21110324 – Lương Đăng Khôi
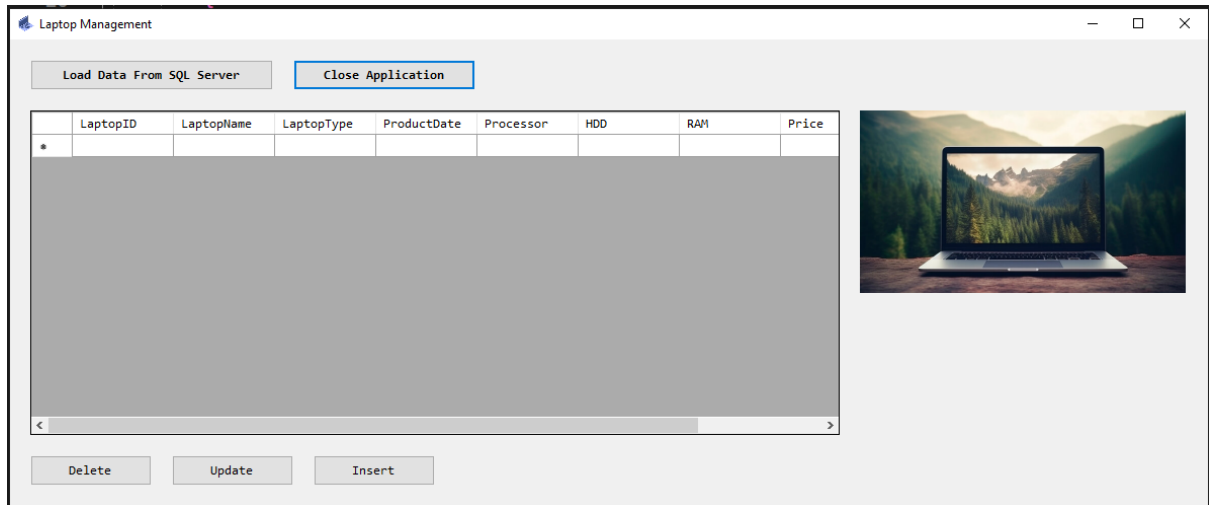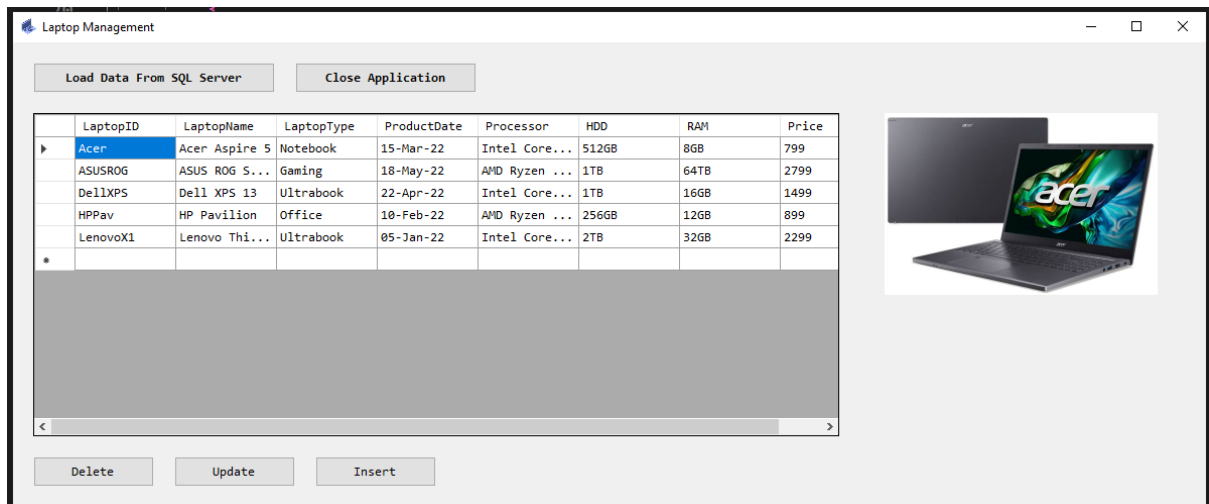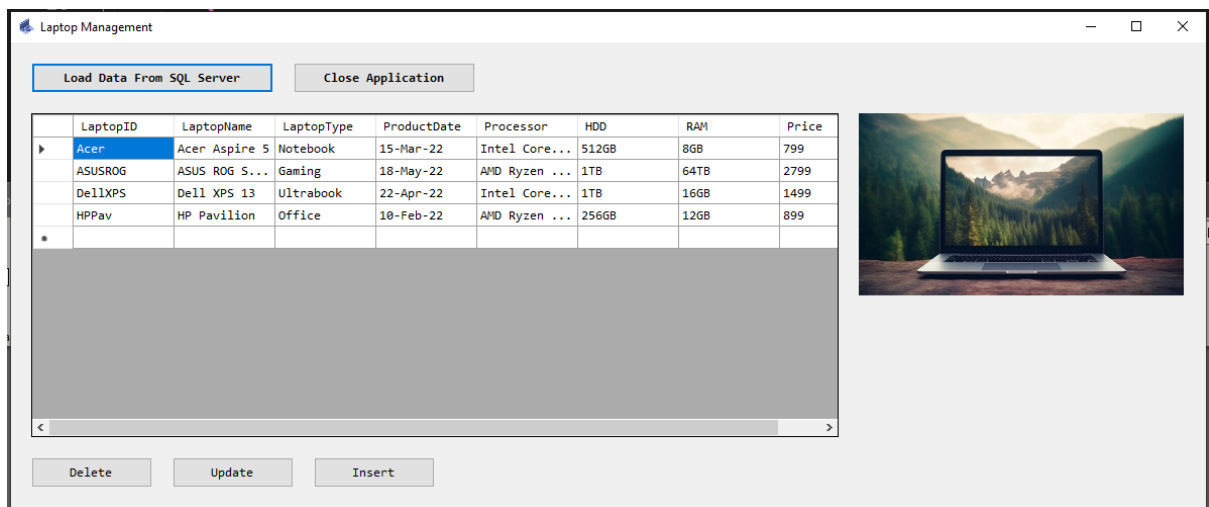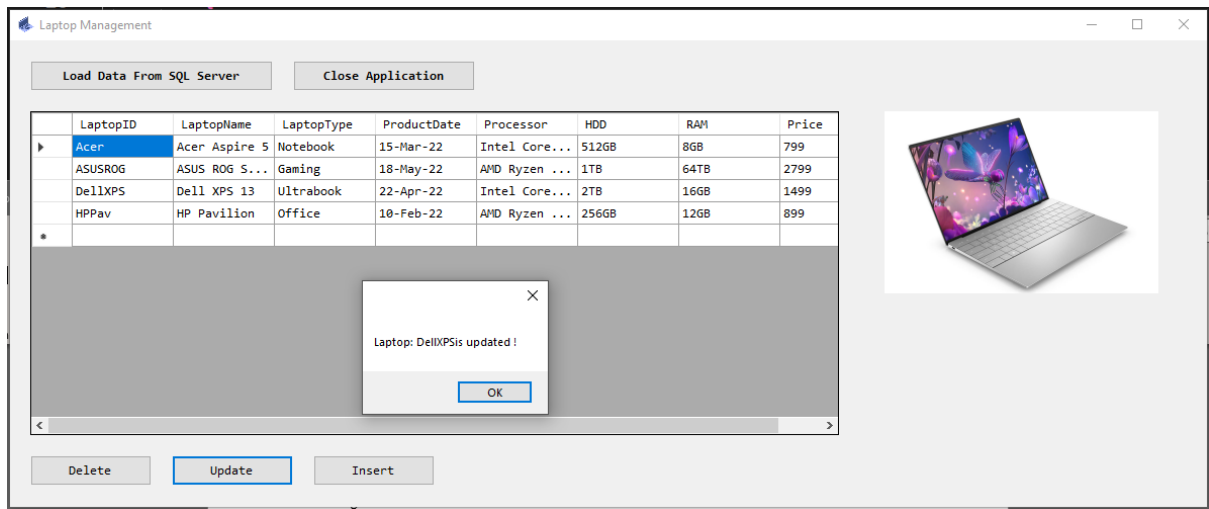
Yêu cầu 1:

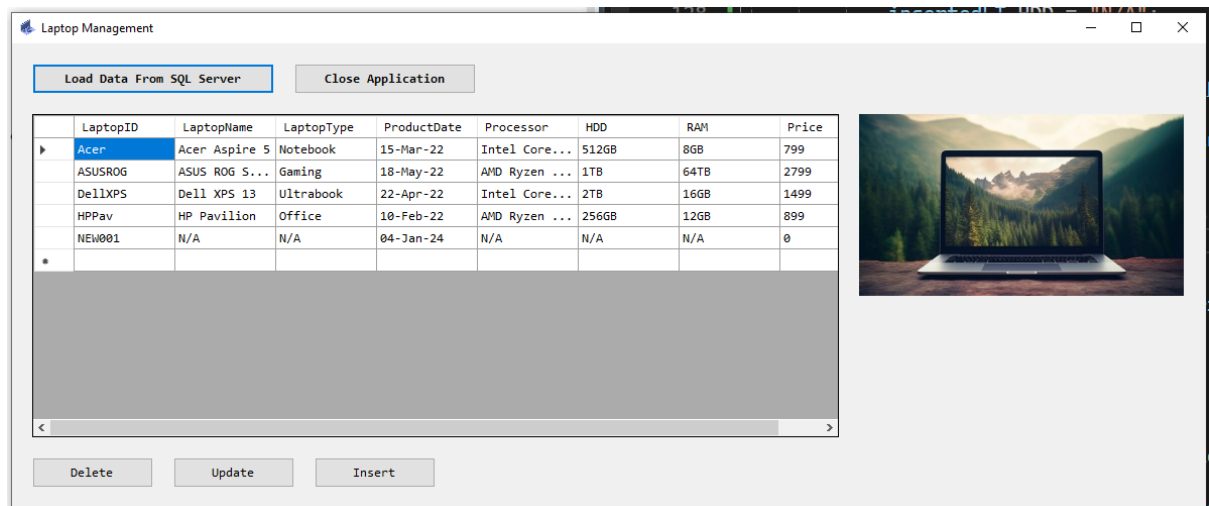- Tương tác với nút Load Data

- Tương tác với nút Delete

- Tương tác với nét Update

- Tương tác với nút Insert

Source code:

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

namespace LaptopLINQ
{
    public partial class MainForm : Form
    {
        static string ProjectPath = Directory.GetParent(Directory.GetCurrentDirectory()).Parent.FullName;
        LaptopDBDataContext db = new LaptopDBDataContext();

        public MainForm()
        {
            InitializeComponent();
        }
```

```csharp
private void MainForm_Load(object sender, EventArgs e)
{
    picLaptop.Image = Image.FromFile(ProjectPath + "\\Data\\Laptop.jpg");
    picLaptop.SizeMode = PictureBoxSizeMode.Zoom;
}

private void btnClose_Click(object sender, EventArgs e)
{
    Application.Exit();
}

private void DisplayData()
{
    dgwLaptopList.DataSource = (from lt in db.Laptops select lt);
    dgwLaptopList.Columns["colImageName"].Visible = false;
    dgwLaptopList.Columns["colLaptopID"].ReadOnly = true;
}

private void btnLoadDataFromSQL_Click(object sender, EventArgs e)
{
    DisplayData();
}

private void dgwLaptopList_SelectionChanged(object sender, EventArgs e)
{
    if (dgwLaptopList.Focused == false || dgwLaptopList.CurrentCell == null)
        return;

    int RowIndex = dgwLaptopList.CurrentRow.Index;
    string SelectedLaptopID = dgwLaptopList.Rows[RowIndex].Cells[0].Value.ToString();

    string LaptopImage = (from lt in db.Laptops
                where lt.LaptopID == SelectedLaptopID
                select lt.ImageName).FirstOrDefault();

    if (LaptopImage == null)
        return;

    picLaptop.Image = Image.FromFile(ProjectPath + "\\Data\\" + LaptopImage);
    picLaptop.SizeMode = PictureBoxSizeMode.Zoom;
}

private void btnDelete_Click(object sender, EventArgs e)
{
    // Delete in Database
    int RowIndex = dgwLaptopList.CurrentRow.Index;
    string SelectedLaptopID = dgwLaptopList.Rows[RowIndex].Cells[0].Value.ToString();

    Laptop deleteLT = (from lt in db.Laptops
                where lt.LaptopID == SelectedLaptopID
                select lt).SingleOrDefault();
    db.Laptops.DeleteOnSubmit(deleteLT);
    db.SubmitChanges();
    DisplayData();

    MessageBox.Show("Laptop: " + deleteLT.LaptopID + "is deleted !");
}

private void btnUpdate_Click(object sender, EventArgs e)
{
    int RowIndex = dgwLaptopList.CurrentRow.Index;
    string SelectedLaptopID = dgwLaptopList.Rows[RowIndex].Cells[0].Value.ToString();
```

```csharp
            Laptop updateLT = (from lt in db.Laptops
                              where lt.LaptopID == SelectedLaptopID
                              select lt).SingleOrDefault();

            updateLT.LaptopID = dgwLaptopList.Rows[RowIndex].Cells[0].Value.ToString();
            updateLT.LaptopName = dgwLaptopList.Rows[RowIndex].Cells[1].Value.ToString();
            updateLT.LaptopType = dgwLaptopList.Rows[RowIndex].Cells[2].Value.ToString();
            updateLT.ProductDate =
DateTime.Parse(dgwLaptopList.Rows[RowIndex].Cells[3].Value.ToString());
            updateLT.Processor = dgwLaptopList.Rows[RowIndex].Cells[4].Value.ToString();
            updateLT.HDD = dgwLaptopList.Rows[RowIndex].Cells[5].Value.ToString();
            updateLT.RAM = dgwLaptopList.Rows[RowIndex].Cells[6].Value.ToString();
            updateLT.Price = Convert.ToInt32(dgwLaptopList.Rows[RowIndex].Cells[7].Value.ToString());
            updateLT.ImageName = (updateLT.LaptopID.ToString() + ".jpg");

            db.SubmitChanges();
            DisplayData();
            MessageBox.Show("Laptop: " + updateLT.LaptopID + "is updated !");
        }

        private void btnInsert_Click(object sender, EventArgs e)
        {
            Laptop checkedLT;
            List<Laptop> LTList = (from lt in db.Laptops
                                 where lt.LaptopID.Contains("NEW")
                                 select lt).ToList();

            string NewID = "NEW";
            if (LTList.Count == 0)
                NewID = NewID + "001";
            else
            {
                checkedLT = LTList.Last();
                NewID = NewID + NextId(checkedLT.LaptopID.Substring(checkedLT.LaptopID.Length - 3));
            }

            Laptop insertedLT = new Laptop();
            insertedLT.LaptopID = NewID;
            insertedLT.LaptopName = "N/A";
            insertedLT.LaptopType = "N/A";
            insertedLT.ProductDate = DateTime.Now;
            insertedLT.Processor = "N/A";
            insertedLT.HDD = "N/A";
            insertedLT.RAM = "N/A";
            insertedLT.Price = 0;
            insertedLT.ImageName = NewID + ".jpg";

            db.Laptops.InsertOnSubmit(insertedLT);
            db.SubmitChanges();

            MessageBox.Show("New Laptop is inserted! Please update data and press button Update.");
        }

        public string NextId(string currentId)
        {
            int i = 0;
            if (int.TryParse(currentId, out i))
            {
                i++;
                return (i.ToString().PadLeft(3, '0'));
            }
            throw new System.ArgumentException("Non-numeric string passed as argument");
        }
```

```
    }
}
```

Yêu cầu 2:

- Chương trình có nhiều form và nhiều tính năng nên để tóm gọn, dưới đây sẽ là các tính năng theo yêu cầu.

* Login Form:

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

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

namespace CoffeeM
{
    public partial class LoginForm : Form
    {
        CoffeeShopDataContext db = new CoffeeShopDataContext();

        public LoginForm()
        {
            InitializeComponent();
        }

        public string UserName
        {
            get { return txtUserName.Text; }
        }

        private void LoginForm_Load(object sender, EventArgs e)
        {

        }

        private void btnLogin_Click(object sender, EventArgs e)
        {
            Account account = (from ac in db.Accounts
                        where ac.UserName == txtUserName.Text
                        select ac).SingleOrDefault();

            if ((txtUserName.Text == "" && txtPassword.Text == "")
                || (txtUserName.Text == "admin" && txtPassword.Text == "admin")
                || account.UserName != null)
            {
                ManageForm mf = new ManageForm();
                Hide();
                mf.ShowDialog();
                txtUserName.Clear();
                txtPassword.Clear();
                txtUserName.Focus();
                Show();
            }
            else
            {
                MessageBox.Show("Account is not exist");
                txtPassword.Clear();
                txtPassword.Focus();
            }

        }

        private void btnCancel_Click(object sender, EventArgs e)
        {
            Application.Exit();
        }
    }
}
```

* Manage Form:

```csharp
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 CoffeeM
{
    public partial class ManageForm : Form
    {
        public ManageForm()
        {
            InitializeComponent();
        }

        private void smiLogOut_Click(object sender, EventArgs e)
        {
            Hide();
            LoginForm lf = new LoginForm();
            lf.ShowDialog();
        }

        private void smiChangePassword_Click(object sender, EventArgs e)
        {
            LoginForm lf = new LoginForm();
            string CurrentUserName = lf.UserName;

            if (CurrentUserName == "admin" || CurrentUserName == "")
            {
                MessageBox.Show("Admin can't change password here!");
            }
            else
            {
                AccountInfo ai = new AccountInfo();
                ai.ShowDialog();
            }
        }

        private void smiAdmin_Click(object sender, EventArgs e)
        {
            LoginForm lf = new LoginForm();
            string CurrentUserName = lf.UserName;

            if (CurrentUserName == "admin" || CurrentUserName == "")
            {
                Hide();
                AdminForm af = new AdminForm();
                af.ShowDialog();
            }
            else
                MessageBox.Show("You are not ADMIN!");

        }

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

```
            // TODO: This line of code loads data into the 'coffeeShopDataSet5.Coffee' table. You can move, or
remove it, as needed.
            this.coffeeTableAdapter1.Fill(this.coffeeShopDataSet5.Coffee);
            // TODO: This line of code loads data into the 'coffeeShopDataSet4.Coffee' table. You can move, or
remove it, as needed.
            this.coffeeTableAdapter.Fill(this.coffeeShopDataSet4.Coffee);
            // TODO: This line of code loads data into the 'coffeeShopDataSet3.CoffeeTable' table. You can
move, or remove it, as needed.
            this.coffeeTableTableAdapter.Fill(this.coffeeShopDataSet3.CoffeeTable);

        }
    }
}


* Admin Form:

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

namespace CoffeeM
{
    public partial class AdminForm : Form
    {
        public string ProjectPath = Directory.GetParent(Directory.GetCurrentDirectory()).Parent.FullName;
        CoffeeShopDataContext db = new CoffeeShopDataContext();

        public AdminForm()
        {
            InitializeComponent();

            AddBindingCoffee();
            AddBindingTable();
            AddBindingAccount();
        }

        private void AdminForm_Load(object sender, EventArgs e)
        {
            // TODO: This line of code loads data into the 'coffeeShopDataSet2.Coffee' table. You can move, or
remove it, as needed.
            this.coffeeTableAdapter.Fill(this.coffeeShopDataSet2.Coffee);
            // TODO: This line of code loads data into the 'coffeeShopDataSet1.CoffeeTable' table. You can
move, or remove it, as needed.
            this.coffeeTableTableAdapter.Fill(this.coffeeShopDataSet1.CoffeeTable);
            // TODO: This line of code loads data into the 'coffeeShopDataSet.Account' table. You can move, or
remove it, as needed.
            this.accountTableAdapter.Fill(this.coffeeShopDataSet.Account);

            picCoffee.Image = Image.FromFile(ProjectPath + "\\Data\\" + "Coffee.jpg");
        }

        void AddBindingCoffee()
        {
            txtIDCoffee.DataBindings.Add(new Binding("Text", dgvCoffee.DataSource, "CoffeeID"));
            txtNameCoffee.DataBindings.Add(new Binding("Text", dgvCoffee.DataSource, "CoffeeName"));
```

```csharp
            txtPriceCoffee.DataBindings.Add(new Binding("Text", dgvCoffee.DataSource, "Price"));
        }

        void AddBindingTable()
        {
            txtIDTable.DataBindings.Add(new Binding("Text", dgvTable.DataSource, "TableID"));
            txtNameTable.DataBindings.Add(new Binding("Text", dgvTable.DataSource, "TableName"));
            cbxStatus.DataBindings.Add(new Binding("Text", dgvTable.DataSource, "TableStatus"));
        }

        void AddBindingAccount()
        {
            txtUserNameAccount.DataBindings.Add(new Binding("Text", dgvAccount.DataSource,
"UserName"));
            txtPasswordAccount.DataBindings.Add(new Binding("Text", dgvAccount.DataSource, "Password"));
            txtTypeAccount.DataBindings.Add(new Binding("Text", dgvAccount.DataSource, "Type"));
        }

        private void btnFindAccount_Click(object sender, EventArgs e)
        {
            DataTable dt = (DataTable)dgvAccount.DataSource;

            int RowIndex = dt.DefaultView.Find(txtFindAccount.Text);

            if (RowIndex != -1)
            {
                dgvAccount.CurrentCell = dgvAccount.Rows[RowIndex].Cells[0];
            }
            else
                MessageBox.Show("Account doesn't exist");
        }

        private void btnFindCoffee_Click(object sender, EventArgs e)
        {
            DataTable dt = (DataTable)dgvCoffee.DataSource;

            int RowIndex = dt.DefaultView.Find(txtFindCoffee.Text);

            if (RowIndex != -1)
            {
                dgvCoffee.CurrentCell = dgvCoffee.Rows[RowIndex].Cells[0];
            }
            else
                MessageBox.Show("Coffee not found");
        }
    }
}
```

* Account Info Form:

```csharp
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;
using static System.Windows.Forms.VisualStyles.VisualStyleElement.StartPanel;

namespace CoffeeM
```

```csharp
{
    public partial class AccountInfo : Form
    {
        CoffeeShopDataContext db = new CoffeeShopDataContext();

        public AccountInfo()
        {
            InitializeComponent();

            LoginForm lf = new LoginForm();
            txtUserName.Text = lf.UserName;

            Account CurrentAccount = (from ac in db.Accounts
                        where ac.UserName == txtUserName.Text
                        select ac).SingleOrDefault();
            txtUserName.Text = CurrentAccount.UserName;
            txtPassword.Text = CurrentAccount.Password;
        }

        private void btnCancel_Click(object sender, EventArgs e)
        {
            Hide();
        }

        private void btnAccept_Click(object sender, EventArgs e)
        {
            if (txtNewPass.Text == "")
                MessageBox.Show("You must enter new password or cancel to keep using password");
            else if (txtReNewPass.Text == "")
                MessageBox.Show("You must re enter new password");
            else if (txtNewPass.Text != txtReNewPass.Text)
            {
                MessageBox.Show("New password is not matching, please check again!");
            }
            else
            {
                // Update the password in the database
                Account CurrentAccount = (from ac in db.Accounts
                            where ac.UserName == txtUserName.Text
                            select ac).SingleOrDefault();

                CurrentAccount.Password = txtNewPass.Text;
                db.SubmitChanges();

                MessageBox.Show("Password changed successfully");
            }
        }

        private void AccountInfo_Load(object sender, EventArgs e)
        {

        }
    }
}
```