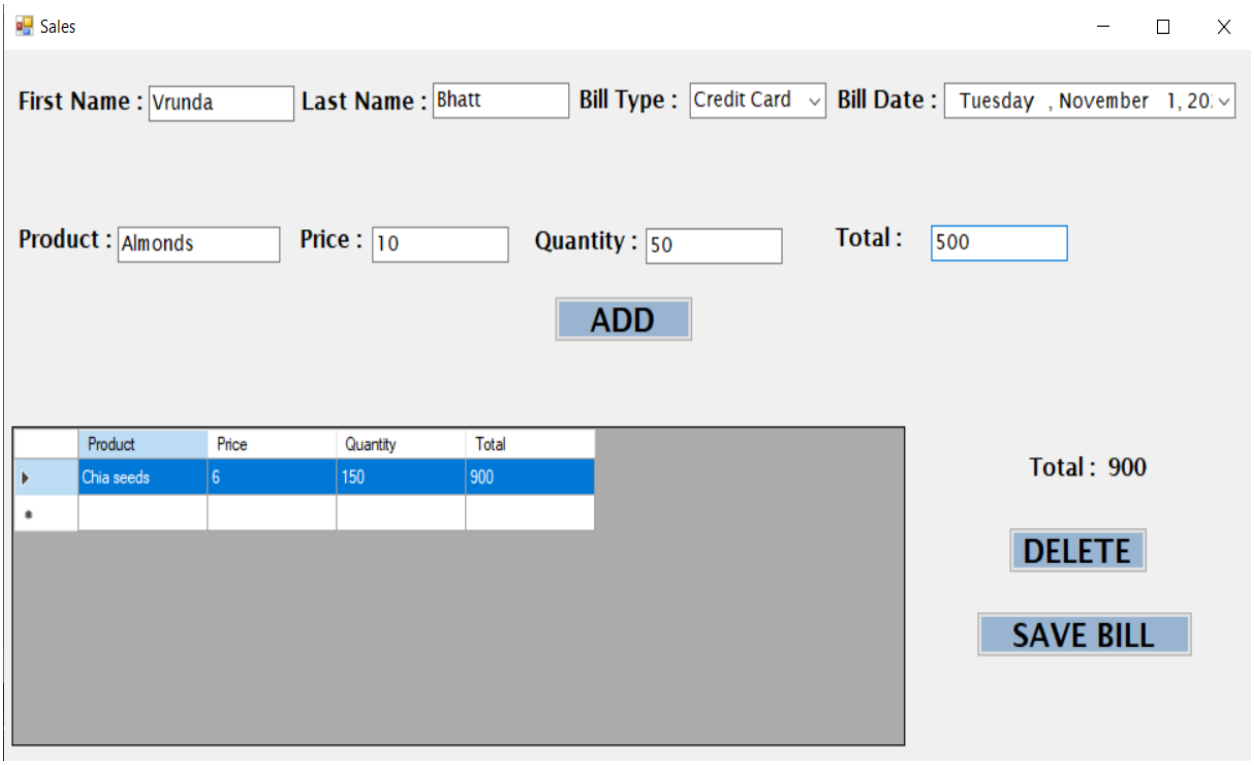
Functionality 4:

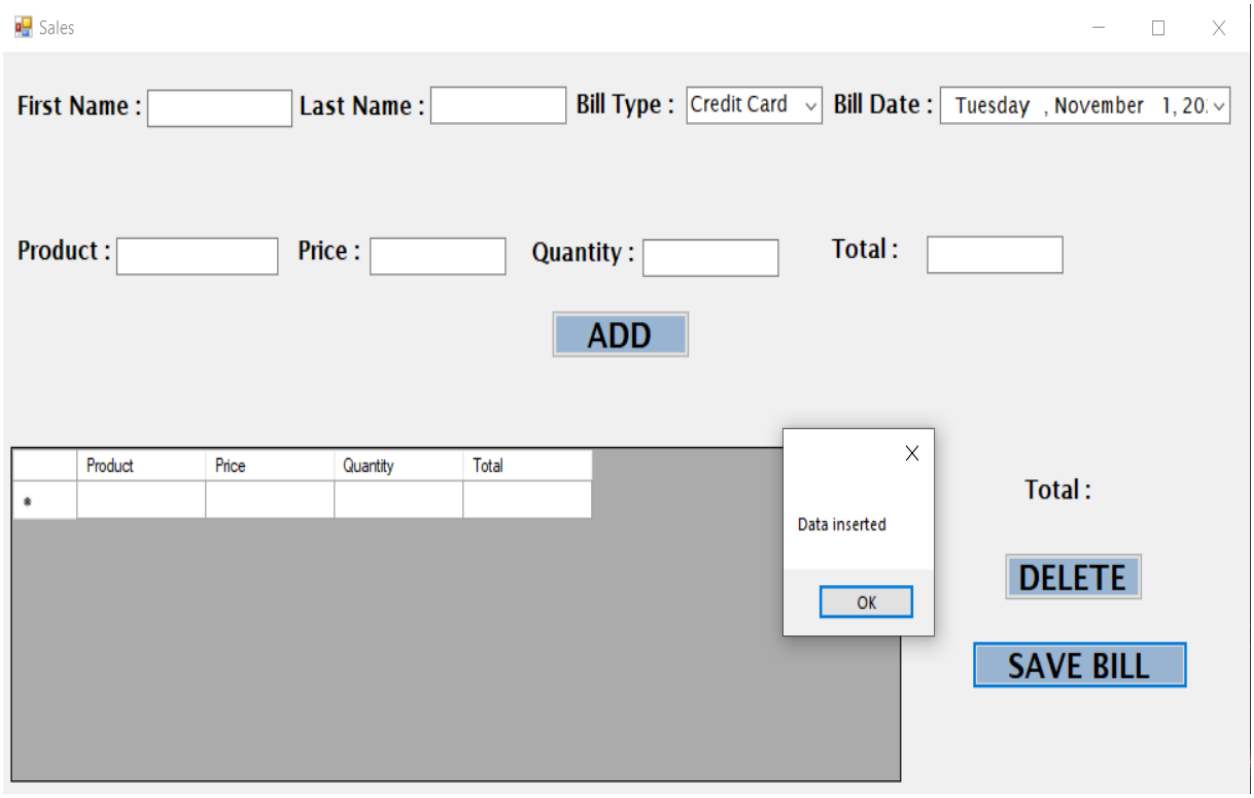
Products-> Sales products

- This form is used when the admin has to create bill of all products sold to the consumer.
- It enters data in “Order_Item” and “Order_user” and update the product quantity in the “Stock” table.
- Admin can
- Enter the consumer details like first name, last name, choose the bill type (cash, upi, debit card, etc.) and select the bill date from dateTimePicker.
- Enter the first letter and the listbox will display all the Product_name from the “Stock” table that starts with that letter and the admin can select from that list.
- The price field will be auto filled with the Product_price from the Purchase_master table in database depending on the product name selected.
- Enter the product quantity that the consumer wants.
- The total will be auto filled with the product of the quantity and price.
- Click on Add button to add the item to the dataGridView and continue to add all the items that the consumer wants one by one.
- Every time a product is added, the total will keep on updating the total price of all the products.
- Select a row in the dataGridView and delete it by clicking on Delete button.
- After properly creating the table in front end, save the bill by clicking on the SAVE BILL button. This action will insert the consumer details in the “Order_user” table, all the product details in the “Order_Item” table and update the product_qty of the products bought in the “Stock” table.


Sales
First Name :
Last Name :
Bill Type :
Bill Date :
Saturday , November 5, 20.
Product :
Price :
Quantity :
Total :
ADD
Total : 0
DELETE
SAVE BILL


Sales
First Name :
Vrunda
Last Name :
Bhatt
Bill Type :
Credit Card
Bill Date :
Tuesday , November 1, 20.
Product :
c
Chia seeds
Corn flour
Price :
Quantity :
Total :
ADD
Total : 0
DELETE
SAVE BILL

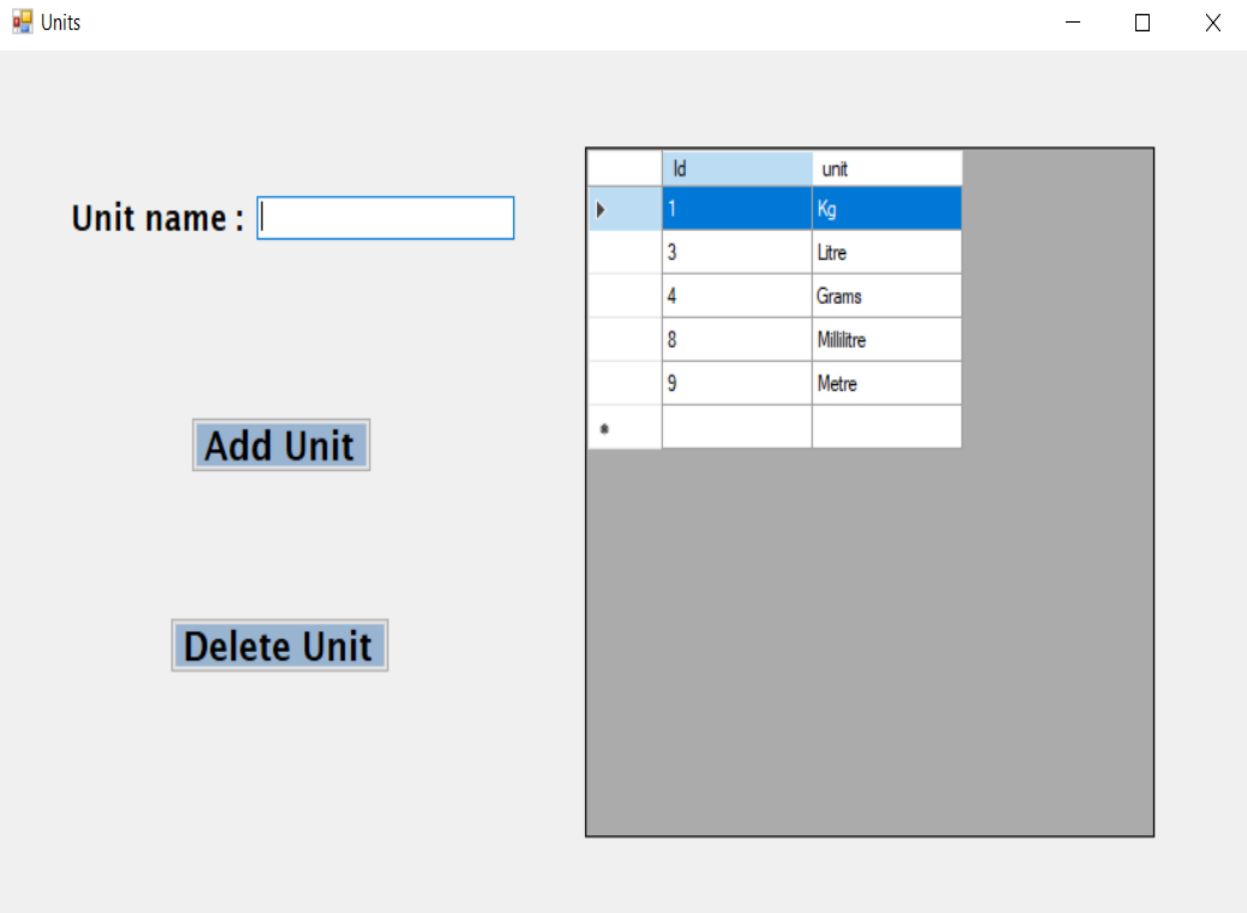

Sales
First Name : Vrunda
Last Name : Bhatt
Bill Type : Credit Card
Bill Date : Tuesday , November 1, 20
Product : Almonds
Price : 10
Quantity : 50
Total : 500
ADD
Product
Price
Quantity
Total
Chia seeds
6
150
900
Total : 900
DELETE
SAVE BILL

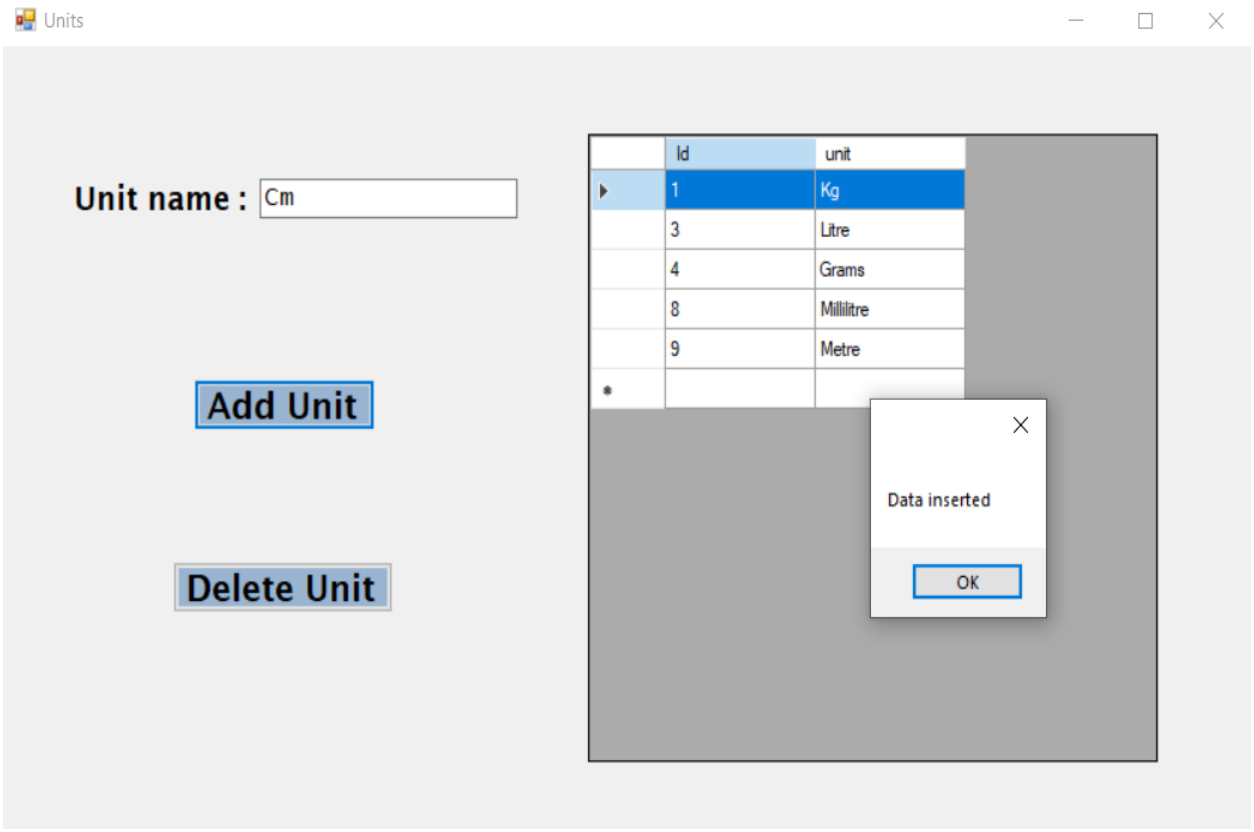

Sales
First Name :
Last Name :
Bill Type : Credit Card
Bill Date : Tuesday , November 1, 20
Product :
Price :
Quantity :
Total :
ADD
Product
Price
Quantity
Total
Data inserted
OK
Total :
DELETE
SAVE BILL

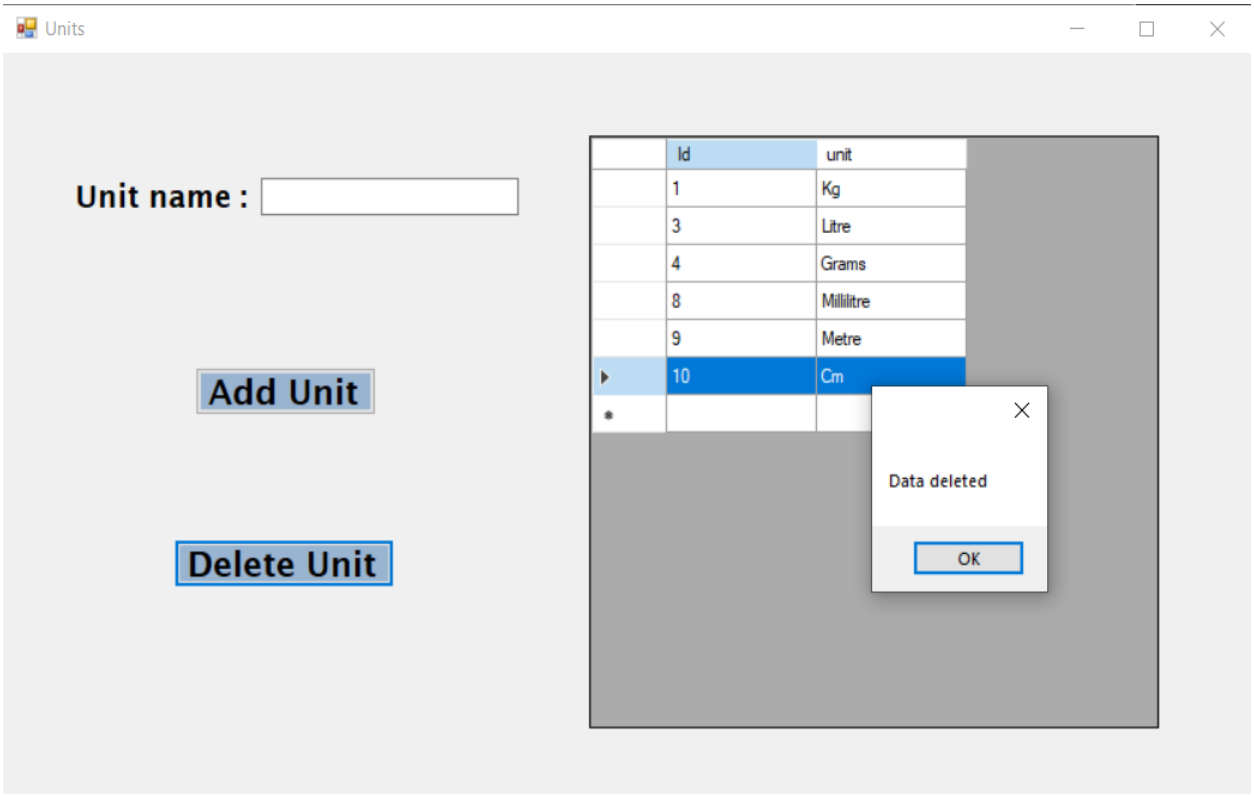
Functionality 5:

Units -> Modify units

- Admin can modify the “Units” table in database.
- The already existing units is displayed in the dataGridView when the form loads.
- Click on INSERT button after entering a new unit which will be inserted into the “Units” table and can viewed in the dataGridView
- Click on the delete button after selecting a row from the dataGridView



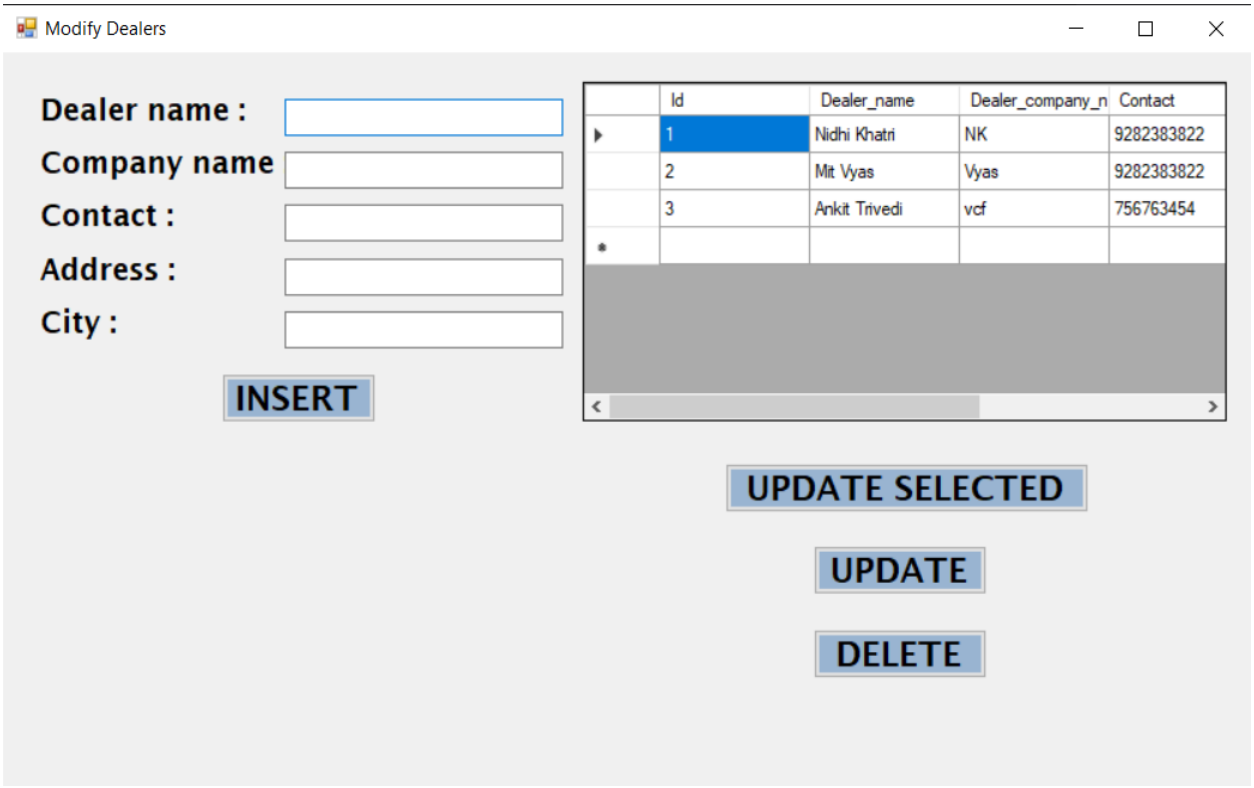

Units
Unit name :
Cm
Add Unit
Delete Unit
Id
unit
1
Kg
3
Litre
4
Grams
8
Millilitre
9
Metre
Data inserted
OK

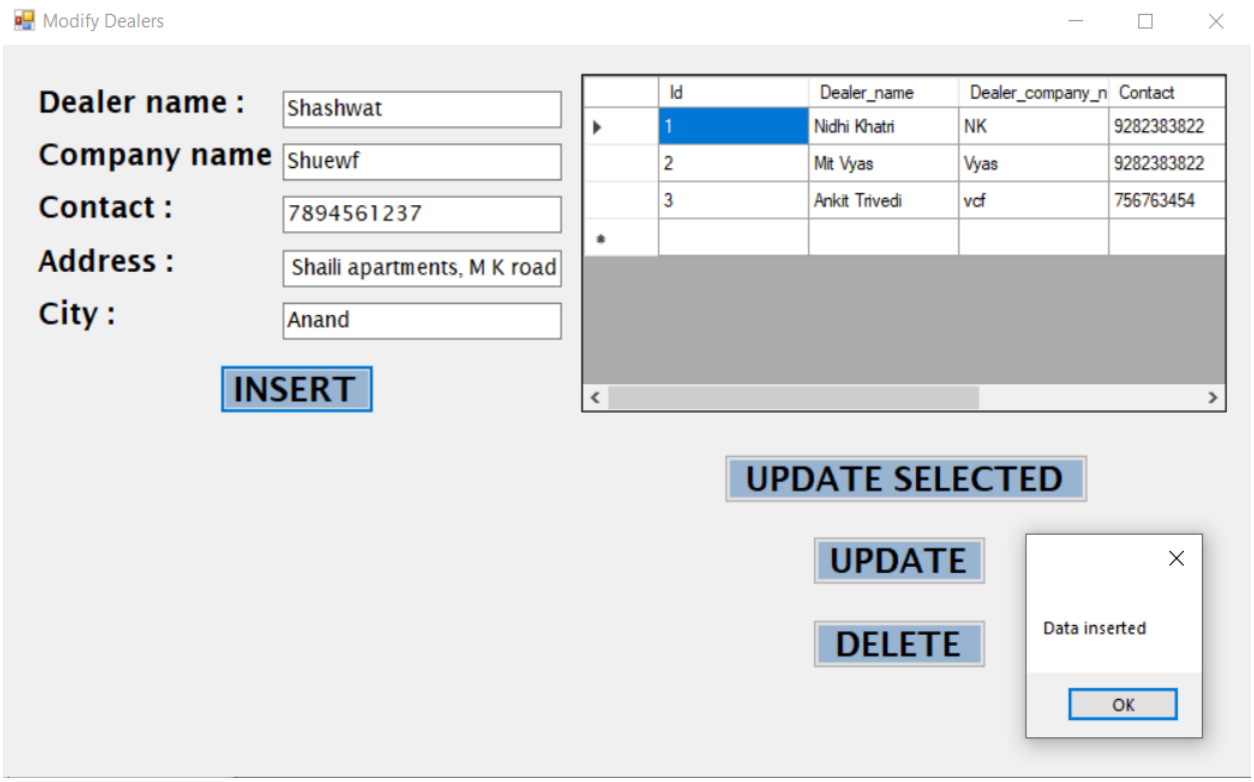

Units
Unit name :
Add Unit
Delete Unit
Id
unit
1
Kg
3
Litre
4
Grams
8
Millilitre
9
Metre
10
Cm
Data deleted
OK

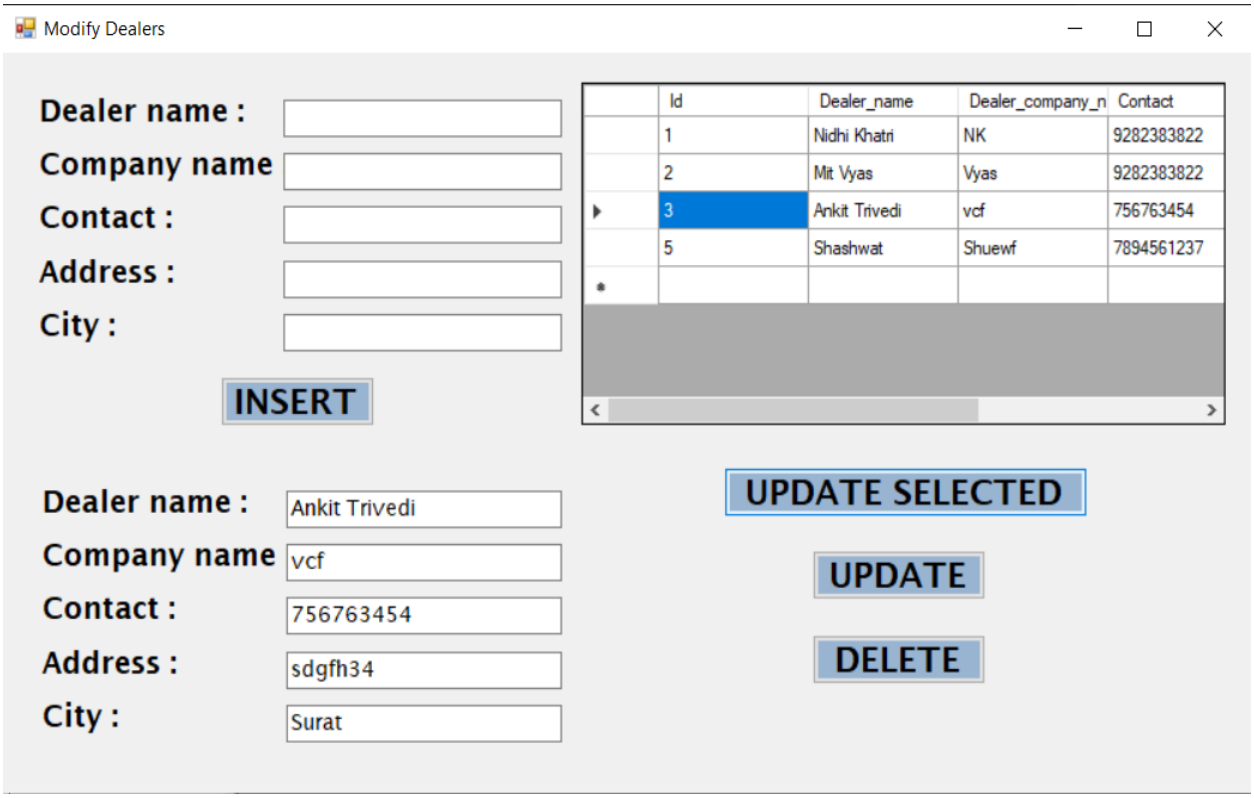
Functionality 6:

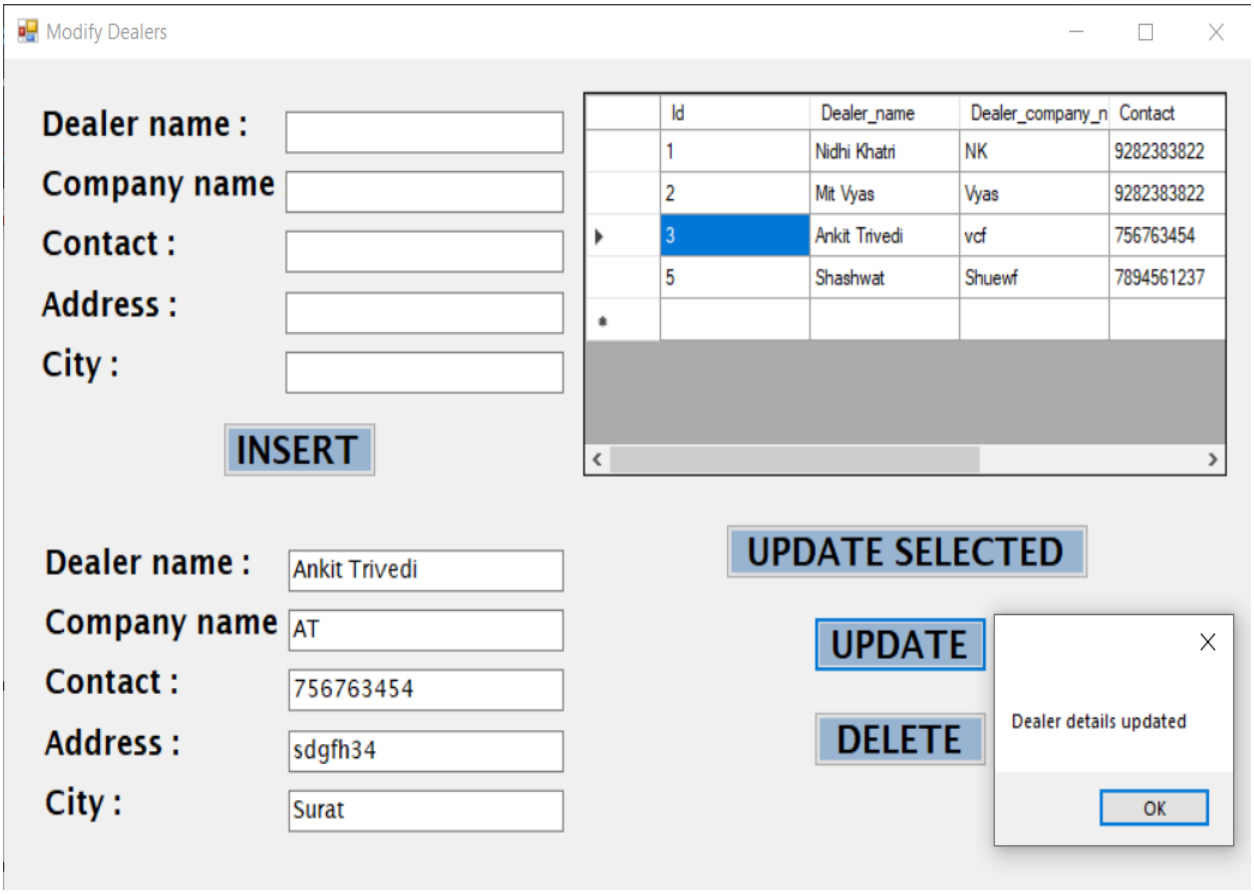
Dealer -> Modify dealers

- Admin can modify the “Dealer_info” table in database
- The already existing Dealer info is displayed in the dataGridView when the form loads.
- Click on INSERT button after entering dealer details which will be inserted into the “Dealer_info” table and can viewed in the dataGridView
- Select a row from the dataGridView and click on Update Selected button which will display a panel with all the details of the dealer selected on which the necessary changes can be made and then updated by clicking on Update button.
- Click on the delete button after selecting a row from the dataGridView




Modify Dealers
Dealer name :
Shashwat
Company name
Shuewf
Contact :
7894561237
Address :
Shaili apartments, M K road
City :
Anand
INSERT
Id
Dealer_name
Dealer_company_n
Contact
1
Nidhi Khatri
NK
9282383822
2
Mit Vyas
Vyas
9282383822
3
Ankit Trivedi
vcf
756763454
UPDATE SELECTED
UPDATE
DELETE
Data inserted
OK

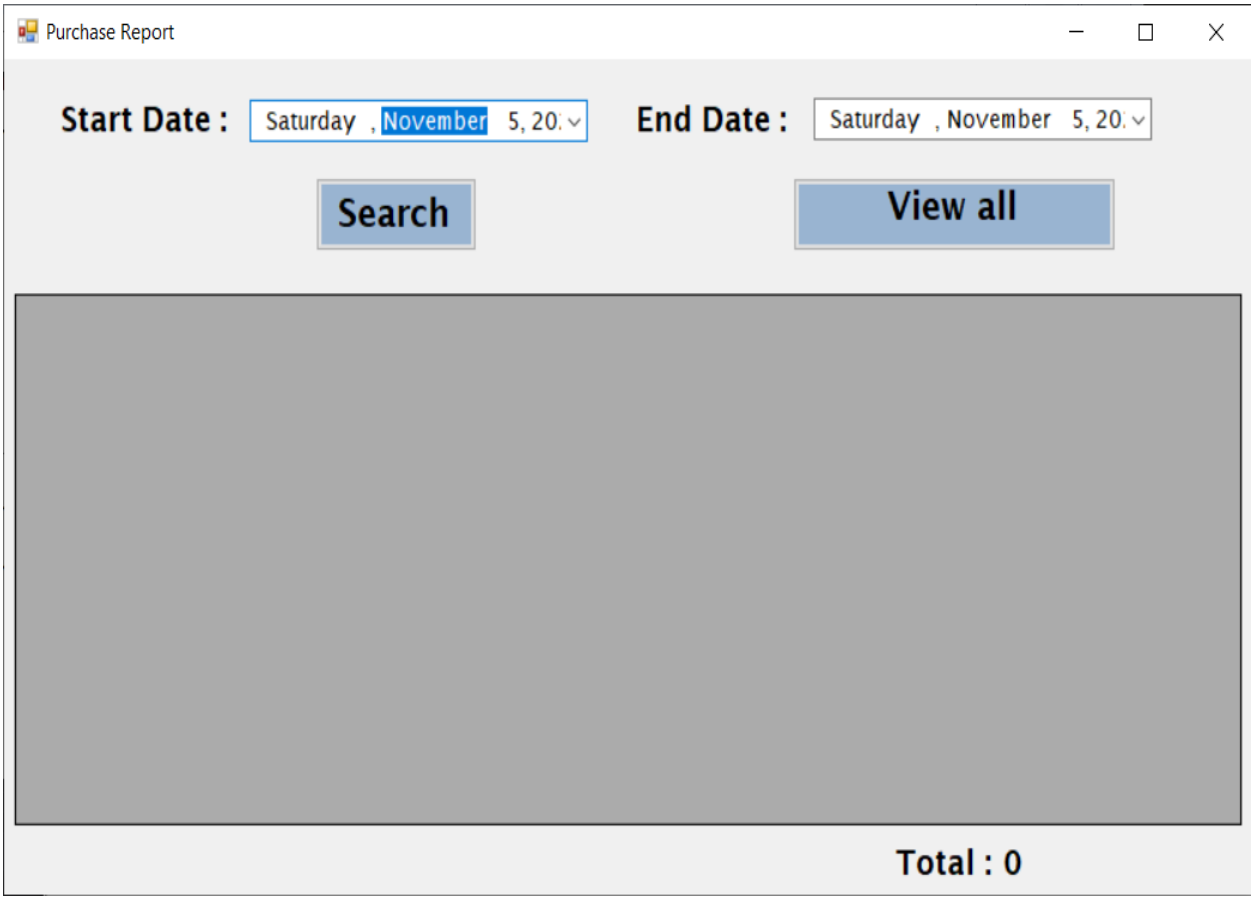

Modify Dealers
Dealer name :
Company name
Contact :
Address :
City :
INSERT
Id
Dealer_name
Dealer_company_n
Contact
1
Nidhi Khatri
NK
9282383822
2
Mit Vyas
Vyas
9282383822
3
Ankit Trivedi
vcf
756763454
5
Shashwat
Shuewf
7894561237
UPDATE SELECTED
Dealer name :
Ankit Trivedi
Company name
vcf
Contact :
756763454
Address :
sdgfh34
City :
Surat
UPDATE
DELETE


Modify Dealers
Dealer name :
Company name
Contact :
Address :
City :
INSERT
Id
Dealer_name
Dealer_company_n
Contact
1
Nidhi Khatri
NK
9282383822
2
Mit Vyas
Vyas
9282383822
3
Ankit Trivedi
vcf
756763454
5
Shashwat
Shuewf
7894561237
Dealer name :
Ankit Trivedi
Company name
AT
Contact :
756763454
Address :
sdgfh34
City :
Surat
UPDATE SELECTED
UPDATE
DELETE
Dealer details updated
OK

Functionality 7:

Report -> Purchase Report

- It shows the report of all the purchases that the institution has made from the dealer from the “Purchase_master” table in database
- Select a start date and end date from the dateTimePicker and then click on Search button which will display all items between those dates.
- View the entire purchase history by clicking on View all purchase button
- The total of all product prices that is displayed in the dataGridView can be viewed under the report



Purchase Report

Start Date : Saturday , November 5, 20 **End Date :** Saturday , November 5, 20

Search **View all**

	Id	Product_name	Product_qty	Product_unit	Product_price	Product_total	Purchase_date	Purchace
▶	1	Wheat	50	Kg	24	1200	21-10-2022	Nidhi Khat
	2	Rice	35	Kg	37	1295	30-10-2022	Mit Vyas
	3	Rice	45	Kg	39	1755	09-08-2022	Mit Vyas
	4	Sunflower oil	25	Litre	190	4750	05-07-2022	Nidhi Khat
	5	Peanut	500	Grams	0.06	30.00	17-08-2022	Nidhi Khat
	6	Wheat	3	Kg	30	90	31-10-2022	Nidhi Khat
	7	Rice	23	Kg	35	805	25-10-2022	Mit Vyas
	8	Sunflower oil	5	Litre	100	500	12-06-2022	Nidhi Khat
	9	Rice	40	Kg	39	1560	03-11-2022	Ankit Trive
	10	Almonds	900	Grams	10	9000	01-11-2022	Ankit Trive
	11	Almonds	900	Grams	10	9000	01-11-2022	Ankit Trive

Total : 39985.00

Purchase Report

Start Date : Tuesday , November 1, 20 **End Date :** Friday , November 4, 20

Search **View all**

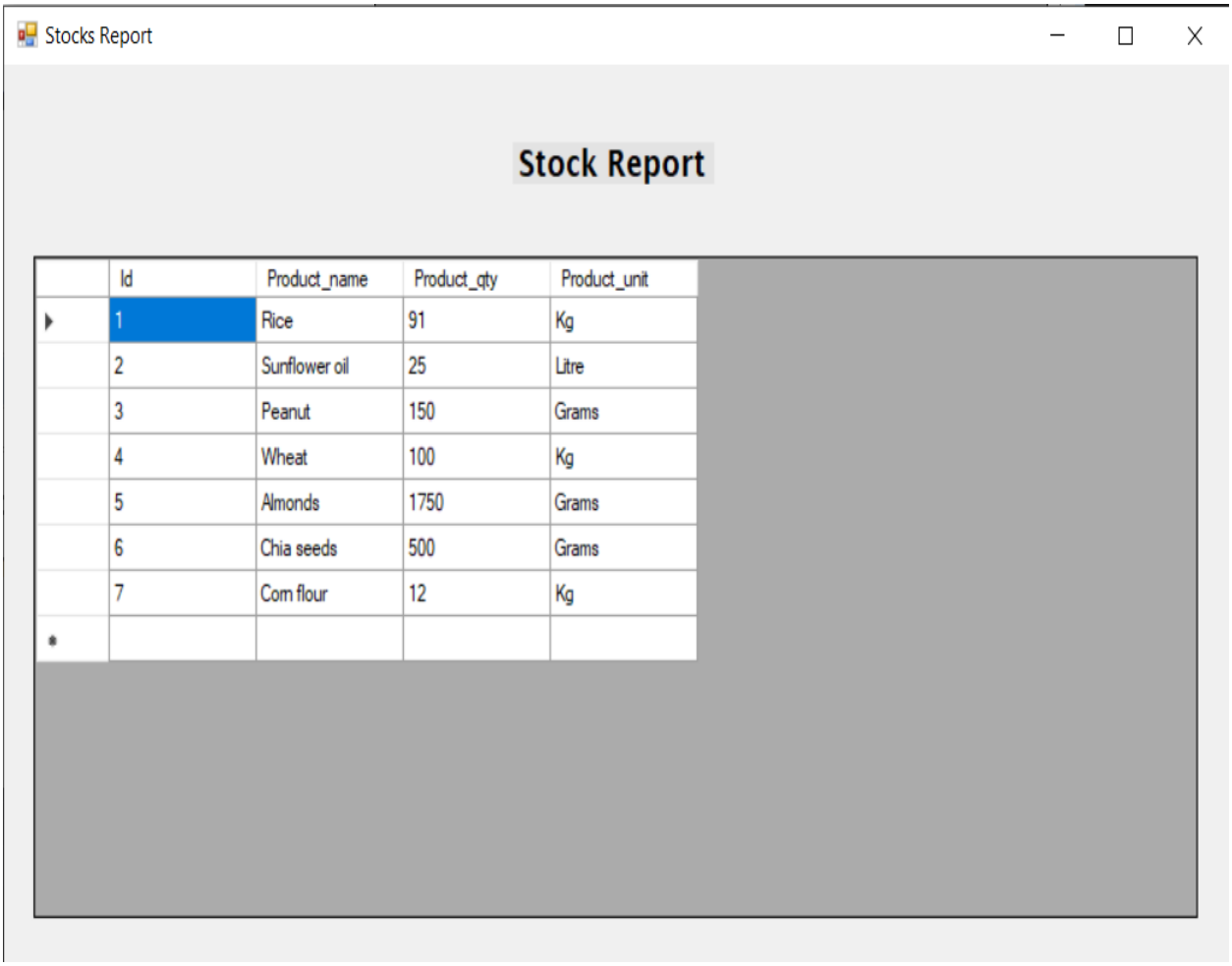
	Id	Product_name	Product_qty	Product_unit	Product_price	Product_total	Purchase_date	Purchace_pa
▶	9	Rice	40	Kg	39	1560	03-11-2022	Ankit Trivedi
	12	Chia seeds	750	Grams	6	4500	04-11-2022	Mit Vyas
	14	Wheat	100	Kg	45	4500	03-11-2022	Nidhi Khatri
*								

Total : 10560

Functionality 8:

Report -> Stocks Report

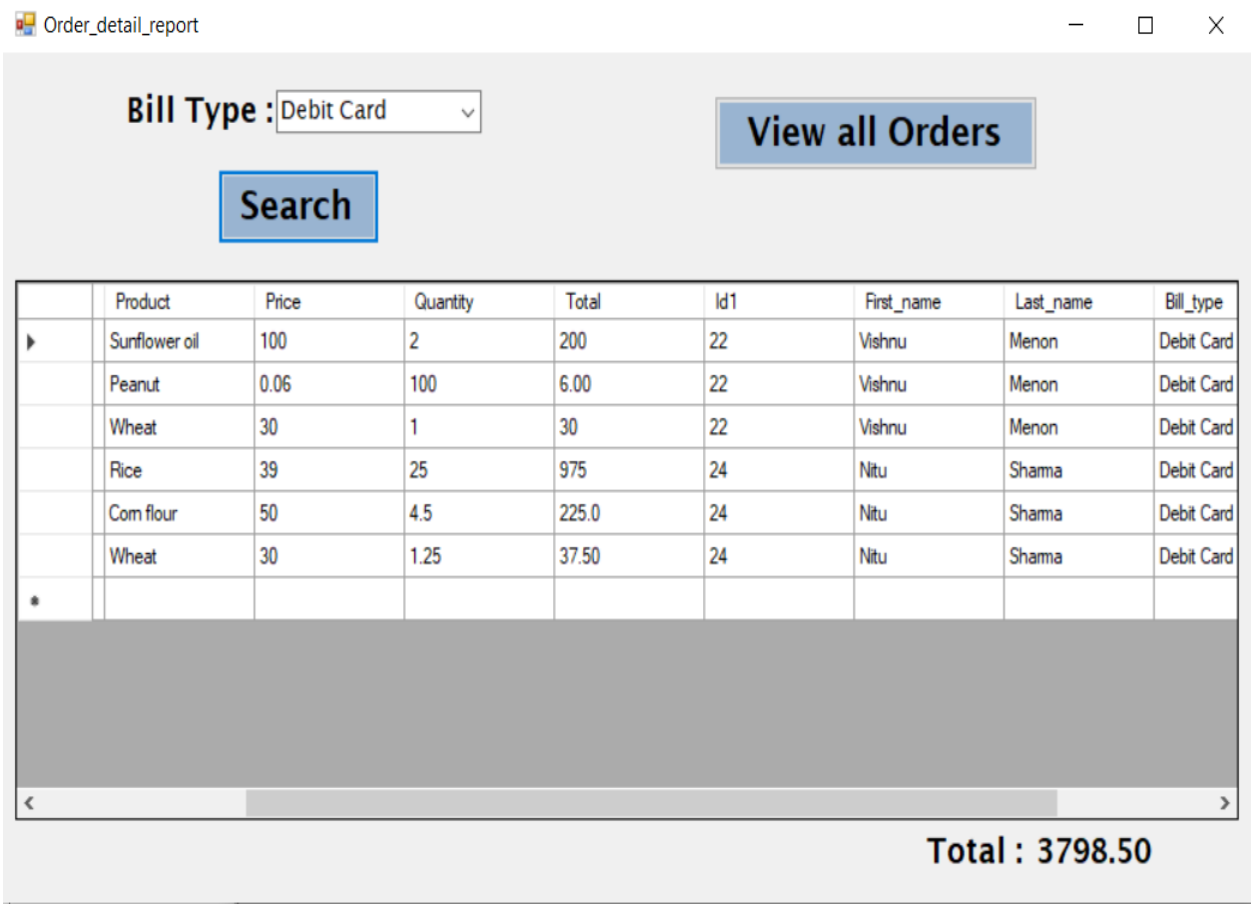
- It shows the report of all the stocks that is available in the “Stock” table in database

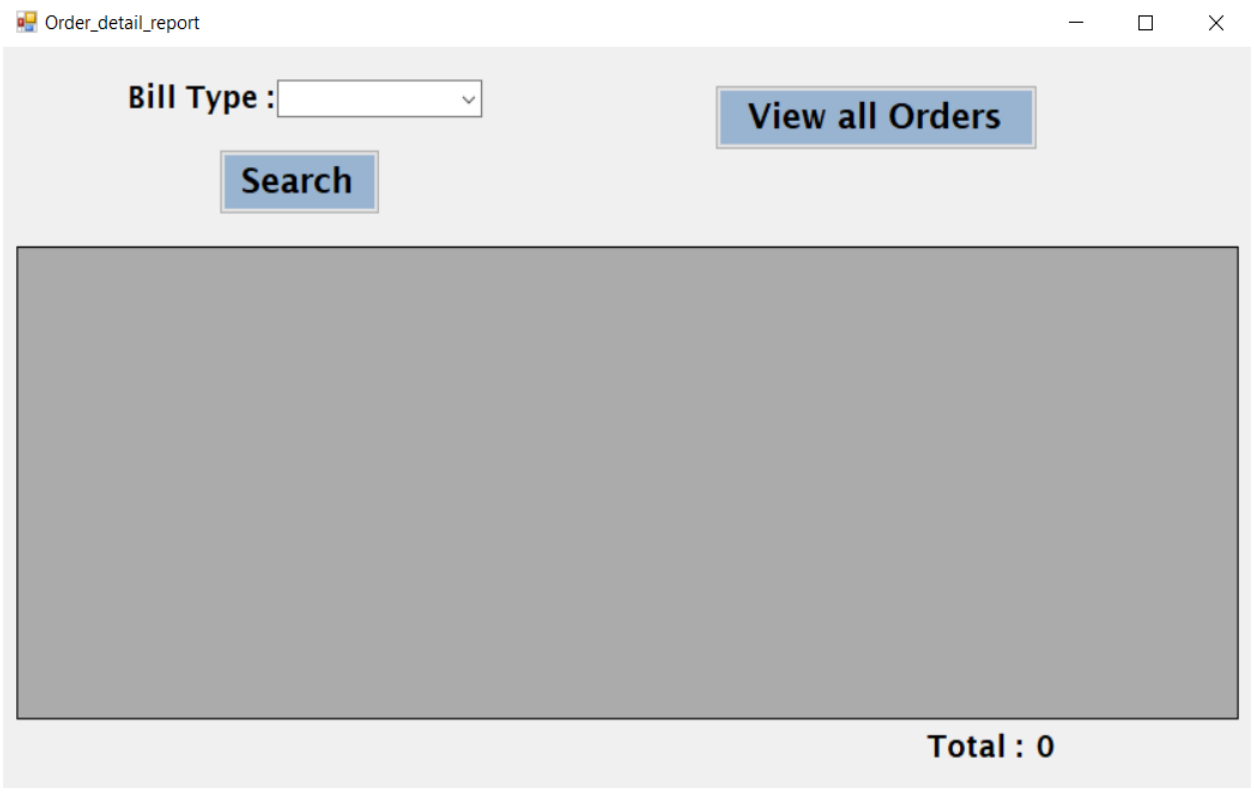


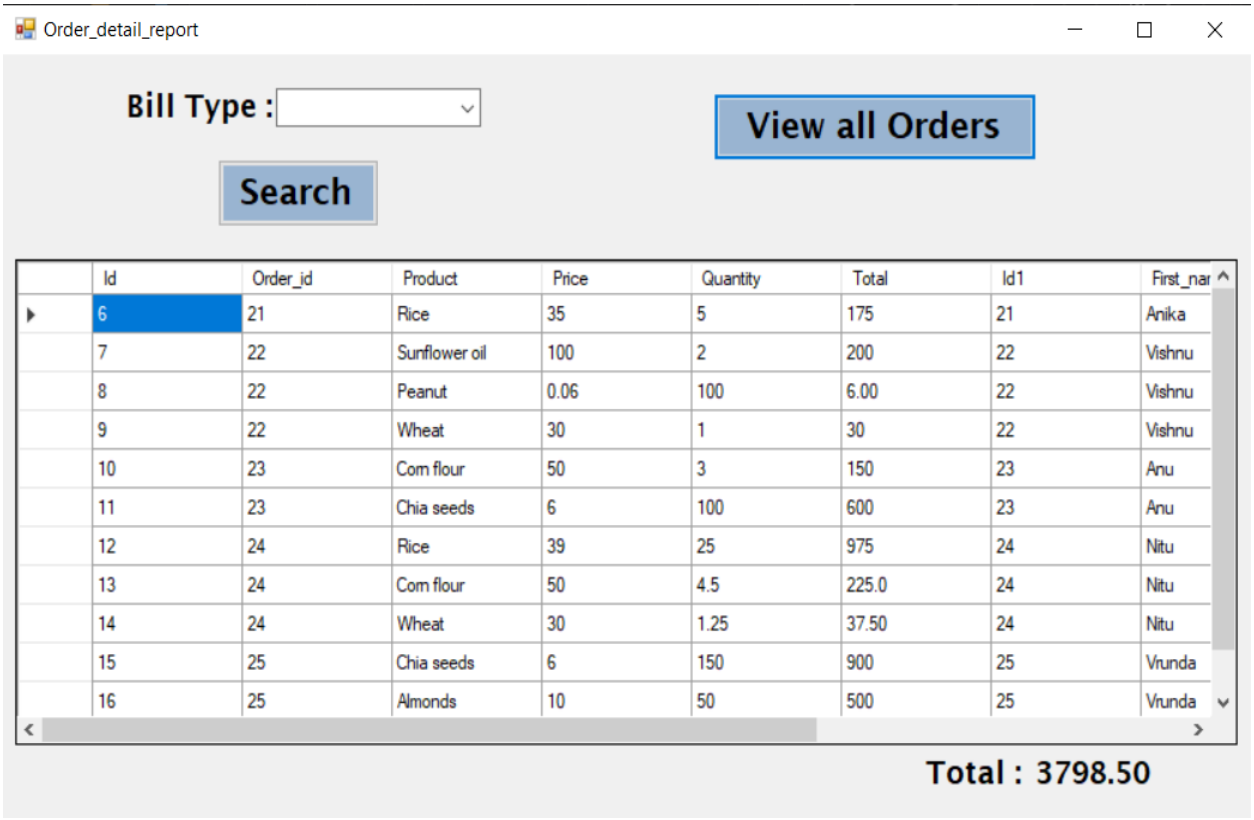
Functionality 9:

Report -> Order Detail Report

- It shows the report of all the orders that have been made from the “Order_Item” and “Order_user” table in database
- Select a bill type from the combobox (cash, UPI, Debit card, etc) and then click on Search button which will display all orders which used that payment method.
- View the entire order history by clicking on View all Orders button
- The total of all product prices that is displayed in the dataGridView can be viewed under the report







Id	Order_id	Product	Price	Quantity	Total	Id1	First_nar
6	21	Rice	35	5	175	21	Anika
7	22	Sunflower oil	100	2	200	22	Vishnu
8	22	Peanut	0.06	100	6.00	22	Vishnu
9	22	Wheat	30	1	30	22	Vishnu
10	23	Corn flour	50	3	150	23	Anu
11	23	Chia seeds	6	100	600	23	Anu
12	24	Rice	39	25	975	24	Nitu
13	24	Corn flour	50	4.5	225.0	24	Nitu
14	24	Wheat	30	1.25	37.50	24	Nitu
15	25	Chia seeds	6	150	900	25	Vrunda
16	25	Almonds	10	50	500	25	Vrunda

THANK YOU