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 System.IO;

using Excel = Microsoft.Office.Interop.Excel;

using System.Reflection;

using Microsoft.Office.Core;

// This is the code for your desktop app.

// Press Ctrl+F5 (or go to Debug > Start Without Debugging) to run your app.

namespace DesktopApp1

{

public partial class DogMomApp : Form

{

public DogMomApp()

{

InitializeComponent();

}

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

{

var fileContent = string.Empty;

var filePath = string.Empty;

using (OpenFileDialog openFileDialog = new OpenFileDialog())

{

openFileDialog.InitialDirectory = "c:\\";

openFileDialog.Filter = "Microsoft Excel Comma Separated Values Files (\*.csv\*)|\*.csv\*|All files (\*.\*)|\*.\*";

openFileDialog.Title = "Please select your Excel file:";

openFileDialog.FilterIndex = 2;

openFileDialog.RestoreDirectory = true;

if (openFileDialog.ShowDialog() == DialogResult.OK)

{

//Get the path of specified file

filePath = openFileDialog.FileName;

var newFilePath = Path.ChangeExtension(filePath, ".csv");

File.Move(filePath, newFilePath);

if (checkBox1.Checked)

{

System.Diagnostics.Process.Start(newFilePath);

}

else { }

}

}

}

private void button2\_Click(object sender, EventArgs e)

{

var fileContent = string.Empty;

var filePath = string.Empty;

using (OpenFileDialog openFileDialog = new OpenFileDialog())

{

openFileDialog.InitialDirectory = "c:\\";

openFileDialog.Filter = "Microsoft Excel Comma Separated Values Files (\*.csv\*)|\*.csv\*|All files (\*.\*)|\*.\*";

openFileDialog.Title = "Please select your Excel file:";

openFileDialog.FilterIndex = 2;

openFileDialog.RestoreDirectory = true;

if (openFileDialog.ShowDialog() == DialogResult.OK)

{

//Get the path of specified file

filePath = openFileDialog.FileName;

var newFilePath = Path.ChangeExtension(filePath, ".xls");

File.Move(filePath, newFilePath);

if (checkBox1.Checked)

{

System.Diagnostics.Process.Start(newFilePath);

}

else { }

}

}

}

private void button3\_Click(object sender, EventArgs e)

{

var fileContent = string.Empty;

var filePath = string.Empty;

using (OpenFileDialog openFileDialog = new OpenFileDialog())

{

openFileDialog.InitialDirectory = "c:\\";

openFileDialog.Filter = "Microsoft Excel Comma Separated Values Files (\*.csv\*)|\*.csv\*|All files (\*.\*)|\*.\*";

openFileDialog.Title = "Please select your Excel file:";

openFileDialog.FilterIndex = 2;

openFileDialog.RestoreDirectory = true;

if (openFileDialog.ShowDialog() == DialogResult.OK)

{

//Get the path of specified file

filePath = openFileDialog.FileName;

var newFilePath = Path.ChangeExtension(filePath, ".ods");

File.Move(filePath, newFilePath);

if (checkBox1.Checked)

{

System.Diagnostics.Process.Start(newFilePath);

}

else { }

}

}

}

private void button6\_Click\_2(object sender, EventArgs e)

{

string excelFile = string.Empty;

using (OpenFileDialog excelOpen = new OpenFileDialog())

{

excelOpen.InitialDirectory = "c:\\";

excelOpen.Filter = "Microsoft Excel Comma Separated Values Files (\*.csv\*)|\*.csv\*|All files (\*.\*)|\*.\*";

excelOpen.Title = "Please select your Excel file:";

if (excelOpen.ShowDialog() == DialogResult.OK)

{

excelFile = excelOpen.FileName;

}

}

//ALT F11

//ALT I + M

Excel.Application oXL;

Excel.\_Workbook oWB;

Excel.\_Worksheet oSheet;

try

{

//Start Excel and get Application object.

oXL = new Excel.Application();

oXL.Visible = true;

oXL.WindowState = Excel.XlWindowState.xlMaximized;

//Get a new workbook.

oWB = oXL.Workbooks.Open(excelFile);

oSheet = (Excel.\_Worksheet)oWB.ActiveSheet;

keyStrokes(oSheet);

}

catch (Exception theException)

{

String errorMessage;

errorMessage = "Error: ";

errorMessage = String.Concat(errorMessage, theException.Message);

errorMessage = String.Concat(errorMessage, " Line: ");

errorMessage = String.Concat(errorMessage, theException.Source);

MessageBox.Show(errorMessage, "Error");

}

}

private void keyStrokes(Excel.\_Worksheet oWS)

{

DogMomApp.ActiveForm.WindowState = FormWindowState.Minimized;

string excelVBA;

excelVBA = Properties.Resources.ExcelCode;

Clipboard.SetText(excelVBA);

SendKeys.Send("%{F11}%{I}{M}");

SendKeys.Send("^{v}");

SendKeys.SendWait("{F5}");

SendKeys.SendWait("%{F11}");

}

private void keyStrokes2(bool tF)

{

if(tF == true)

{

DogMomApp.ActiveForm.WindowState = FormWindowState.Minimized;

string excelCalc;

excelCalc = Properties.Resources.ExcelVBA;

Clipboard.SetText(excelCalc);

SendKeys.Send("%{I}{M}");

SendKeys.Send("^{v}");

SendKeys.SendWait("{F5}");

SendKeys.SendWait("%{Q}");

}

else

{

//

}

}

private void button4\_Click(object sender, EventArgs e)

{

try

{

keyStrokes2(true);

}

catch (Exception theException)

{

String errorMessage;

errorMessage = "Error: ";

errorMessage = String.Concat(errorMessage, theException.Message);

errorMessage = String.Concat(errorMessage, " Line: ");

errorMessage = String.Concat(errorMessage, theException.Source);

MessageBox.Show(errorMessage, "Error");

}

}

}

}