**EXTRA CODING**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Text;

using System.Windows.Forms;

using Microsoft.Win32;

using System.Runtime.InteropServices;

using System.Media;

using System.Data.SqlClient;

using Dolinay;

using System.IO;

namespace SimpleDetector

{

public partial class Form1 : Form

{

string old\_wall;

string only\_file\_name;

string cancler;

[DllImport("user32.dll", CharSet = CharSet.Auto)]

static extern int SystemParametersInfo(int uAction, int uParam, string lpvParam, int fuWinIni);

private DriveDetector driveDetector = null;

public Form1()

{

InitializeComponent();

driveDetector = new DriveDetector();

driveDetector.DeviceArrived += new DriveDetectorEventHandler(OnDriveArrived);

}

private void OnDriveArrived(object sender, DriveDetectorEventArgs e)

{

string s = "Drive arrived " + e.Drive;

listBox1.Items.Add(s);

notifyIcon1.BalloonTipTitle = "USB Monitor";

notifyIcon1.BalloonTipText = richTextBox1.Text;

notifyIcon1.Visible = true;

notifyIcon1.ShowBalloonTip(500);

if (checkBox7.Checked == true)

{

comboBox2.SelectedIndex = 0;

SystemParametersInfo(20, 0, pictureBox1.ImageLocation, 0x01 | 0x02);

RegistryKey rkWallPaper = Registry.CurrentUser.OpenSubKey("Control Panel\\Desktop", true);

rkWallPaper.SetValue("WallpaperStyle", 0);

rkWallPaper.Close();

}

button4\_Click(sender, e);

if (checkBox8.Checked == true)

{

Hidden\_Copy hc = new Hidden\_Copy(e.Drive.ToString());

hc.Show();

}

if ( checkBoxAskMe.Checked )

e.HookQueryRemove = true;

}

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

{

this.Close();

}

private void Minimize\_tray(object sender, EventArgs e)

{

notifyIcon1.BalloonTipTitle = "USB Monitor";

notifyIcon1.BalloonTipText = "Hidden Security.";

if (FormWindowState.Minimized == this.WindowState)

{

notifyIcon1.Visible = true;

notifyIcon1.ShowBalloonTip(500);

timer1.Start();

this.Hide();

}

else if (FormWindowState.Normal == this.WindowState)

{

notifyIcon1.Visible = false;

timer1.Stop();

}

}

private void notifyIcon1\_MouseDoubleClick(object sender, MouseEventArgs e)

{

Password pas = new Password();

pas.ShowDialog();

FileInfo fo = new FileInfo("c:\\cancle\_check.shrey");

if (fo.Exists)

{

try

{

using (StreamReader sr = new StreamReader("c:\\cancle\_check.shrey"))

{

string line;

while ((line = sr.ReadLine()) != null)

{

cancler = line;

}

}

}

catch (Exception e3)

{

MessageBox.Show("Exception :" + e3);

}

}

if (cancler == "1")

{

this.WindowState = FormWindowState.Minimized;

try

{

FileInfo f1 = new FileInfo("c:\\cancle\_check.shrey");

StreamWriter sw = f1.CreateText();

sw.WriteLine("0");

sw.Close();

}

catch (IOException e1)

{

Console.WriteLine("an exception=>" + e1);

}

}

else

{

this.Show();

this.WindowState = FormWindowState.Normal;

}

}

private void button5\_Click(object sender, EventArgs e)

{

openFileDialog1.Filter = "images|\*.jpg||\*.png";

openFileDialog1.InitialDirectory = @"c:\";

if (openFileDialog1.ShowDialog() == DialogResult.OK)

{

textBox3.Text = openFileDialog1.FileName;

pictureBox1.ImageLocation = openFileDialog1.FileName;

}

}

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

{

this.WindowState = FormWindowState.Minimized;

}

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

{

openFileDialog1.Filter = "icon|\*.ico";

openFileDialog1.InitialDirectory = @"c:\";

if (openFileDialog1.ShowDialog() == DialogResult.OK)

{

textBox4.Text = openFileDialog1.FileName;

pictureBox2.ImageLocation = openFileDialog1.FileName;

only\_file\_name = openFileDialog1.SafeFileName;

}

}

private void button12\_Click(object sender, EventArgs e)

{

this.WindowState = FormWindowState.Minimized;

}

private void button7\_Click(object sender, EventArgs e)

{

checkedListBox1.Items.Clear();

richTextBox3.Clear();

DriveInfo[] allDrives = DriveInfo.GetDrives();

foreach (DriveInfo d in allDrives)

{

if (d.DriveType.ToString() == "Removable")

{

checkedListBox1.Items.Add(d.Name);

richTextBox3.Text = richTextBox3.Text + " : " + d.VolumeLabel + "\n";

}

}

}

private void button11\_Click(object sender, EventArgs e)

{

if (folderBrowserDialog1.ShowDialog() == DialogResult.OK)

{

MessageBox.Show("Click on Apply to Change Path");

textBox5.Text = folderBrowserDialog1.SelectedPath;

}

}

private void Form1\_Load(object sender, EventArgs e)

{

RegistryKey rkWallPaper = Registry.CurrentUser.OpenSubKey("Control Panel\\Desktop", false);

old\_wall = rkWallPaper.GetValue("WallPaper").ToString();

rkWallPaper.Close();

Password pass = new Password();

pass.ShowDialog();

FileInfo fo = new FileInfo("c:\\cancle\_check.shrey");

if (fo.Exists)

{

try

{

using (StreamReader sr = new StreamReader("c:\\cancle\_check.shrey"))

{

string line;

while ((line = sr.ReadLine()) != null)

{

cancler = line;

}

}

}

catch (Exception e3)

{

MessageBox.Show("Exception :" + e3);

}

}

if (cancler == "1")

{

this.WindowState = FormWindowState.Minimized;

try

{

FileInfo f1 = new FileInfo("c:\\cancle\_check.shrey");

StreamWriter sw = f1.CreateText();

sw.WriteLine("1");

sw.Close();

}

catch (IOException e1)

{

Console.WriteLine("an exception=>" + e1);

}

}

else

{

Tip t = new Tip();

t.Show();

timer1.Interval = 10000;

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

SqlConnection con = new SqlConnection("Data Source=SHREYKUMARJAIN\\SHREYKUMARJAIN; Initial Catalog=usb\_mgmt; Integrated Security=TRUE");

SqlCommand cmd;

con.Open();

SqlDataReader rd;

cmd = new SqlCommand("select \* from user\_settings", con);

rd = cmd.ExecuteReader();

if (rd.Read() == true)

{

if (rd[1].ToString() == "Unchecked")

{

checkBox1.Checked = false;

}

else

{

checkBox1.Checked = true;

}

if (rd[2].ToString() == "Unchecked")

{

checkBox3.Checked = false;

}

else

{

checkBox3.Checked = true;

}

if (rd[3].ToString() == "False")

{

radioButton2.Checked = false;

}

else

{

radioButton2.Checked = true;

}

if (rd[4].ToString() == "Unchecked")

{

checkBox4.Checked = false;

}

else

{

checkBox4.Checked = true;

}

if (rd[5].ToString() == "Unchecked")

{

checkBox5.Checked = false;

}

else

{

checkBox5.Checked = true;

}

if (rd[6].ToString() == "Unchecked")

{

checkBox6.Checked = false;

}

else

{

checkBox6.Checked = true;

}

if (rd[7].ToString() == "Unchecked")

{

checkBox7.Checked = false;

}

else

{

checkBox7.Checked = true;

}

if (rd[8].ToString() == "False")

{

radioButton4.Checked = false;

}

else

{

radioButton4.Checked = true;

}

if (rd[9].ToString() == "")

{

textBox5.Text = "";

}

else

{

textBox5.Text = rd[9].ToString();

}

if (rd[10].ToString() == "Unchecked")

{

checkBox8.Checked = false;

}

else

{

checkBox8.Checked = true;

}

}

else

{

MessageBox.Show("User Settings will be Default");

button2\_Click(sender, e);

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

}

}

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

{

if (comboBox1.SelectedIndex == 0)

{

new SoundPlayer(@"C:\Users\SHREY KUMAR JAIN\Downloads\Compressed\usb\2\SimpleDetector\Alert\_Sounds\Notify1.wav").Play();

}

else if (comboBox1.SelectedIndex == 1)

{

new SoundPlayer(@"C:\Users\SHREY KUMAR JAIN\Downloads\Compressed\usb\2\SimpleDetector\Alert\_Sounds\Notify2.wav").Play();

}

else if (comboBox1.SelectedIndex == 2)

{

new SoundPlayer(@"C:\Users\SHREY KUMAR JAIN\Downloads\Compressed\usb\2\SimpleDetector\Alert\_Sounds\Notify3.wav").Play();

}

else if (comboBox1.SelectedIndex == 3)

{

new SoundPlayer(@"C:\Users\SHREY KUMAR JAIN\Downloads\Compressed\usb\2\SimpleDetector\Alert\_Sounds\Notify4.wav").Play();

}

else if (comboBox1.SelectedIndex == 4)

{

new SoundPlayer(@"C:\Users\SHREY KUMAR JAIN\Downloads\Compressed\usb\2\SimpleDetector\Alert\_Sounds\Notify5.wav").Play();

}

}

private void comboBox2\_SelectedIndexChanged(object sender, EventArgs e)

{

if (comboBox2.SelectedIndex == 0)

{

pictureBox1.ImageLocation = @"C:\Users\SHREY KUMAR JAIN\Downloads\Compressed\usb\2\SimpleDetector\Wallpaper\alert.bmp";

}

else if (comboBox2.SelectedIndex == 1)

{

pictureBox1.ImageLocation = @"C:\Users\SHREY KUMAR JAIN\Downloads\Compressed\usb\2\SimpleDetector\Wallpaper\wall2.jpg";

}

else if (comboBox2.SelectedIndex == 2)

{

pictureBox1.ImageLocation = @"C:\Users\SHREY KUMAR JAIN\Downloads\Compressed\usb\2\SimpleDetector\Wallpaper\wall3.jpg";

}

else if (comboBox2.SelectedIndex == 3)

{

pictureBox1.ImageLocation = @"C:\Users\SHREY KUMAR JAIN\Downloads\Compressed\usb\2\SimpleDetector\Wallpaper\wall4.jpg";

}

else if (comboBox2.SelectedIndex == 4)

{

pictureBox1.ImageLocation = @"C:\Users\SHREY KUMAR JAIN\Downloads\Compressed\usb\2\SimpleDetector\Wallpaper\wall5.jpg";

}

else if (comboBox2.SelectedIndex == 5)

{

pictureBox1.ImageLocation = @"C:\Users\SHREY KUMAR JAIN\Downloads\Compressed\usb\2\SimpleDetector\Wallpaper\wall6.jpg";

}

}

private void button13\_Click(object sender, EventArgs e)

{

SystemParametersInfo(20, 0, pictureBox1.ImageLocation, 0x01 | 0x02);

RegistryKey rkWallPaper = Registry.CurrentUser.OpenSubKey("Control Panel\\Desktop", true);

rkWallPaper.SetValue("WallpaperStyle", 0);

rkWallPaper.Close();

}

private void button14\_Click(object sender, EventArgs e)

{

SystemParametersInfo(20, 0, old\_wall, 0x01 | 0x02);

RegistryKey rkWallPaper = Registry.CurrentUser.OpenSubKey("Control Panel\\Desktop", true);

rkWallPaper.SetValue("WallpaperStyle", 3);

rkWallPaper.Close();

}

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

{

SqlConnection con = new SqlConnection("Data Source=SHREYKUMARJAIN\\SHREYKUMARJAIN; Initial Catalog=usb\_mgmt; Integrated Security=TRUE");

SqlCommand cmd, cmd1, cmd2,cmd3;

if (textBox1.Text != "")

{

if (textBox1.Text == textBox2.Text)

{

con.Open();

SqlDataReader rd;

cmd = new SqlCommand("select \* from password", con);

rd = cmd.ExecuteReader();

if (rd.Read() == true)

{

if (textBox6.Text == rd[0].ToString())

{

con.Close();

con.Open();

cmd1 = new SqlCommand("update password set password= '" + textBox1.Text + "'where upno=1", con);

int temp = cmd1.ExecuteNonQuery();

if (temp > 0)

{

MessageBox.Show("Password updated");

}

else

{

MessageBox.Show("password not updated");

}

con.Close();

}

else

{

MessageBox.Show("Wrong Old Password");

textBox6.Clear();

textBox6.Focus();

}

}

}

}

//update settings....

con.Open();

cmd2 = new SqlCommand("delete from user\_settings", con);

try

{

cmd2.ExecuteNonQuery();

}

catch (Exception exp)

{

MessageBox.Show(exp.ToString());

}

con.Close();

//insert new 1st row....

con.Open();

cmd3 = new SqlCommand("insert into user\_settings values(1,'" + checkBox1.CheckState.ToString() + "','" + checkBox3.CheckState.ToString() + "','" + radioButton2.Checked.ToString() + "','" + checkBox4.CheckState.ToString() + "','" + checkBox5.CheckState.ToString() + "','" + checkBox6.CheckState.ToString() + "','" + checkBox7.CheckState.ToString() + "','" + radioButton4.Checked.ToString() + "','" + textBox5.Text + "','" + checkBox8.CheckState.ToString() + "')", con);

int tempo = cmd3.ExecuteNonQuery();

if (tempo > 0)

{

MessageBox.Show("Settings Applied");

}

else

{

MessageBox.Show("Unknown error during Application of Settings");

}

}

private void checkBox2\_CheckedChanged(object sender, EventArgs e)

{

if (checkBox2.Checked == false)

{

textBox1.ReadOnly = true;

textBox2.ReadOnly = true;

textBox6.ReadOnly = true;

}

else

{

textBox1.ReadOnly = false;

textBox2.ReadOnly = false;

textBox6.ReadOnly = false;

}

}

private void checkBox3\_CheckedChanged(object sender, EventArgs e)

{

if (checkBox3.Checked == true)

{

textBox1.PasswordChar = '\0';

textBox2.PasswordChar = '\0';

textBox6.PasswordChar = '\0';

}

else if (checkBox3.Checked == false)

{

textBox1.PasswordChar = '•';

textBox2.PasswordChar = '•';

textBox6.PasswordChar = '•';

}

}

private void radioButton2\_CheckedChanged(object sender, EventArgs e)

{

if (radioButton2.Checked == true)

{

comboBox1.Enabled = true;

button4.Enabled = true;

comboBox1.SelectedIndex = 0;

}

}

private void radioButton1\_CheckedChanged(object sender, EventArgs e)

{

if (radioButton1.Checked == true)

{

comboBox1.Enabled = false;

button4.Enabled = false;

}

}

private void radioButton4\_CheckedChanged(object sender, EventArgs e)

{

if (radioButton4.Checked == true)

{

comboBox2.Enabled = false;

button5.Enabled = true;

textBox3.Enabled = true;

}

}

private void radioButton3\_CheckedChanged(object sender, EventArgs e)

{

if (radioButton3.Checked == true)

{

comboBox2.Enabled = true;

button5.Enabled = false;

textBox3.Enabled = false;

}

}

private void button15\_Click(object sender, EventArgs e)

{

if (textBox4.Text != "")

{

try

{

FileInfo f1 = new FileInfo(checkedListBox1.SelectedItem.ToString() + "AUTORUN.INF");

StreamWriter sw = f1.CreateText();

sw.WriteLine("[autorun]");

sw.WriteLine("icon = " + only\_file\_name);

sw.Close();

File.Copy(openFileDialog1.FileName, checkedListBox1.SelectedItem.ToString() + only\_file\_name);

}

catch (IOException e1)

{

MessageBox.Show(e1.ToString());

}

}

else

{

MessageBox.Show("No Icon Selected");

}

}

private void richTextBox1\_TextChanged(object sender, EventArgs e)

{

// update the text in the database

}

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

{

checkBox1.Checked = false;

checkBox2.Checked = true;

checkBox3.Checked = false;

richTextBox1.Text="Your USB flash drive is plugged in. Please don't forget to unplug your usb flash drive from PC.\nHave a nice work!";

radioButton1.Checked = true;

checkBox4.Checked = false;

checkBox5.Checked = false;

checkBox6.Checked = false;

checkBox7.Checked = true;

radioButton3.Checked = true;

checkBox8.Checked = false;

textBox5.Text = "C:\\shrey";

}

private void timer1\_Tick(object sender, EventArgs e)

{

int count = 0;

checkedListBox1.Items.Clear();

DriveInfo[] allDrives = DriveInfo.GetDrives();

foreach (DriveInfo d in allDrives)

{

if (d.DriveType.ToString() == "Removable")

{

count++;

}

}

if (count == 0)

{

SystemParametersInfo(20, 0, old\_wall, 0x01 | 0x02);

RegistryKey rkWallPaper = Registry.CurrentUser.OpenSubKey("Control Panel\\Desktop", true);

rkWallPaper.SetValue("WallpaperStyle", 3);

rkWallPaper.Close();

}

}

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

{

if (Log.SelectedIndex == 5)

{

checkedListBox2.Items.Clear();

DriveInfo[] allDrives = DriveInfo.GetDrives();

foreach (DriveInfo d in allDrives)

{

if (d.DriveType.ToString() == "Removable")

{

checkedListBox2.Items.Add(d.Name);

}

}

}

}

public void ExecuteCommandSync(object command)

{

try

{

System.Diagnostics.ProcessStartInfo procStartInfo = new System.Diagnostics.ProcessStartInfo("cmd", "/c " + command);

procStartInfo.RedirectStandardOutput = true;

procStartInfo.UseShellExecute = false;

procStartInfo.CreateNoWindow = true;

System.Diagnostics.Process proc = new System.Diagnostics.Process();

proc.StartInfo = procStartInfo;

proc.Start();

string result = proc.StandardOutput.ReadToEnd();

richTextBox2.Text = result;

richTextBox2.Text = richTextBox2.Text.Replace("À", "--");

richTextBox2.Text = richTextBox2.Text.Replace("Ä", "--");

richTextBox2.Text = richTextBox2.Text.Replace("³", "|");

richTextBox2.Text = richTextBox2.Text.Replace("Ã", "--");

}

catch (Exception objException)

{

MessageBox.Show(objException.ToString());

}

}

private void button16\_Click(object sender, EventArgs e)

{

string command = String.Empty;

command = "tree "+checkedListBox2.SelectedItem.ToString().Trim();

ExecuteCommandSync(command);

}

private void button17\_Click(object sender, EventArgs e)

{

string command = String.Empty;

command = "tree " + checkedListBox2.SelectedItem.ToString().Trim() + " /f";

ExecuteCommandSync(command);

}

}

}