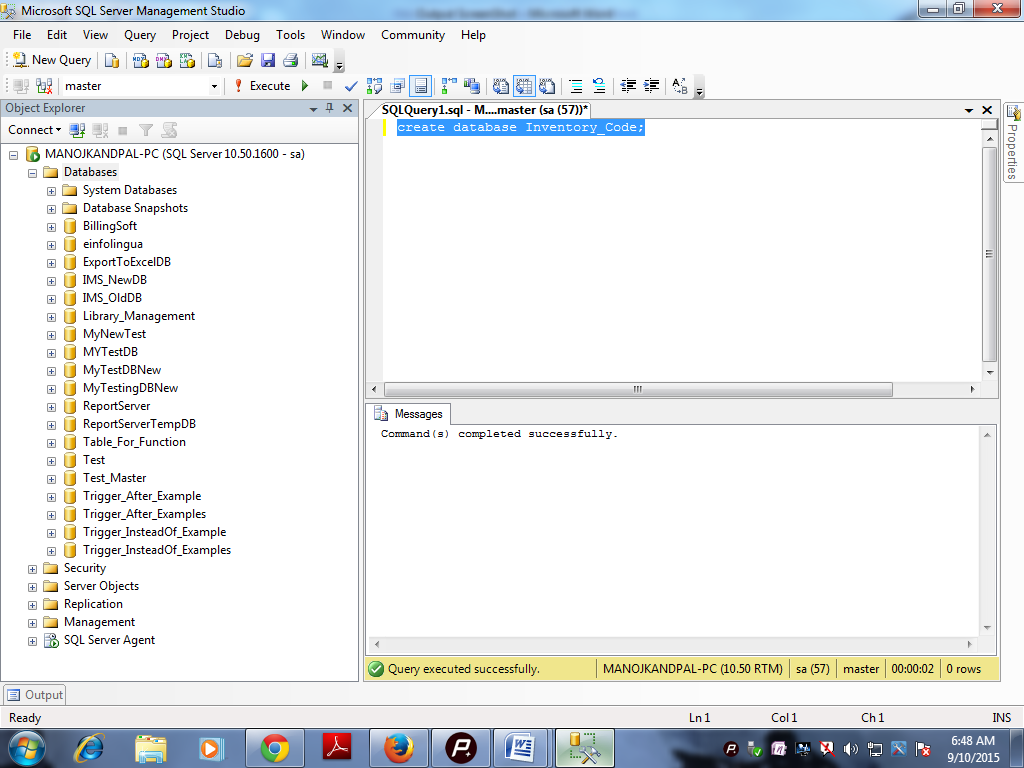
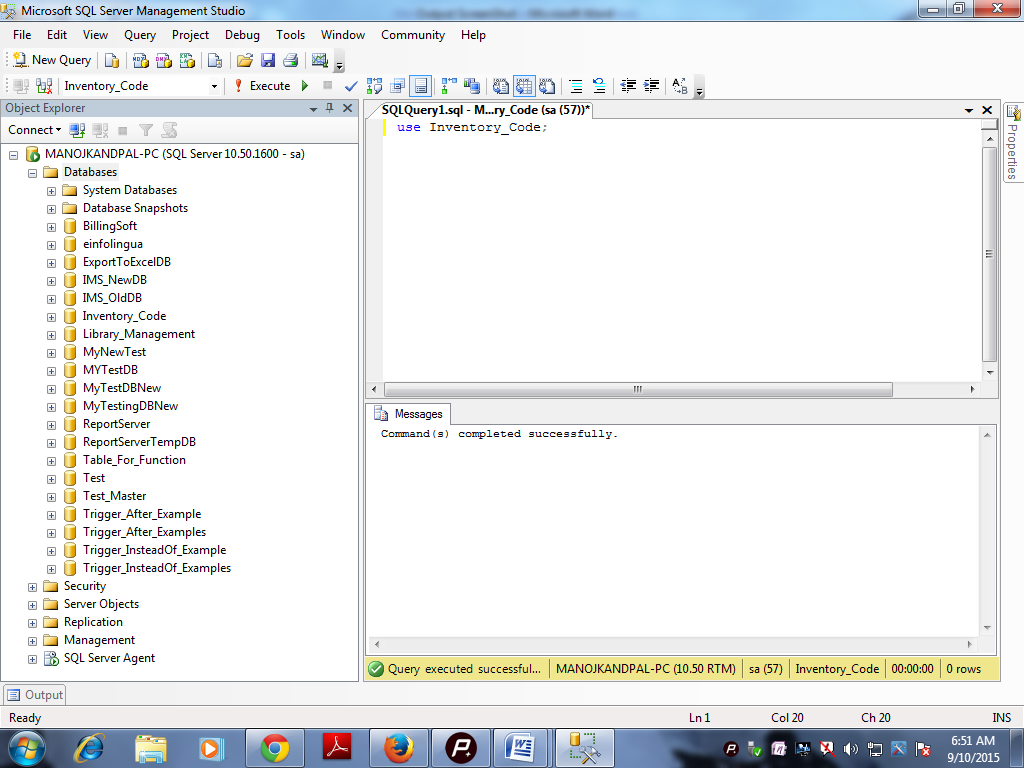
**First of all** we create a database in SQL server by command:

**Command:** create database Inventory\_Code;



Now we can use **Inventory\_Code** database by command:

**Command:** use Inventory\_Code;



Now create a table for **Admin\_Login** by Command:

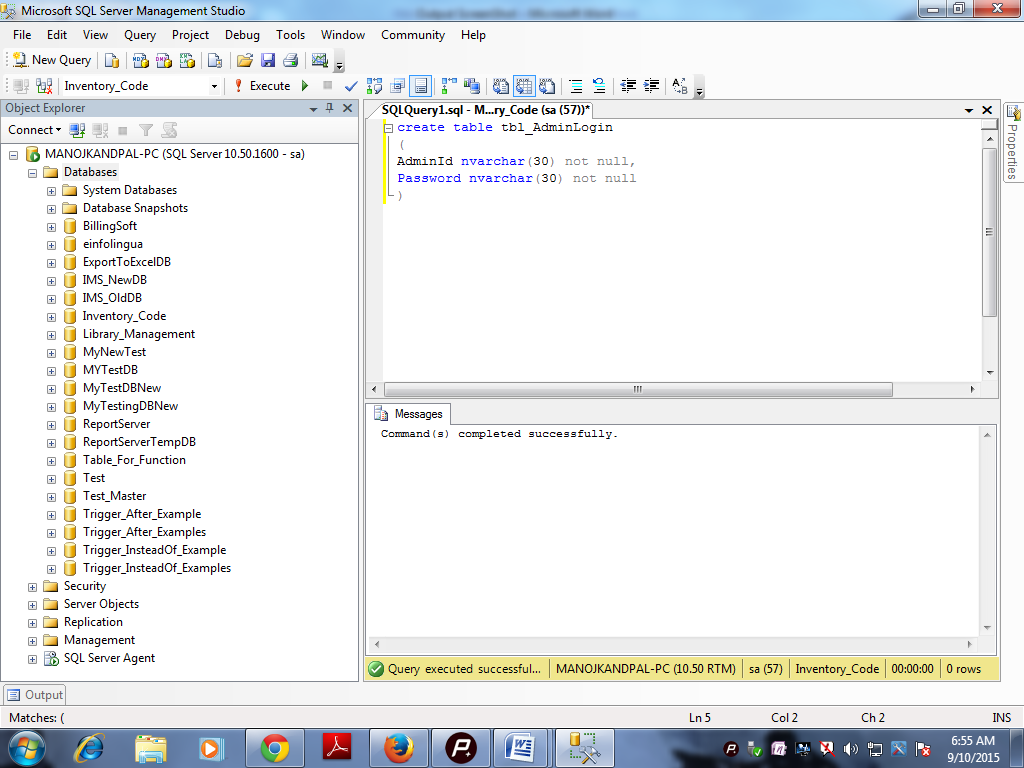
**Command:** create table tbl\_AdminLogin

(

AdminId nvarchar(30) not null,

Password nvarchar(30) not null

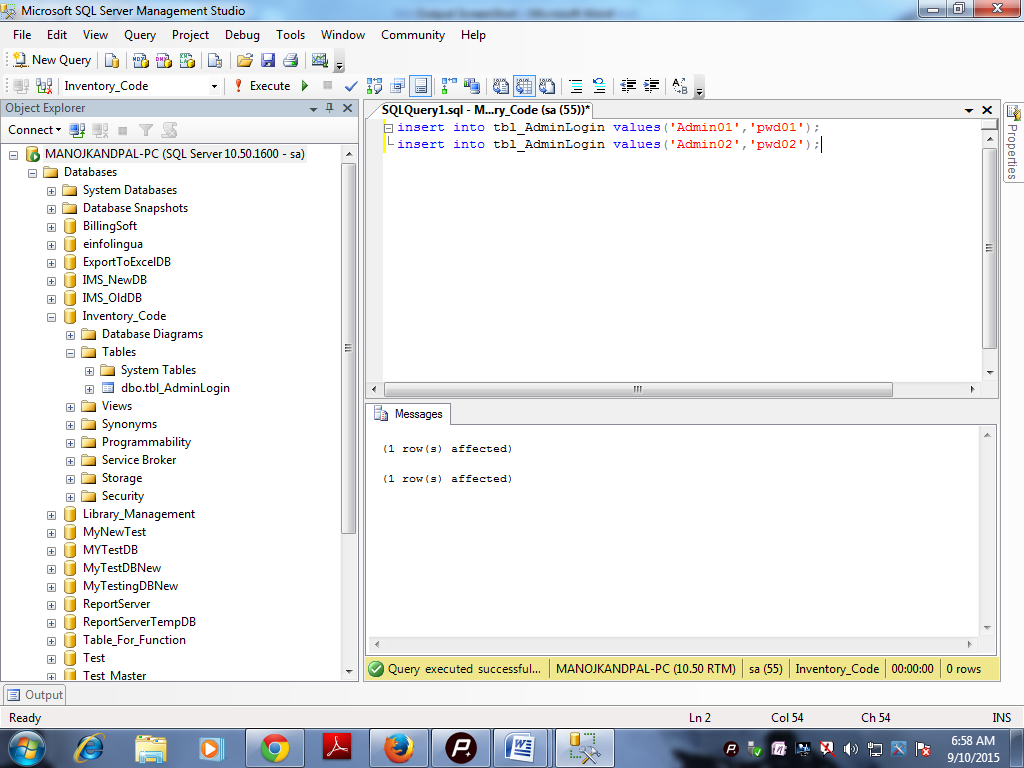
)



Now insert one or two records by command

**Command:** insert into tbl\_AdminLogin values('Admin01','pwd01');

insert into tbl\_AdminLogin values('Admin02','pwd02');



Now create a table for store product record by command:

**Command:** create table tbl\_ProductRecords

(

Product\_Id bigint not null,

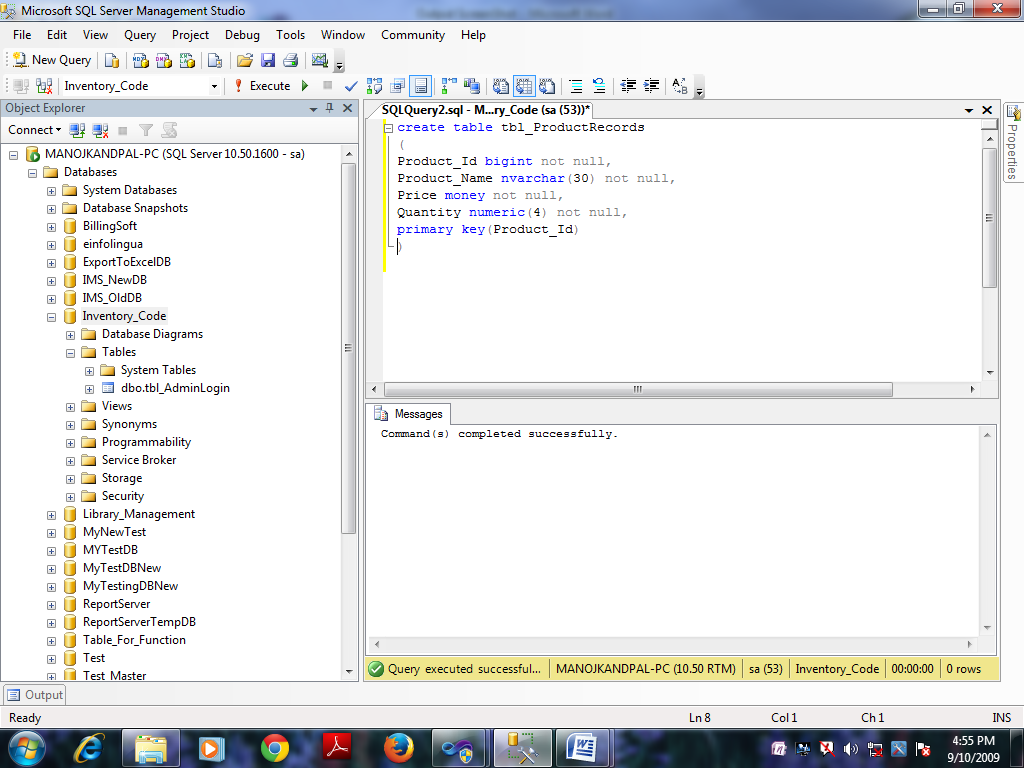
Product\_Name nvarchar(30) not null,

Price money not null,

Quantity numeric(4) not null,

primary key(Product\_Id)

)



**Note:** we can use store procedure for admin login and insert records in **tbl\_ProductRecords.**

**Command:** create procedure sp\_AdminLogin

@AdminId nvarchar(30),

@Password nvarchar(30)

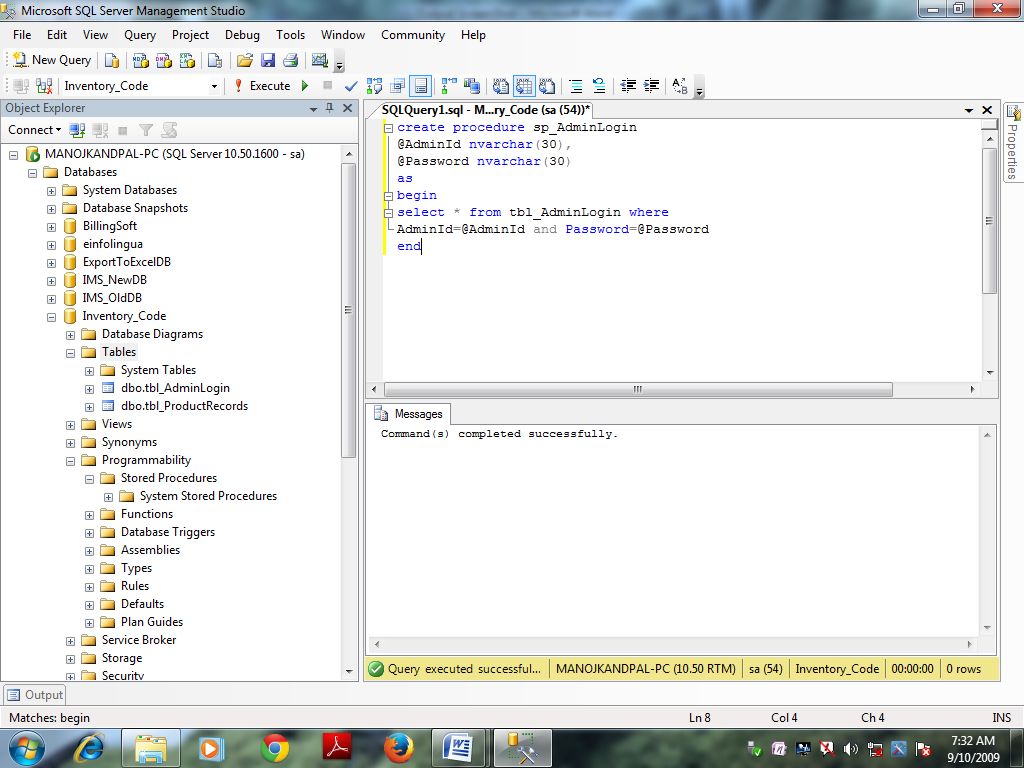
as

begin

select \* from tbl\_AdminLogin where

AdminId=@AdminId and Password=@Password

End

****

**Command:** create procedure sp\_InsertProduct

@Product\_Id bigint,

@Product\_Name nvarchar(30),

@Price money,

@Quantity numeric(4)

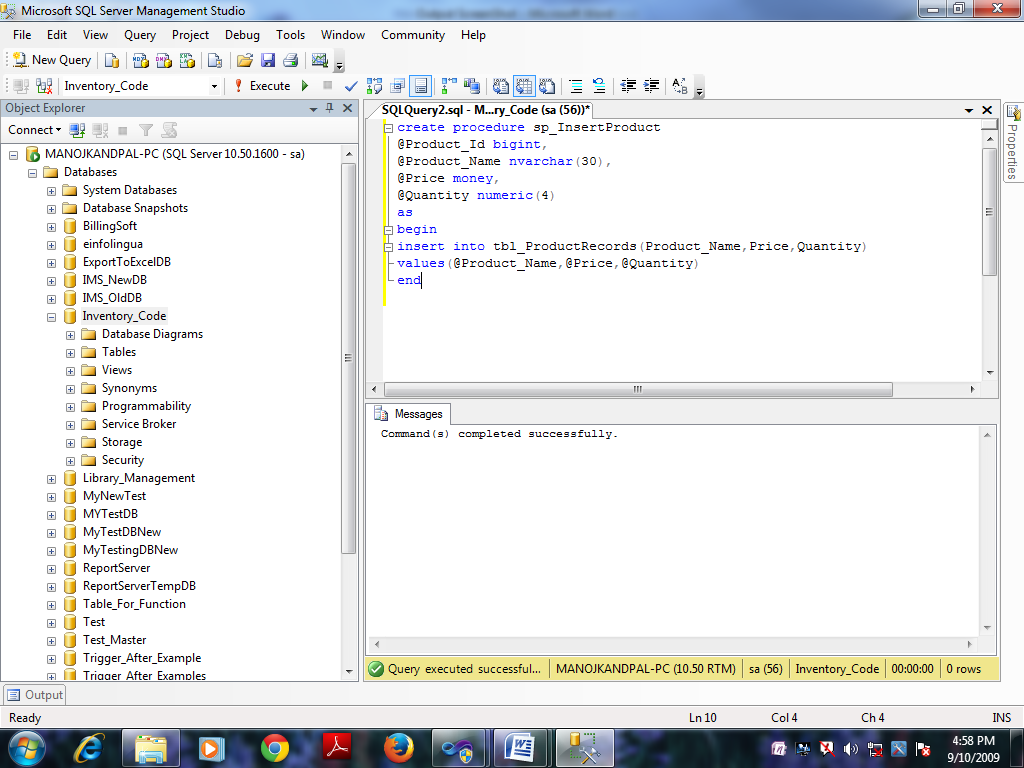
as

begin

insert into tbl\_ProductRecords(Product\_Name,Price,Quantity)

values(@Product\_Name,@Price,@Quantity)

end

****

**Now we can create another store procedure for show product records**

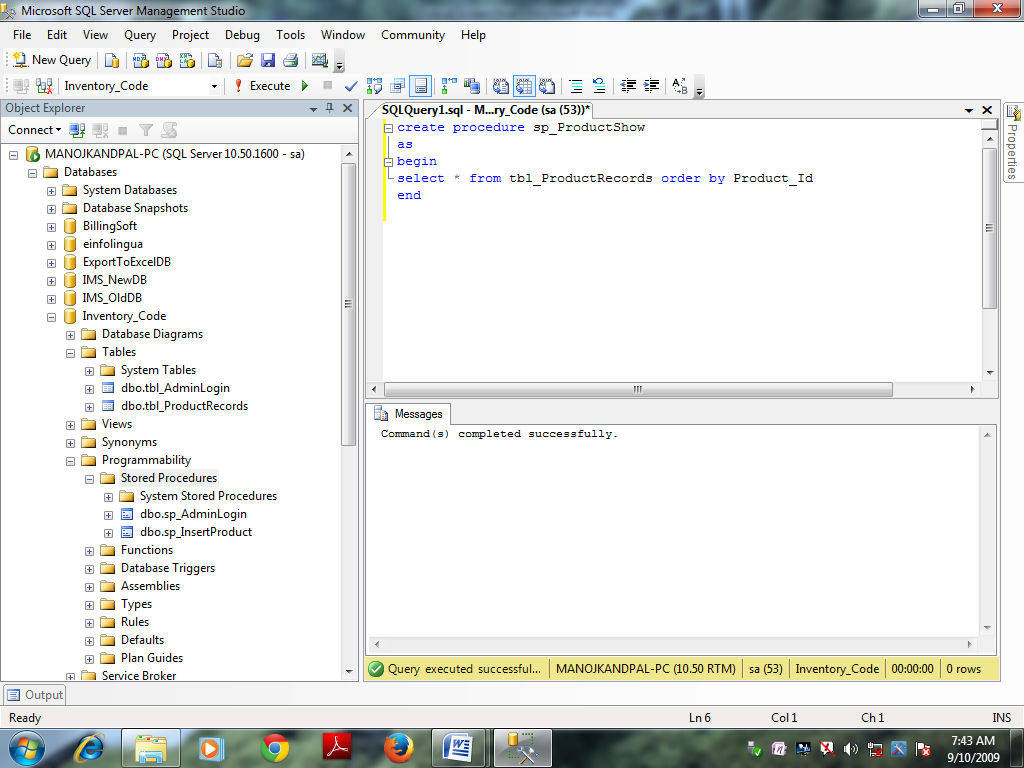
**Command:** create procedure sp\_ProductShow

as

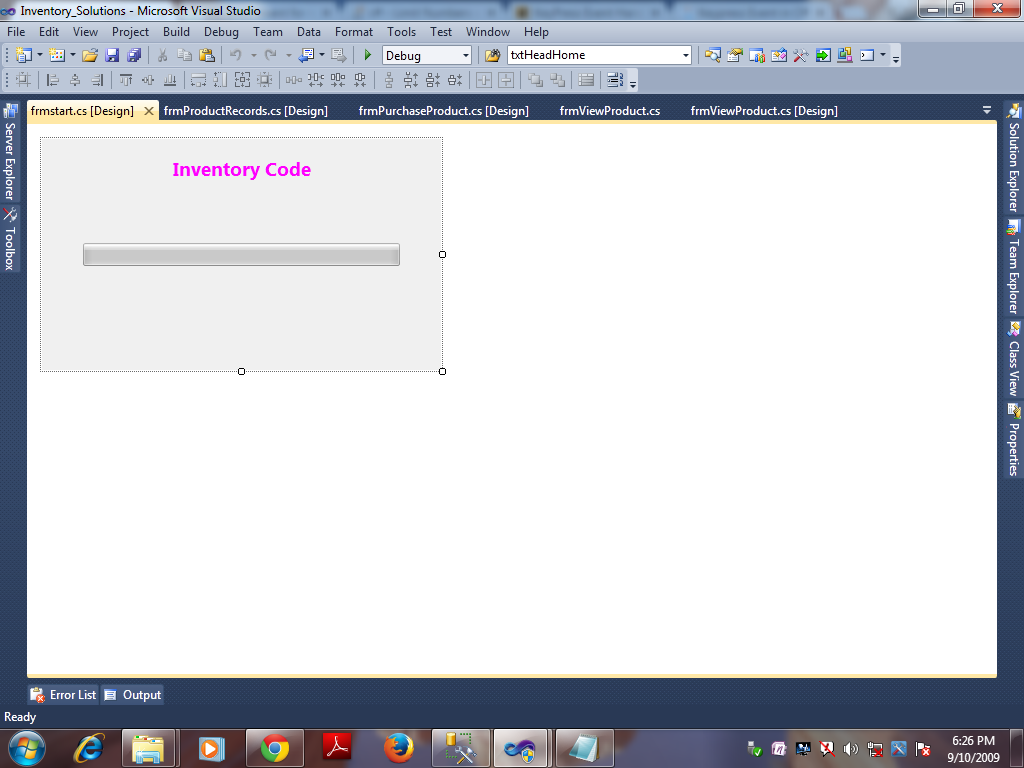
begin

select \* from tbl\_ProductRecords order by Product\_Id

end



**Note:** Our database task has been completed and Now we will work on **C# .NET**



And my code of this form in .cs file is

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

namespace Inventory\_Solutions

{

public partial class frmstart : Form

{

Timer timer1 = new Timer();

public frmstart()

{

InitializeComponent();

}

private void frmstart\_Load(object sender, EventArgs e)

{

timer1.Start();

timer1.Tick += new EventHandler(timer1\_Tick);

}

void timer1\_Tick(object sender, EventArgs e)

{

int i = 0;

for (i = 0; i <= 100; i++)

{

progressBar1.Value = i;

}

timer1.Enabled = false;

frmAdminLogin obj = new frmAdminLogin();

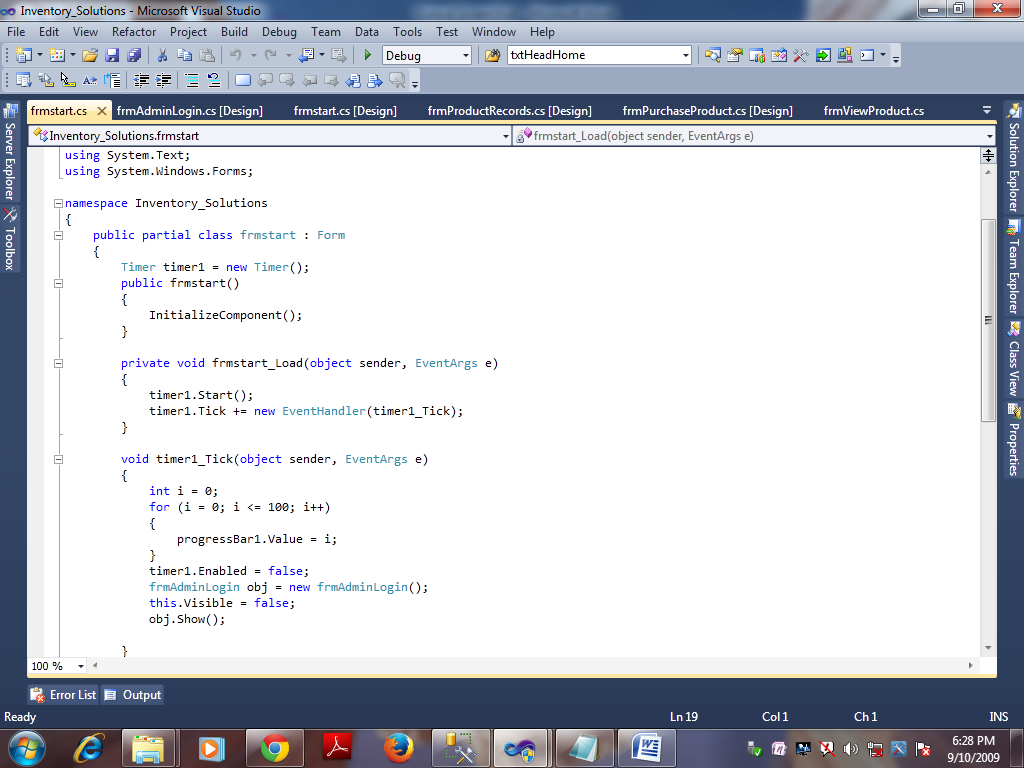
this.Visible = false;

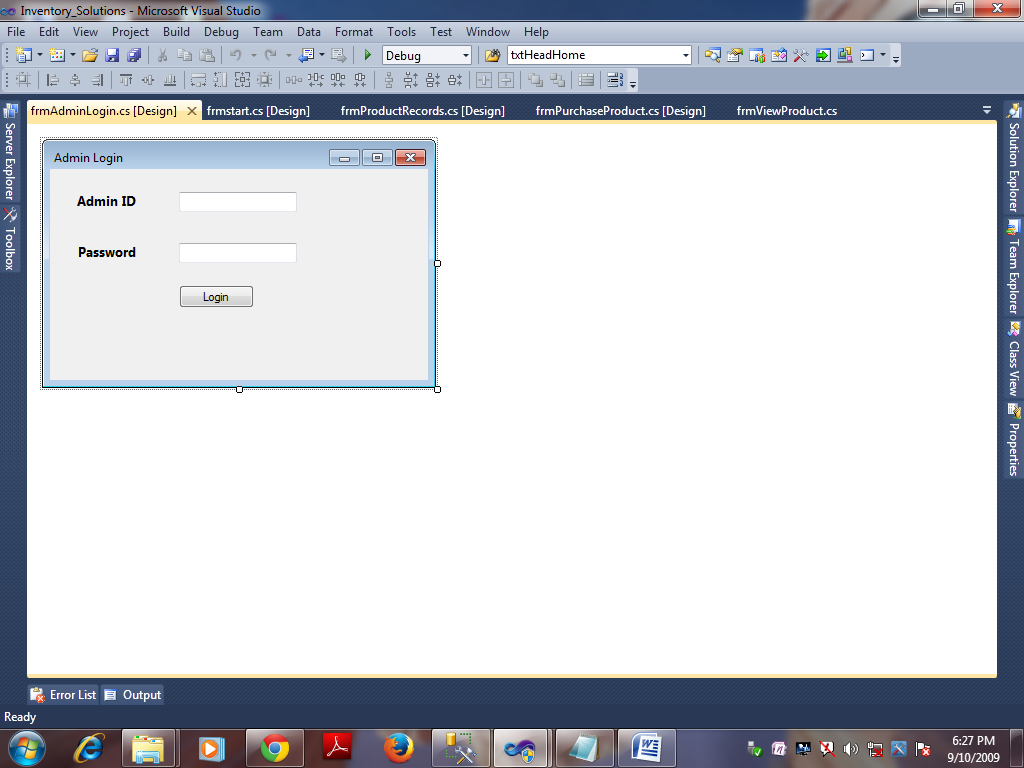
obj.Show();

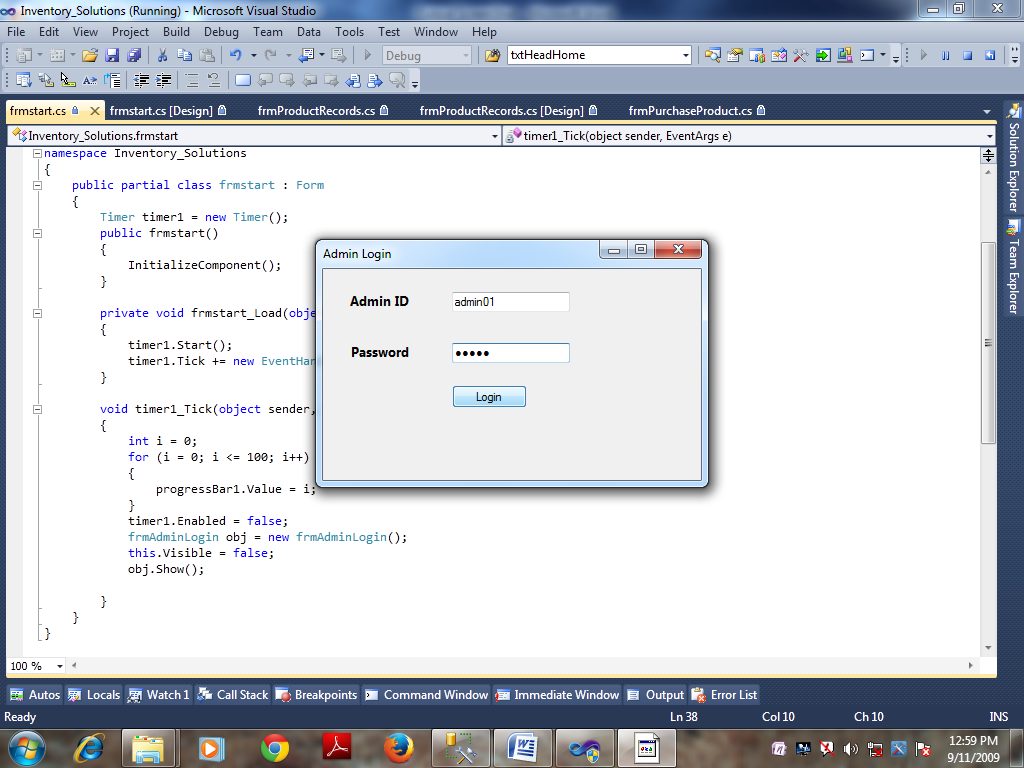
}

}

}







my code of this form in .cs file is

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace Inventory\_Solutions

{

public partial class frmAdminLogin : Form

{

DBConnect con = new DBConnect();

public static string AdminId;

public frmAdminLogin()

{

InitializeComponent();

}

private void btnlogin\_Click(object sender, EventArgs e)

{

string str = "select \* from tbl\_AdminLogin where AdminId='" + txtadminid.Text + "' and Password='" + txtpwd.Text + "'";

try

{

con.OpenConnection();

SqlCommand cmd = new SqlCommand(str, DBConnect.Connection);

SqlDataReader dr = cmd.ExecuteReader();

if (dr.HasRows)

{

while (dr.Read())

{

frmMDIParent obj = new frmMDIParent();

obj.Show();

this.Hide();

AdminId = txtadminid.Text;

obj.label1.Text = "Welcome " + AdminId;

}

}

else

{

lblerrorfull.Text = "Invalid Admin Id or Password \n Please Enter Correct Admin Id And Password";

lblerrorfull.ForeColor = Color.Red;

}

}

catch (SqlException ex)

{

MessageBox.Show(ex.Message);

}

finally

{

con.CloseConnection();

}

}

private void txtadminid\_Leave(object sender, EventArgs e)

{

if (txtadminid.Text.Trim().Length.Equals(0))

{

lblerroradminid.Visible = true;

lblerroradminid.Text = "Please Enter Admin Id First";

lblerroradminid.ForeColor = Color.Red;

txtadminid.Focus();

}

else

{

lblerroradminid.Visible = false;

}

}

private void txtpwd\_Leave(object sender, EventArgs e)

{

if (txtpwd.Text.Trim().Length.Equals(0))

{

lblerrorpwd.Visible = true;

lblerrorpwd.Text = "Please Enter Password First";

lblerrorpwd.ForeColor = Color.Red;

txtpwd.Focus();

}

else

{

lblerrorpwd.Visible = false;

}

}

private void frmAdminLogin\_FormClosing(object sender, FormClosingEventArgs e)

{

if (MessageBox.Show("Do You Want To Exit", "Exit", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

this.Dispose();

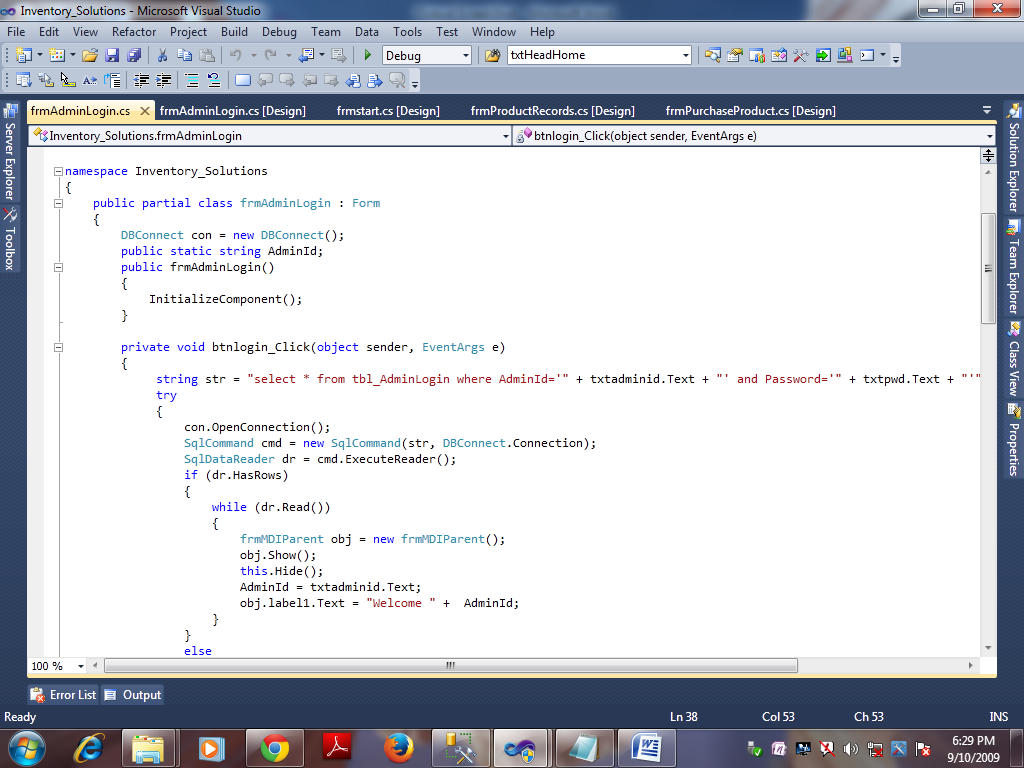
Application.Exit();

}

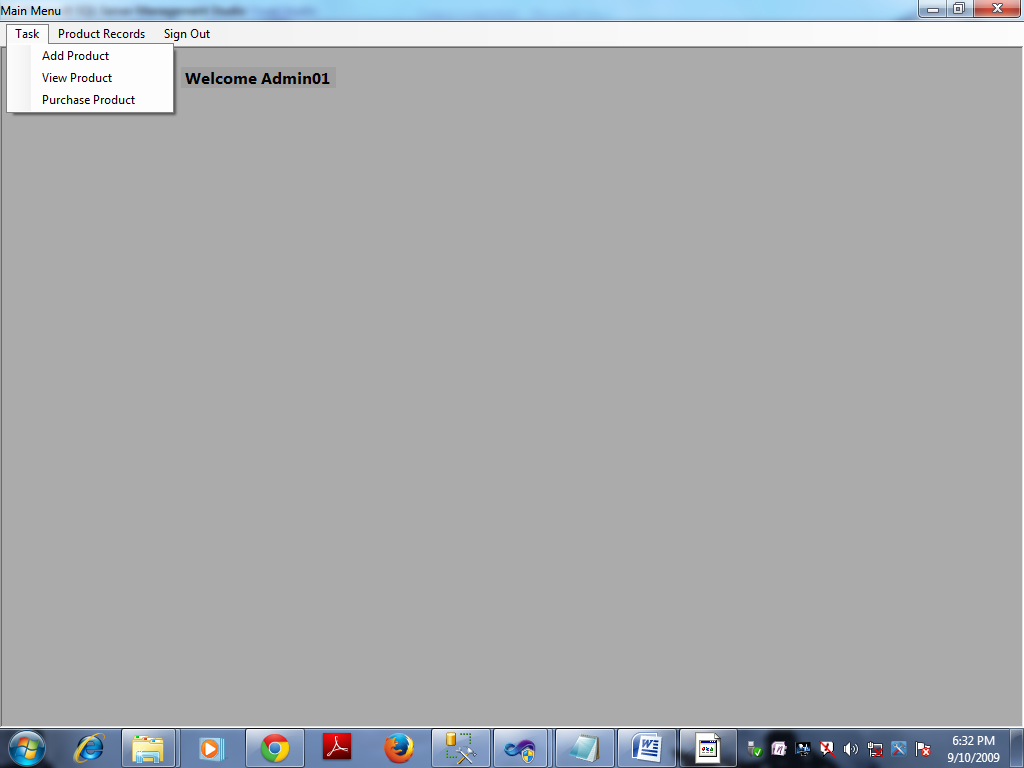
}

}

}



When ever we will login then our window will be as (Running output Window)



And my code of MDIparent form in .cs file is

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

namespace Inventory\_Solutions

{

public partial class frmMDIParent : Form

{

public frmMDIParent()

{

InitializeComponent();

}

private bool CheckForDuplicationForm(Form NewForm)

{

bool b = false;

foreach (Form frm in this.MdiChildren)

{

if (frm.GetType() == NewForm.GetType())

{

frm.Activate();

b = true;

}

}

return b;

}

private void frmMDIParent\_FormClosing(object sender, FormClosingEventArgs e)

{

if (MessageBox.Show("Do You Want to Exit?", "Exit", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

frmAdminLogin obj = new frmAdminLogin();

obj.Show();

this.Dispose();

}

else

{

e.Cancel = true;

}

}

private void frmMDIParent\_Load(object sender, EventArgs e)

{

frmAdminLogin obj = new frmAdminLogin();

obj.Dispose();

}

private void signOutToolStripMenuItem\_Click(object sender, EventArgs e)

{

if (MessageBox.Show("Do You Want To LogOut", "LogOut", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

frmAdminLogin obj = new frmAdminLogin();

obj.Show();

this.Dispose();

}

else

{

}

}

private void viewToolStripMenuItem\_Click(object sender, EventArgs e)

{

frmAddProduct obj = new frmAddProduct();

bool frmpresent = CheckForDuplicationForm(obj);

if (frmpresent)

return;

else if (!frmpresent)

{

obj.MdiParent = this;

obj.Show();

}

}

private void purchaseToolStripMenuItem\_Click(object sender, EventArgs e)

{

frmViewProduct obj = new frmViewProduct();

bool frmpresent = CheckForDuplicationForm(obj);

if (frmpresent)

return;

else if (!frmpresent)

{

obj.MdiParent = this;

obj.Show();

}

}

private void purchaseProductToolStripMenuItem\_Click(object sender, EventArgs e)

{

frmPurchaseProduct obj = new frmPurchaseProduct();

bool frmpresent = CheckForDuplicationForm(obj);

if (frmpresent)

return;

else if (!frmpresent)

{

obj.MdiParent = this;

obj.Show();

}

}

private void productRecordsToolStripMenuItem\_Click(object sender, EventArgs e)

{

frmProductRecords obj = new frmProductRecords();

bool frmpresent = CheckForDuplicationForm(obj);

if (frmpresent)

return;

else if (!frmpresent)

{

obj.MdiParent = this;

obj.Show();

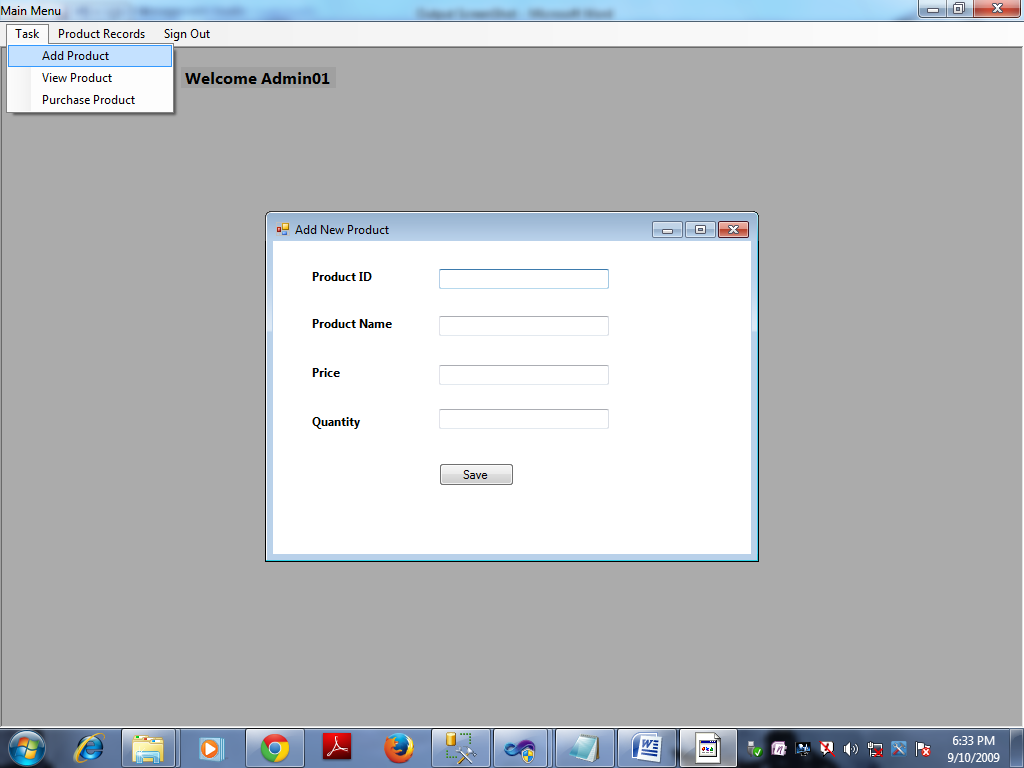
}

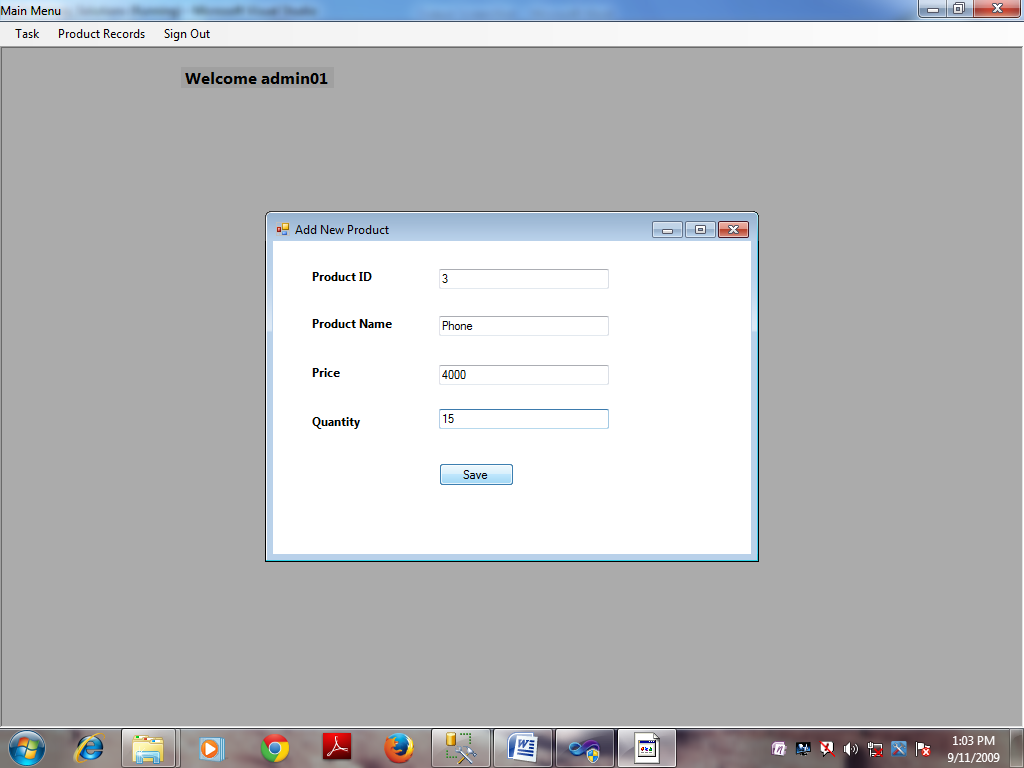
}

}

}

And Now Adding Product here





My code in .cs file for this form is

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Data.SqlClient;

using System.Linq;

using System.Text;

using System.Windows.Forms;

namespace Inventory\_Solutions

{

public partial class frmAddProduct : Form

{

DBConnect con = new DBConnect();

public frmAddProduct()

{

InitializeComponent();

}

private void txtproductid\_Leave(object sender, EventArgs e)

{

if (txtproductid.Text.Trim().Length.Equals(0))

{

lblerrorproduct\_id.Visible = true;

lblerrorproduct\_id.Text = "Please Enter Product ID";

lblerrorproduct\_id.ForeColor = Color.Red;

txtproductid.Focus();

}

else

{

lblerrorproduct\_id.Visible = false;

}

}

private void txtproductname\_Leave(object sender, EventArgs e)

{

if (txtproductname.Text.Trim().Length.Equals(0))

{

lblerrorproductname.Visible = true;

lblerrorproductname.Text = "Please Enter Product Name";

lblerrorproductname.ForeColor = Color.Red;

txtproductname.Focus();

}

else

{

lblerrorproductname.Visible = false;

}

}

private void txtprice\_Leave(object sender, EventArgs e)

{

if (txtprice.Text.Trim().Length.Equals(0))

{

lblerrorprice.Visible = true;

lblerrorprice.Text = "Please Enter Price";

lblerrorprice.ForeColor = Color.Red;

txtprice.Focus();

}

else

{

lblerrorprice.Visible = false;

}

}

private void txtquantity\_Leave(object sender, EventArgs e)

{

if (txtquantity.Text.Trim().Length.Equals(0))

{

lblerrorquantity.Visible = true;

lblerrorquantity.Text = "Please Enter Quantity First";

lblerrorquantity.ForeColor = Color.Red;

txtquantity.Focus();

}

else

{

lblerrorquantity.Visible = false;

}

}

private void frmAddProduct\_FormClosing(object sender, FormClosingEventArgs e)

{

if (MessageBox.Show("Do You Want To Exit", "Exit", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

this.Dispose();

}

else

{

e.Cancel = true;

}

}

private void btnsave\_Click(object sender, EventArgs e)

{

string str = "insert into tbl\_ProductRecords (Product\_Id,Product\_Name,Price,Quantity) values ('" + txtproductid.Text + "','" + txtproductname.Text + "','" + txtprice.Text + "','" + txtquantity.Text + "')";

try

{

con.OpenConnection();

SqlCommand cmd = new SqlCommand(str, DBConnect.Connection);

int res = cmd.ExecuteNonQuery();

if (res == 1)

{

MessageBox.Show("Data Successfully Added");

txtproductid.Clear();

txtproductname.Clear();

txtprice.Clear();

txtquantity.Clear();

}

}

catch (SqlException ex)

{

MessageBox.Show(ex.Message);

}

finally

{

con.CloseConnection();

}

}

private void txtprice\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar) && e.KeyChar != '.')

{

e.Handled = true;

}

if (e.KeyChar == '.' && (sender as TextBox).Text.IndexOf('.') > -1)

{

e.Handled = true;

}

}

private void txtproductid\_KeyPress(object sender, KeyPressEventArgs e)

{

if ((((int)e.KeyChar < 48) || ((int)e.KeyChar > 57)) && ((int)e.KeyChar != 8))

{

e.Handled = true;

MessageBox.Show("Please Enter Numbers Only");

}

else

{

e.Handled = false;

}

}

private void txtquantity\_KeyPress(object sender, KeyPressEventArgs e)

{

if ((((int)e.KeyChar < 48) || ((int)e.KeyChar > 57)) && ((int)e.KeyChar != 8))

{

e.Handled = true;

MessageBox.Show("Please Enter Numbers Only");

}

else

{

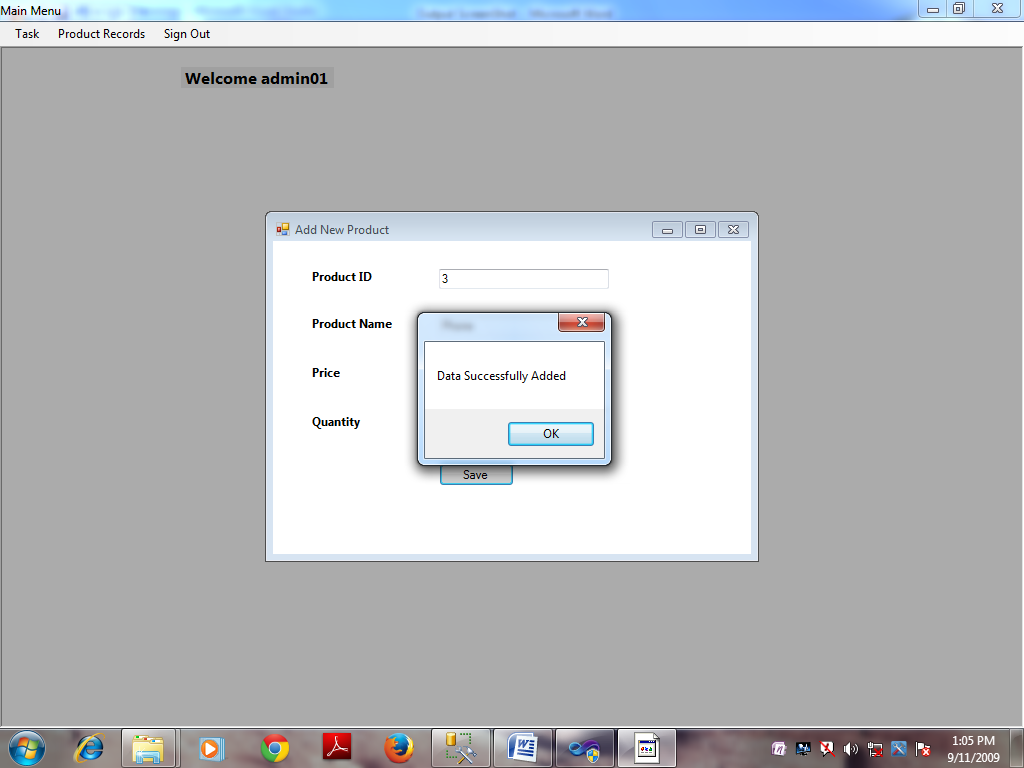
e.Handled = false;

}

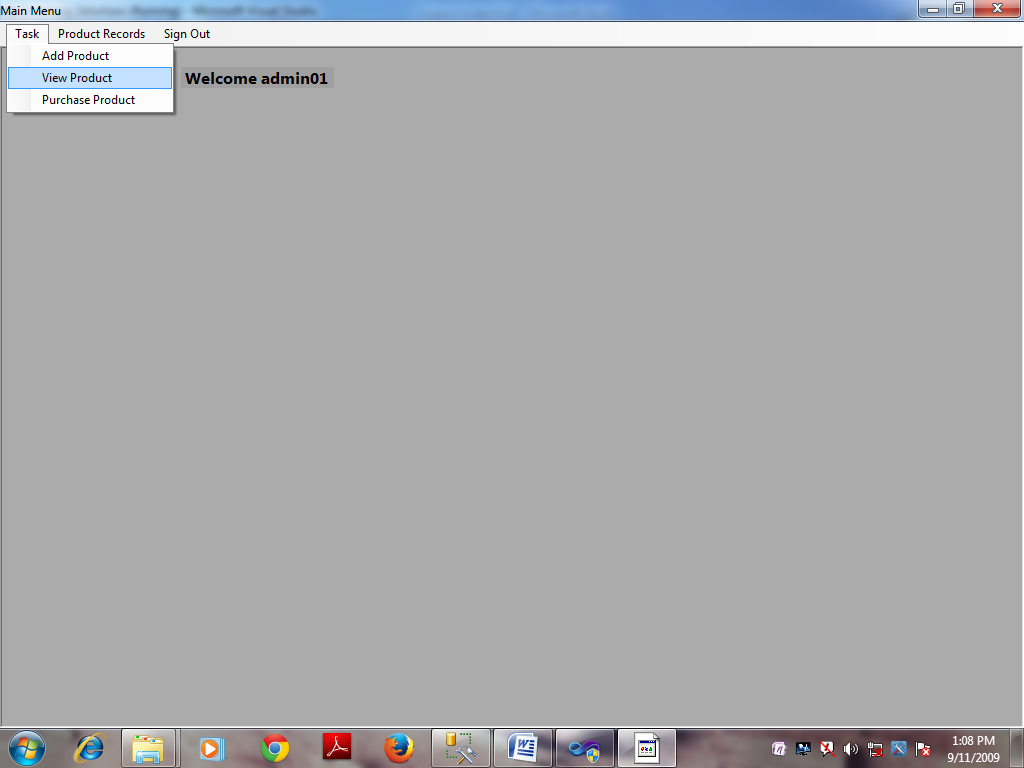
}

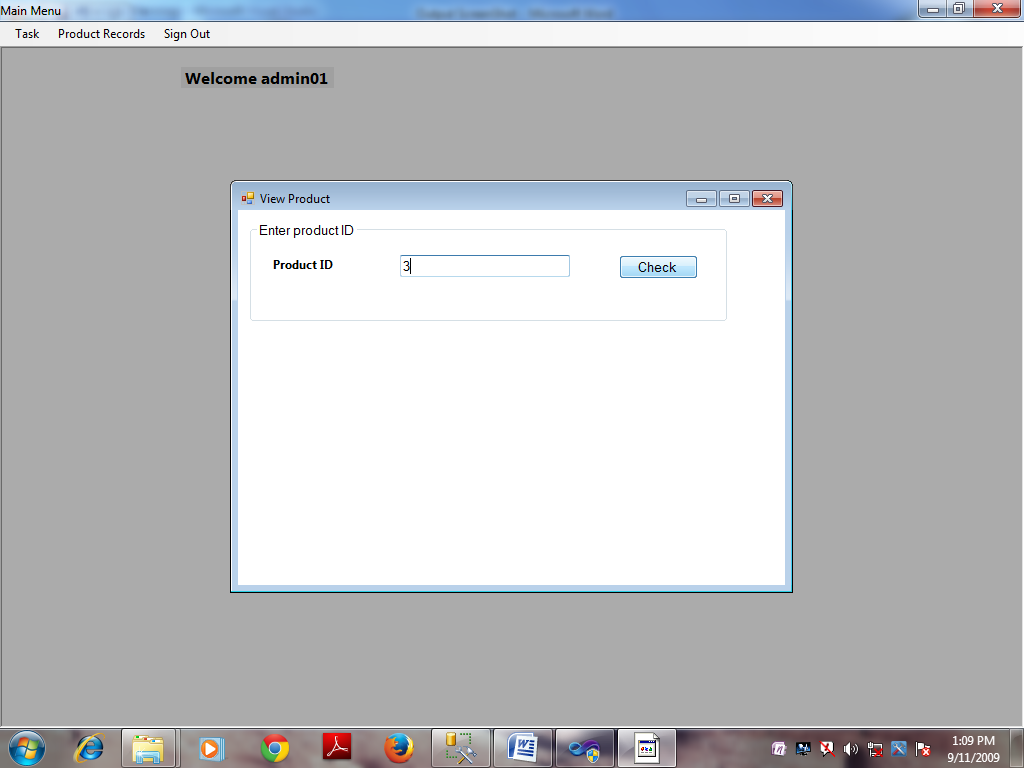
}

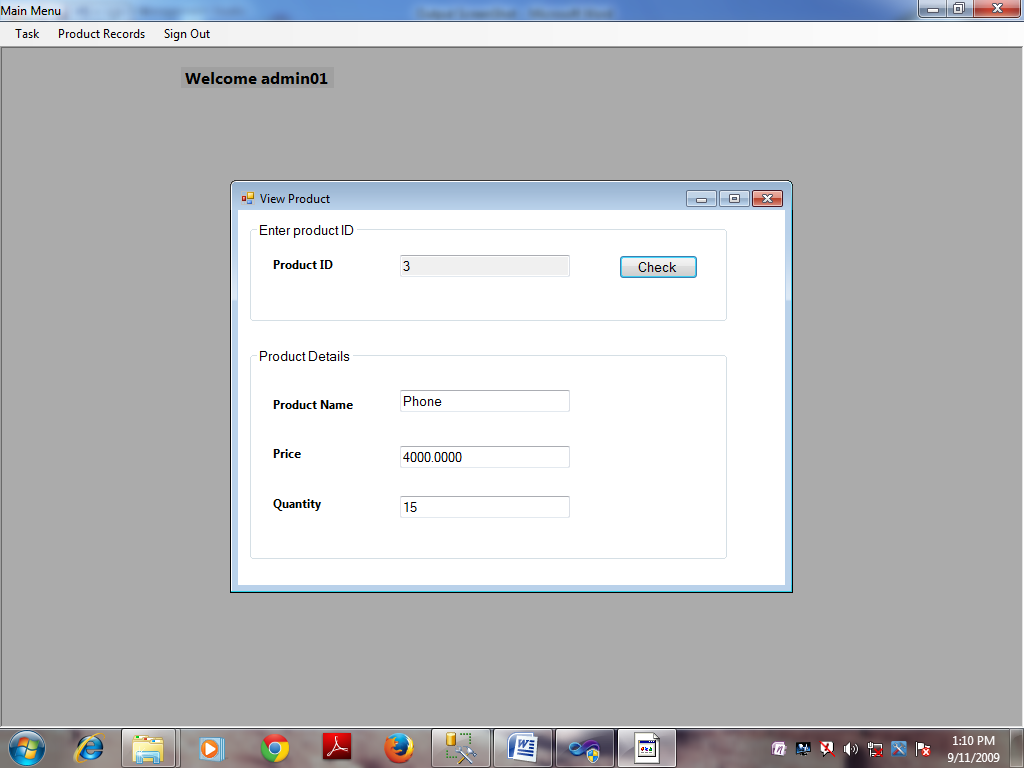
}



Viewing Product Here







My code in .cs file for this form is

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Data.SqlClient;

using System.Linq;

using System.Text;

using System.Windows.Forms;

namespace Inventory\_Solutions

{

public partial class frmViewProduct : Form

{

DBConnect con = new DBConnect();

public frmViewProduct()

{

InitializeComponent();

}

private void txtproductid\_Leave(object sender, EventArgs e)

{

if (txtproductid.Text.Trim().Length.Equals(0))

{

lblerrorproductid.Visible = true;

lblerrorproductid.Text = "Please Enter Product ID";

lblerrorproductid.ForeColor = Color.Red;

txtproductid.Focus();

}

else

{

lblerrorproductid.Visible = false;

}

}

private void btncheck\_Click(object sender, EventArgs e)

{

string str = "select \* from tbl\_ProductRecords where Product\_Id=" + txtproductid.Text + "";

try

{

con.OpenConnection();

SqlCommand cmd = new SqlCommand(str, DBConnect.Connection);

SqlDataReader dr = cmd.ExecuteReader();

if (dr.HasRows)

{

while (dr.Read())

{

groupBox2.Visible = true;

txtproductname.Text = dr.GetValue(1).ToString();

txtprice.Text = dr.GetValue(2).ToString();

txtquantity.Text = dr.GetValue(3).ToString();

txtproductid.ReadOnly = true;

}

}

else

{

lblerrorproductid.Visible = true;

lblerrorproductid.Text = "Record Not Found,\nPlease Enter Corrct Product ID";

lblerrorproductid.ForeColor = Color.Red;

txtproductid.Focus();

}

}

catch (SqlException ex)

{

MessageBox.Show(ex.Message);

}

finally

{

con.CloseConnection();

}

}

private void txtproductid\_KeyPress(object sender, KeyPressEventArgs e)

{

if ((((int)e.KeyChar < 48) || ((int)e.KeyChar > 57)) && ((int)e.KeyChar != 8))

{

e.Handled = true;

MessageBox.Show("Please Enter Numbers Only");

}

else

{

e.Handled = false;

}

}

private void frmViewProduct\_FormClosing(object sender, FormClosingEventArgs e)

{

if (MessageBox.Show("Do You Want To Exit", "Exit", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

this.Dispose();

}

else

{

e.Cancel = true;

}

}

private void frmViewProduct\_Load(object sender, EventArgs e)

{

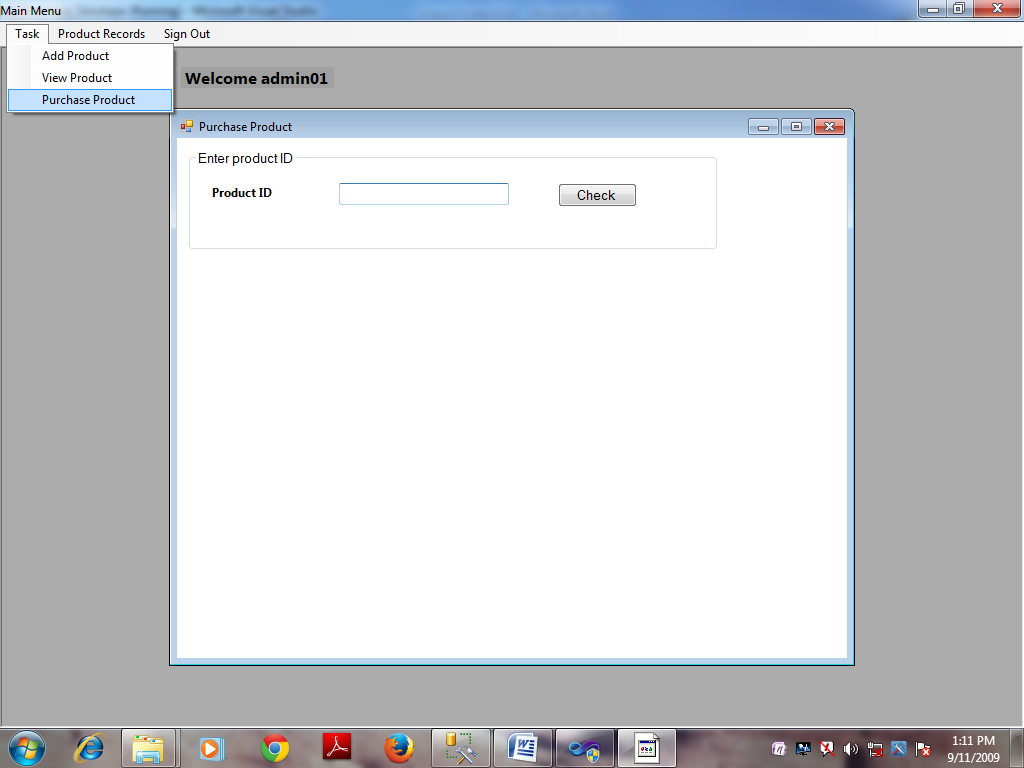
groupBox2.Visible = false;

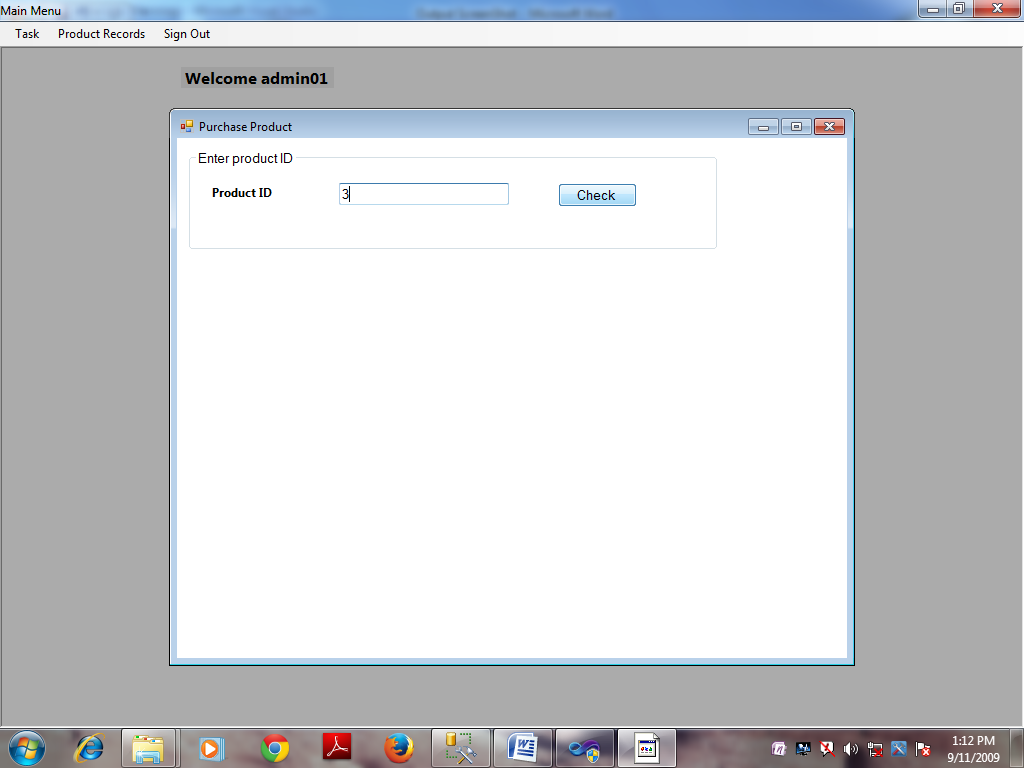
}

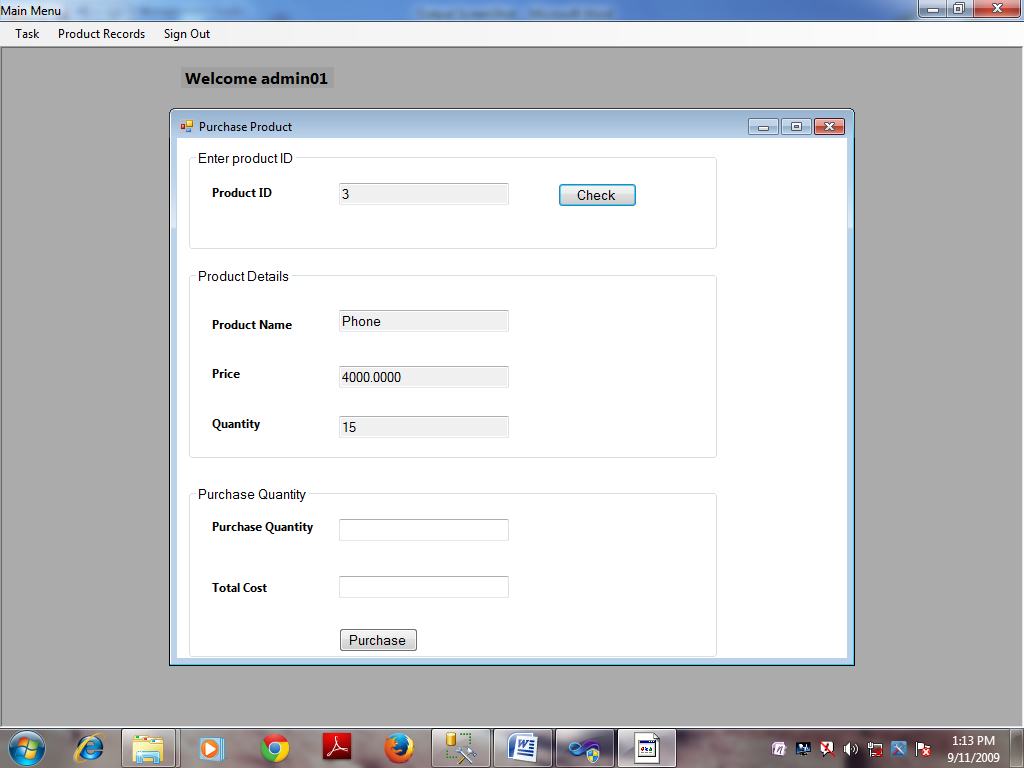
}

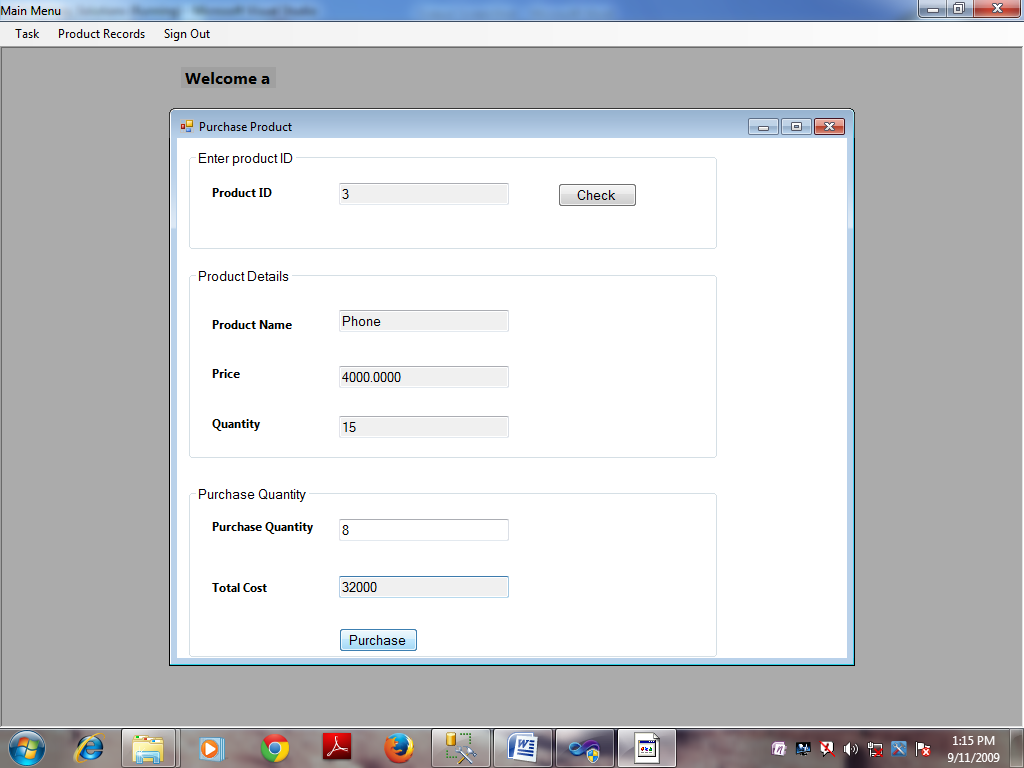
}

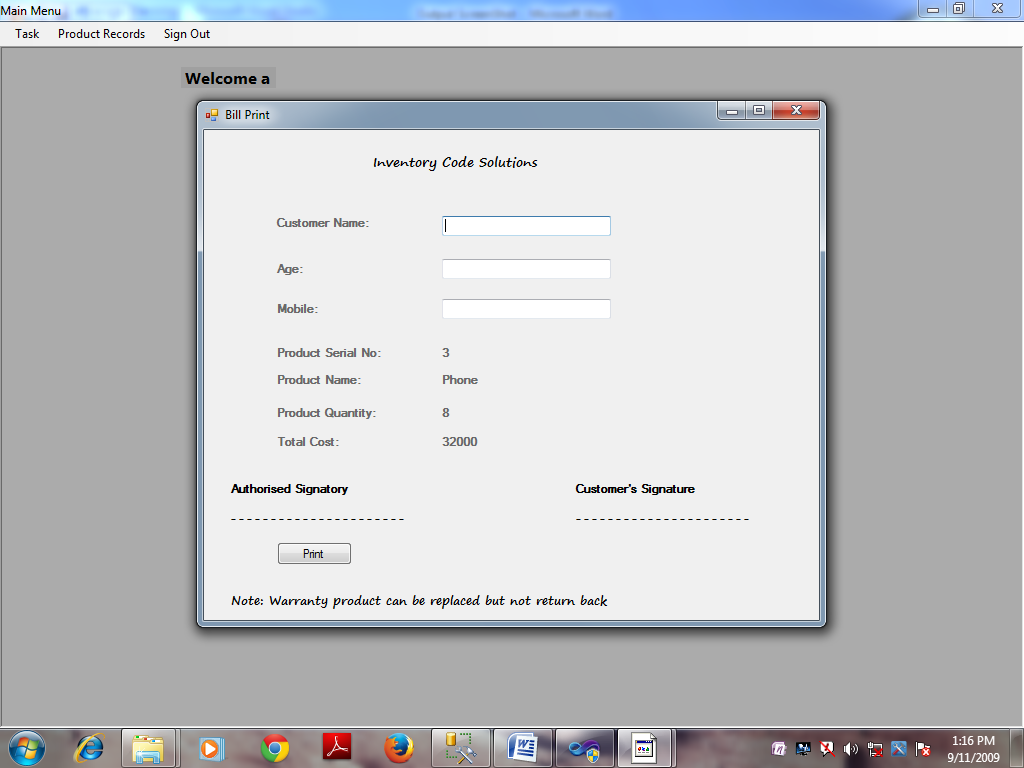
And now purchase product here











My code in .cs file for this form is

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace Inventory\_Solutions

{

public partial class frmPurchaseProduct : Form

{

DBConnect con = new DBConnect();

public frmPurchaseProduct()

{

InitializeComponent();

}

private void txtproductid\_Leave(object sender, EventArgs e)

{

if (txtproductid.Text.Trim().Length.Equals(0))

{

lblerrorproductid.Visible = true;

lblerrorproductid.Text = "Please Enter Product ID";

lblerrorproductid.ForeColor = Color.Red;

txtproductid.Focus();

}

else

{

lblerrorproductid.Visible = false;

}

}

private void btncheck\_Click(object sender, EventArgs e)

{

string str = "select \* from tbl\_ProductRecords where Product\_Id=" + txtproductid.Text + "";

try

{

con.OpenConnection();

SqlCommand cmd = new SqlCommand(str, DBConnect.Connection);

SqlDataReader dr = cmd.ExecuteReader();

if (dr.HasRows)

{

while (dr.Read())

{

groupBox2.Visible = true;

groupBox3.Visible = true;

txtproductname.Text = dr.GetValue(1).ToString();

txtprice.Text = dr.GetValue(2).ToString();

txtquantity.Text = dr.GetValue(3).ToString();

txtproductid.ReadOnly = true;

txtproductname.ReadOnly = true;

txtprice.ReadOnly = true;

txtquantity.ReadOnly = true;

}

}

else

{

lblerrorproductid.Visible = true;

lblerrorproductid.Text = "Record Not Found,\n Please Enter Corrct Product ID";

lblerrorproductid.ForeColor = Color.Red;

txtproductid.Focus();

}

}

catch (SqlException ex)

{

MessageBox.Show(ex.Message);

}

finally

{

con.CloseConnection();

}

}

private void txtproductid\_KeyPress(object sender, KeyPressEventArgs e)

{

if ((((int)e.KeyChar < 48) || ((int)e.KeyChar > 57)) && ((int)e.KeyChar != 8))

{

e.Handled = true;

MessageBox.Show("Please Enter Numbers Only");

}

else

{

e.Handled = false;

}

}

private void txtpurchasequantity\_KeyPress(object sender, KeyPressEventArgs e)

{

if ((((int)e.KeyChar < 48) || ((int)e.KeyChar > 57)) && ((int)e.KeyChar != 8))

{

e.Handled = true;

MessageBox.Show("Please Enter Numbers Only");

}

else

{

e.Handled = false;

}

}

private void txtpurchasequantity\_Leave(object sender, EventArgs e)

{

if (txtpurchasequantity.Text.Trim().Length.Equals(0))

{

lblerrorpurchasequantity.Visible = true;

lblerrorpurchasequantity.Text = "Please Enter Quantity You Want to Purchase";

lblerrorpurchasequantity.ForeColor = Color.Red;

txtpurchasequantity.Focus();

}

else if (Convert.ToInt32(txtpurchasequantity.Text) > (Convert.ToInt32(txtquantity.Text)))

{

lblerrorpurchasequantity.Visible = true;

lblerrorpurchasequantity.Text = "Purchase Quantity Must Be Less Than of Product Quantity";

lblerrorpurchasequantity.ForeColor = Color.Red;

txtpurchasequantity.Focus();

}

else

{

lblerrorpurchasequantity.Visible = false;

txttotalcost.Text = (Int32.Parse(txtpurchasequantity.Text) \* double.Parse(txtprice.Text)).ToString();

txttotalcost.ReadOnly = true;

//txtpurchasequantity.ReadOnly = true;

}

}

private void frmPurchaseProduct\_Load(object sender, EventArgs e)

{

groupBox2.Visible = false;

groupBox3.Visible = false;

}

private void button1\_Click(object sender, EventArgs e)

{

string str = "update tbl\_ProductRecords set Quantity=" + (Int32.Parse(txtquantity.Text) - Int32.Parse(txtpurchasequantity.Text)) + " where Product\_Id=" + txtproductid.Text + "";

try

{

con.OpenConnection();

SqlCommand cmd = new SqlCommand(str, DBConnect.Connection);

int res = cmd.ExecuteNonQuery();

if (res == 1)

{

frmBillPrint obj = new frmBillPrint();

obj.lblproductid.Text = txtproductid.Text;

obj.lblproductname.Text = txtproductname.Text;

obj.lblproductquantity.Text = txtpurchasequantity.Text;

obj.lbltotalcost.Text = txttotalcost.Text;

obj.Show();

}

}

catch (SqlException ex)

{

MessageBox.Show(ex.Message);

}

finally

{

con.CloseConnection();

}

}

private void frmPurchaseProduct\_FormClosing(object sender, FormClosingEventArgs e)

{

if (MessageBox.Show("Do You Want To Exit", "Exit", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

this.Dispose();

}

else

{

e.Cancel = true;

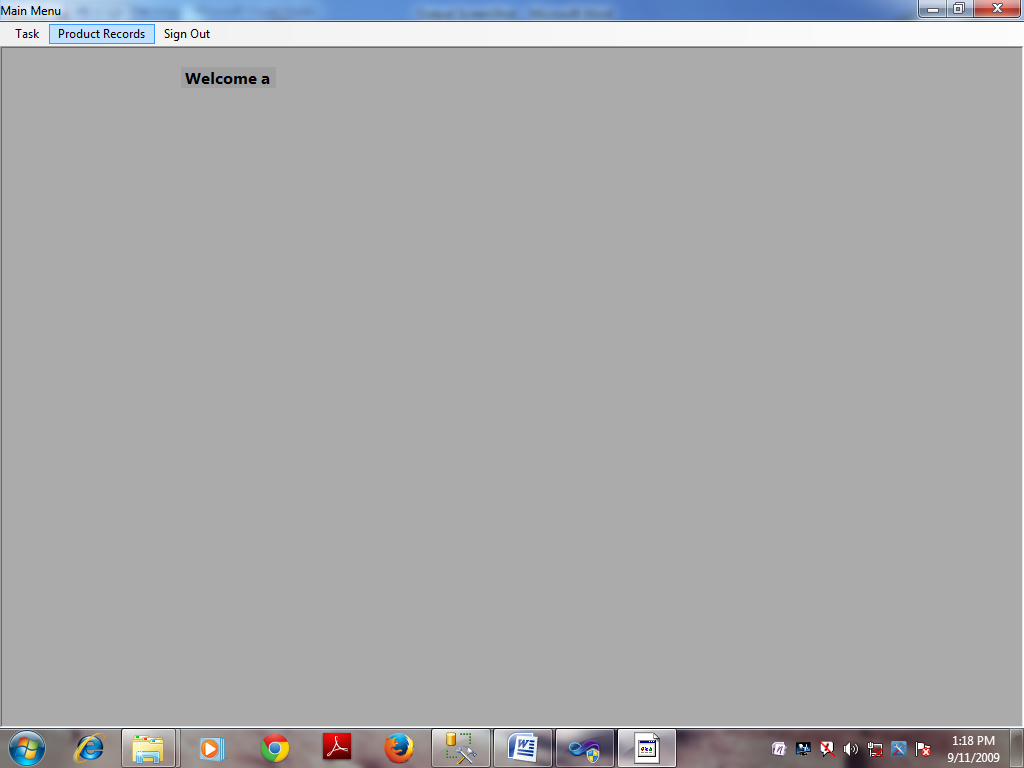
}

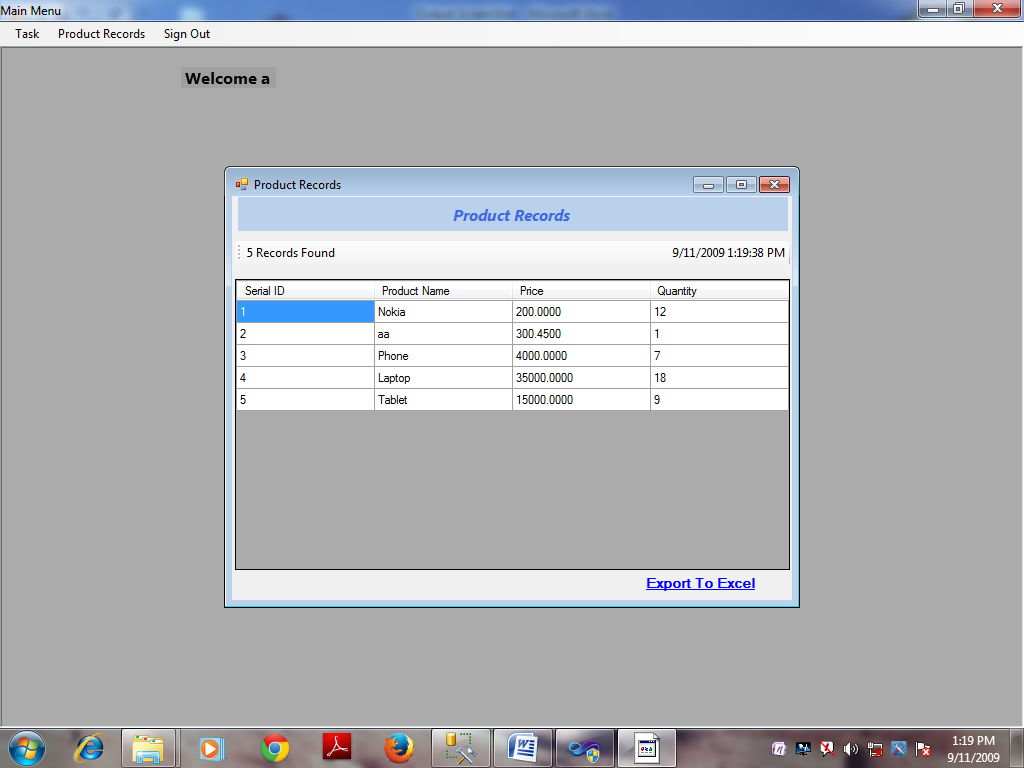
}

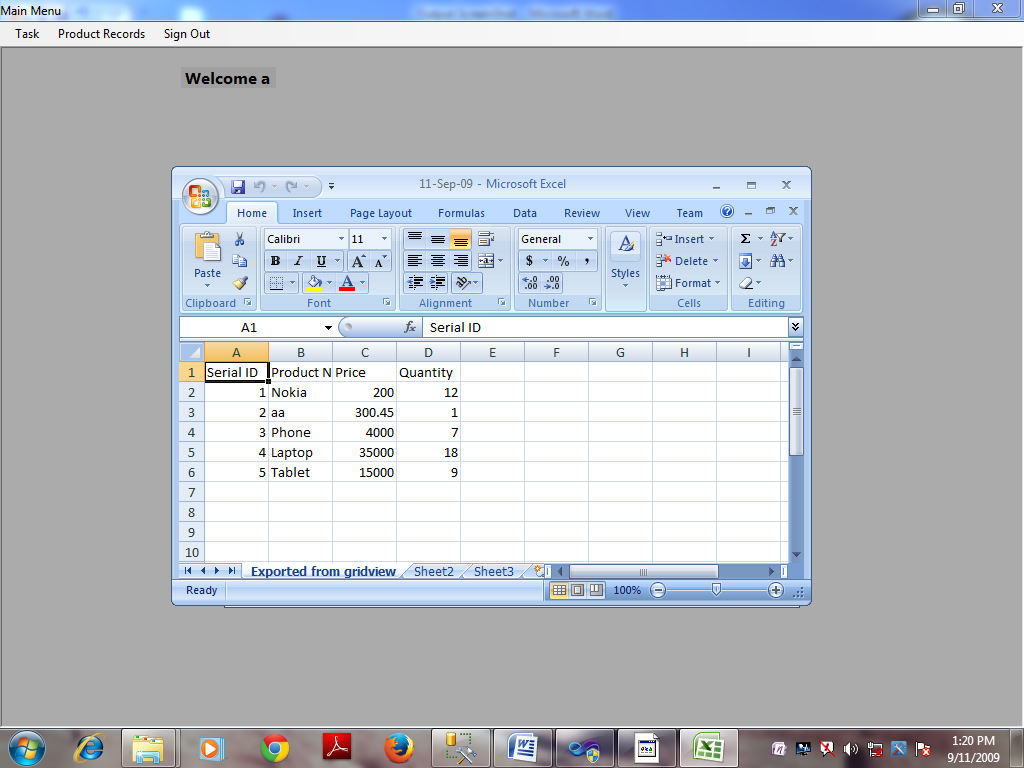
}

}

And now Show product Records module







In this module my .cs file is

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.IO;

using System.Data.SqlClient;

using System.Windows.Forms;

namespace Inventory\_Solutions

{

public partial class frmProductRecords : Form

{

DirectoryInfo Product\_Records;

DBConnect con = new DBConnect();

Timer timer1 = new Timer();

public frmProductRecords()

{

InitializeComponent();

}

private void frmProductRecords\_Load(object sender, EventArgs e)

{

timer1.Start();

timer1.Tick += new EventHandler(timer1\_Tick);

dataGridView1.Columns.Add("1", "Serial ID");

dataGridView1.Columns.Add("2", "Product Name");

dataGridView1.Columns.Add("3", "Price");

dataGridView1.Columns.Add("4", "Quantity");

BindGrid();

}

public void BindGrid()

{

try

{

con.OpenConnection();

string str = "Select \* from tbl\_ProductRecords";

SqlCommand cmd = new SqlCommand(str, DBConnect.Connection);

SqlDataReader dr = cmd.ExecuteReader();

int records = 0;

if (dr.HasRows)

{

while (dr.Read())

{

int n = dataGridView1.Rows.Add();

for (int i = 0; i < 4; i++)

{

dataGridView1.Rows[n].Cells[i].Value = dr.GetValue(i).ToString();

}

records++;

}

toolStripLabel1.Text = records + " Records Found";

}

else

{

toolStripLabel1.Text = "No Records Found";

}

}

catch (SqlException ex)

{

MessageBox.Show(ex.Message);

}

finally

{

con.CloseConnection();

}

}

void timer1\_Tick(object sender, EventArgs e)

{

toolStripLabel2.Text = DateTime.Now.ToString();

}

private void frmProductRecords\_FormClosing(object sender, FormClosingEventArgs e)

{

if (MessageBox.Show("Do You Want to Close", "Close", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

this.Dispose();

}

else

{

e.Cancel = true;

}

}

private void linkLabel1\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

{

Microsoft.Office.Interop.Excel.\_Application app = new Microsoft.Office.Interop.Excel.Application();

// creating new WorkBook within Excel application

Microsoft.Office.Interop.Excel.\_Workbook workbook = app.Workbooks.Add(Type.Missing);

// creating new Excelsheet in workbook

Microsoft.Office.Interop.Excel.\_Worksheet worksheet = null;

// see the excel sheet behind the program

app.Visible = true;

// get the reference of first sheet. By default its name is Sheet1.

// store its reference to worksheet

worksheet = workbook.Sheets["Sheet1"];

worksheet = workbook.ActiveSheet;

// changing the name of active sheet

worksheet.Name = "Exported from gridview";

// storing header part in Excel

for (int i = 1; i < dataGridView1.Columns.Count + 1; i++)

{

worksheet.Cells[1, i] = dataGridView1.Columns[i - 1].HeaderText;

}

// storing Each row and column value to excel sheet

for (int i = 0; i <= dataGridView1.Rows.Count - 1; i++)

{

for (int j = 0; j < dataGridView1.Columns.Count; j++)

{

worksheet.Cells[i + 2, j + 1] = dataGridView1.Rows[i].Cells[j].Value.ToString().Trim();

}

}

// save the application

DirectoryInfo Product\_Records = new DirectoryInfo(Application.StartupPath + "//Product\_Records//");

if (Product\_Records.Exists)

{

}

else

{

Product\_Records.Create();

}

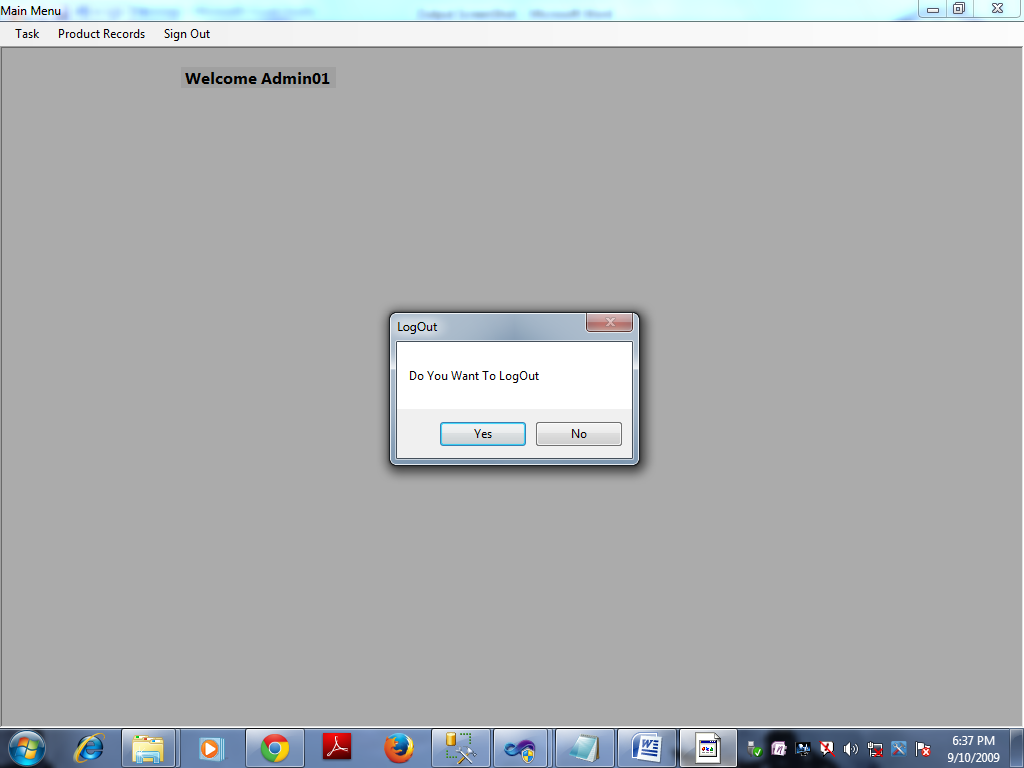
Environment.GetFolderPath(Environment.SpecialFolder.MyDocuments);

workbook.SaveAs(Application.StartupPath + "\\Product\_Records\\" + System.DateTime.Now.ToString("dd-MMM-yy") + ".xls");

}

}

}



Now it’s a set up of my software.



In all program file my software is



This final window application done by me.

Thanks