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.SQLite;

using System.Net.Sockets;

using System.Net;

using System.Threading;

using System.IO;

using System.Drawing.Imaging;

using iTextSharp.text;

using iTextSharp.text.pdf;

using System.Diagnostics;

using System.Security.Cryptography;

namespace Client

{

public partial class Infor\_Cilent : Form

{

private SQLiteConnection conn;

private SQLiteCommand cmd;

static string encryptKey = "Oyea";

public Infor\_Cilent()

{

InitializeComponent();

progressBar1.Maximum = 10;

progressBar1.Value = 0;

progressBar1.Step = 5;

GetFolderTreeView("");

GetListPeakView();

}

/// <summary>

/// 解密字符串

/// </summary>

/// <param name="str">要解密的字符串</param>

/// <returns>解密后的字符串</returns>

static string Decrypt(string str)

{

DESCryptoServiceProvider descsp = new DESCryptoServiceProvider(); //实例化加/解密类对象

byte[] key = Encoding.Unicode.GetBytes(encryptKey); //定义字节数组，用来存储密钥

byte[] data = Convert.FromBase64String(str);//定义字节数组，用来存储要解密的字符串

MemoryStream MStream = new MemoryStream(); //实例化内存流对象

//使用内存流实例化解密流对象

CryptoStream CStream = new CryptoStream(MStream, descsp.CreateDecryptor(key, key), CryptoStreamMode.Write);

CStream.Write(data, 0, data.Length); //向解密流中写入数据

CStream.FlushFinalBlock(); //释放解密流

return Encoding.Unicode.GetString(MStream.ToArray()); //返回解密后的字符串

}

/// <summary> /// 加密字符串

/// </summary>

/// <param name="str">要加密的字符串</param>

/// <returns>加密后的字符串</returns>

static string Encrypt(string str)

{

DESCryptoServiceProvider descsp = new DESCryptoServiceProvider(); //实例化加/解密类对象

byte[] key = Encoding.Unicode.GetBytes(encryptKey); //定义字节数组，用来存储密钥

byte[] data = Encoding.Unicode.GetBytes(str);//定义字节数组，用来存储要加密的字符串

MemoryStream MStream = new MemoryStream(); //实例化内存流对象

//使用内存流实例化加密流对象

CryptoStream CStream = new CryptoStream(MStream, descsp.CreateEncryptor(key, key), CryptoStreamMode.Write);

CStream.Write(data, 0, data.Length); //向加密流中写入数据

CStream.FlushFinalBlock(); //释放加密流

return Convert.ToBase64String(MStream.ToArray());//返回加密后的字符串

}

private void GetListPeakView()

{

ListViewItem item = new ListViewItem();

listView1.View = View.Details;

ColumnHeader clh;

clh = new ColumnHeader();

clh.Text = "#";

listView1.Columns.Add("\*", 10, HorizontalAlignment.Left);

clh = new ColumnHeader();

clh.Text = "PeakName";

listView1.Columns.Add(clh);

listView1.View = View.Details;

clh = new ColumnHeader();

clh.Text = "RT";

listView1.Columns.Add(clh);

listView1.View = View.Details;

clh = new ColumnHeader();

clh.Text = "Area";

listView1.Columns.Add(clh);

listView1.View = View.Details;

clh = new ColumnHeader();

clh.Text = "%Area";

listView1.Columns.Add(clh);

listView1.View = View.Details;

clh = new ColumnHeader();

clh.Text = "Height";

listView1.Columns.Add(clh);

listView1.View = View.Details;

clh = new ColumnHeader();

clh.Text = "Amount";

listView1.Columns.Add(clh);

listView1.View = View.Details;

clh = new ColumnHeader();

clh.Text = "Units";

foreach (ColumnHeader column in this.listView1.Columns)

{

clh.Width = -2;

}

listView1.Columns.Add(clh);

//結束更新

//listView2.EndUpdate();

}

private void GetFolderTreeView(string asPath)

{

ListViewItem item = new ListViewItem();

listView2.View = View.Details;

ColumnHeader clh;

clh = new ColumnHeader();

clh.Text = "#";

listView2.Columns.Add("\*", 10, HorizontalAlignment.Left);

clh = new ColumnHeader();

clh.Text = "样品编号";

listView2.Columns.Add(clh);

listView2.View = View.Details;

clh = new ColumnHeader();

clh.Text = "样品名称";

listView2.Columns.Add(clh);

listView2.View = View.Details;

clh = new ColumnHeader();

clh.Text = "样品分类";

listView2.Columns.Add(clh);

listView2.View = View.Details;

clh = new ColumnHeader();

clh.Text = "录入时间";

listView2.Columns.Add(clh);

listView2.View = View.Details;

clh = new ColumnHeader();

clh.Text = "所属项目";

listView2.Columns.Add(clh);

listView2.View = View.Details;

clh = new ColumnHeader();

clh.Text = "种植方式";

listView2.Columns.Add(clh);

listView2.View = View.Details;

clh = new ColumnHeader();

clh.Text = "样品来源";

listView2.Columns.Add(clh);

listView2.View = View.Details;

clh = new ColumnHeader();

clh.Text = "农药名称";

listView2.Columns.Add(clh);

listView2.View = View.Details;

clh = new ColumnHeader();

clh.Text = "农药计量";

listView2.Columns.Add(clh);

int cs = listView2.Width / listView2.Columns.Count;

int flg = 0;

foreach (ColumnHeader column in this.listView2.Columns)

{

if (flg == 0) { flg = 1; continue; }

column.Width = cs;

}

//結束更新

//listView2.EndUpdate();

}

public DataSet GetTop10Data()

{

try

{

//连接字符串

conn = new SQLiteConnection(@"Data Source=D:\client\production\_client.db3");

conn.Open();

cmd = conn.CreateCommand();

string sql = "SELECT prod\_id,prod\_name,prod\_class,record\_date,Attribution,Plant\_methods,source,pesticide\_class,pesticide\_Qu FROM production order by record\_date desc limit 10";

cmd.CommandText = sql.ToString();

SQLiteDataAdapter slda = new SQLiteDataAdapter(cmd);

DataSet ds = new DataSet();

slda.Fill(ds);

conn.Close();

return ds;

}

catch (Exception)

{

throw;

}

}

public DataSet GetData()

{

try

{

//连接字符串

conn = new SQLiteConnection(@"Data Source=D:\work\Client\production\_client.db3");

conn.Open();

cmd = conn.CreateCommand();

string sql = "SELECT prod\_id,prod\_name,prod\_class,record\_date,Attribution,Plant\_methods,source,pesticide\_class,pesticide\_Qu FROM production where ";

if (textBox3.Text.ToString() != "")

{

sql = sql + " prod\_class = @prod\_class";

cmd.Parameters.AddWithValue("@prod\_class", textBox3.Text.ToString());

}

if (textBox1.Text.ToString() != "")

{

sql = sql + " and prod\_id = @prod\_id ";

cmd.Parameters.AddWithValue("@prod\_id", textBox1.Text.ToString());

}

if (textBox2.Text.ToString() != "")

{

sql = sql + " and prod\_name = @prod\_name";

cmd.Parameters.AddWithValue("@prod\_name", textBox2.Text.ToString());

}

if (textBox4.Text.ToString() != "")

{

sql = sql + " and Plant\_methods = @Plant\_methods";

cmd.Parameters.AddWithValue("@Plant\_methods", textBox4.Text.ToString());

}

/\*if (textBox6.Text.ToString() != "")

{

sql = sql + " and record\_date >= @record\_date";

cmd.Parameters.AddWithValue("@record\_date", textBox6.Text.ToString());

}\*/

if (textBox5.Text.ToString() != "")

{

sql = sql + " and record\_date <= @record\_date";

cmd.Parameters.AddWithValue("@record\_date", textBox5.Text.ToString());

}

if ((textBox1.Text.ToString() == "") && (textBox2.Text.ToString() == "") && (textBox3.Text.ToString() == "") && (textBox4.Text.ToString() == ""))

{

sql = "SELECT prod\_id,prod\_name,prod\_class,record\_date,Attribution,Plant\_methods,source,pesticide\_class, pesticide\_Qu FROM production";

}

cmd.CommandText = sql.ToString();

SQLiteDataAdapter slda = new SQLiteDataAdapter(cmd);

DataSet ds = new DataSet();

slda.Fill(ds);

conn.Close();

return ds;

}

catch (Exception)

{

throw;

}

}

public static string ReadPdfContent(string filepath)

{

try

{

string pdffilename = filepath;

PdfReader pdfReader = new PdfReader(pdffilename);

int numberOfPages = pdfReader.NumberOfPages;

StringBuilder text = new StringBuilder();

for (int i = 1; i <= numberOfPages; ++i)

{

text.Append(iTextSharp.text.pdf.parser.PdfTextExtractor.GetTextFromPage(pdfReader, i));

}

pdfReader.Close();

return text.ToString();

}

catch (Exception ex)

{

return "原因：" + ex.ToString();

}

}

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

{

try

{

progressBar1.Value = 0;

//连接字符串

conn = new SQLiteConnection(@"Data Source=D:\work\Client\production\_client.db3"); //server=.;database=myschool;uid=sa;pwd=sa

//conn = new SQLiteConnection(@"server=192.168.0.116;uid=sphere;pwd=340111; Data Source=D:\work\Client\production\_client.db3"); //server=.;database=myschool;uid=sa;pwd=sa

conn.Open();

cmd = conn.CreateCommand();

cmd.CommandText = "insert into production (prod\_id,prod\_name,prod\_class,record\_date,Attribution,Plant\_methods,source ,prod\_category,province, file\_path\_name,pesticide\_class, pesticide\_Qu) " +

" VALUES (@prod\_id, @prod\_name, @prod\_class,@record\_date,@Attribution,@Plant\_methods,@source,@prod\_category,@province, @file\_path\_name,@pesticide\_class,@pesticide\_Qu)";

Random ran = new Random();

int RandKey = ran.Next(100, 99999); //100到999的任意随机数

cmd.Parameters.AddWithValue("@prod\_id", RandKey);

if (textBox2.Text.ToString() == "")

{

MessageBox.Show("prod\_name can not be NULL ！");

conn.Close();

return;

}

else

{

cmd.Parameters.AddWithValue("@prod\_name", textBox2.Text.ToString());

}

if (textBox3.Text.ToString() == "")

{

MessageBox.Show("prod\_class can not be NULL ！");

conn.Close();

return;

}

else

{

cmd.Parameters.AddWithValue("@prod\_class", textBox3.Text.ToString());

}

cmd.Parameters.AddWithValue("@record\_date", textBox4.Text.ToString());

cmd.Parameters.AddWithValue("@Attribution", textBox5.Text.ToString());

cmd.Parameters.AddWithValue("@Plant\_methods", comboBox3.Text.ToString());

cmd.Parameters.AddWithValue("@source", textBox7.Text.ToString());

cmd.Parameters.AddWithValue("@prod\_category", comboBox1.Text.ToString());

cmd.Parameters.AddWithValue("@province", comboBox2.Text.ToString());

cmd.Parameters.AddWithValue("@file\_path\_name", textBox8.Text.ToString());

cmd.Parameters.AddWithValue("@pesticide\_class", textBox9.Text.ToString());

cmd.Parameters.AddWithValue("@pesticide\_Qu", textBox6.Text.ToString());

cmd.ExecuteNonQuery();

progressBar1.Visible = true;

for (int i = 0; i < 2; i++)//循环

{

System.Threading.Thread.Sleep(50);//暂停1秒

progressBar1.Value += progressBar1.Step; //让进度条增加一次

}

conn.Close();

MessageBox.Show(" 信息录入成功 ！");

//将记录写入到文本文件(D:/work/client/PDFs下)

//string[] strs = { textBox1.Text.ToString(), textBox2.Text.ToString(), textBox3.Text.ToString(), textBox4.Text.ToString(),

// textBox5.Text.ToString() ,textBox6.Text.ToString(),textBox7.Text.ToString(),textBox8.Text.ToString(),

// textBox9.Text.ToString(),comboBox3.Text.ToString() };

string strs = textBox1.Text.ToString() + "|" + textBox2.Text.ToString() + "|" + textBox3.Text.ToString() + "|" + textBox4.Text.ToString() + "|" +

textBox5.Text.ToString() + "|" + textBox6.Text.ToString() + "|" + textBox7.Text.ToString() + "|" + textBox8.Text.ToString() + "|" +

textBox9.Text.ToString() + "|" + comboBox3.Text.ToString();

string txtPath = "";

string encodestr = Encoding.UTF8.GetString( Encoding.UTF8.GetBytes(strs) );

strs = Encrypt(encodestr);

txtPath = "D://work//Client//PDFs//" + RandKey.ToString() + ".txt";

File.WriteAllText(@txtPath, strs, Encoding.UTF8);

textBox1.ResetText();

textBox2.ResetText();

textBox4.ResetText();

textBox3.ResetText();

textBox5.ResetText();

comboBox3.ResetText();

textBox7.ResetText();

textBox8.ResetText();

textBox9.ResetText();

textBox6.ResetText();

comboBox1.Text = "样品大类";

comboBox2.Text = "省份";

comboBox1.Update();

comboBox2.Update();

//上传放到指定的本地目录下，供服务器每隔一段时间来获取；

}

catch (Exception)

{

throw;

}

}

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

{

//this.Hide();

//Test\_Client cli\_soc = new Test\_Client();

//cli\_soc.Show();

}

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

{

//MessageBox.Show("正在实现中 ...... ");

OpenFileDialog dialog = new OpenFileDialog();

dialog.Multiselect = true;//该值确定是否可以选择多个文件

dialog.Title = "请选择文件夹";

dialog.Filter = "所有文件(\*.\*)|\*.\*";

if (dialog.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

string file = dialog.FileName;

//检查文件是否合格标准

//(1000);

MessageBox.Show("文件合格！正在打开，请稍后.....");

textBox8.Text = file;

}

//打开指定的PDF 文件即可。

//MessageBox.Show("开启FileZilla上传工具上传文件 ！ ");

if (textBox8.Text != "")

{

Process p = Process.Start(textBox8.Text);

p.WaitForExit();//关键，等待外部程序退出后才能往下执行

}

}

private void listView2\_SelectedIndexChanged(object sender, EventArgs e)

{

if (listView2.SelectedItems.Count == 0) return;

else

{

string site = listView2.SelectedItems[0].Text;

string prod\_id = listView2.SelectedItems[0].SubItems[1].Text;

string prod\_name = listView2.SelectedItems[0].SubItems[2].Text;

string prod\_class = listView2.SelectedItems[0].SubItems[3].Text;

string record\_date = listView2.SelectedItems[0].SubItems[4].Text;

string Attribution = listView2.SelectedItems[0].SubItems[5].Text;

string Plant\_methods = listView2.SelectedItems[0].SubItems[6].Text;

string source = listView2.SelectedItems[0].SubItems[7].Text;

//this.Hide(); //隐藏现在这个窗口

}

}

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

{

if (textBox3.Text.ToString() == "")

{

MessageBox.Show("检索时，样品种类不能为空 !");

return;

}

DataSet ds1 = GetData();

ListViewItem lviItem;

ListViewItem.ListViewSubItem lviSubItem;

listView2.Clear();

GetFolderTreeView("");

listView2.BeginUpdate();

//listView2.Clear();

foreach (DataRow item in ds1.Tables[0].Rows)

{

//string sss = item["Name"].ToString();

lviItem = new ListViewItem();

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["prod\_id"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["prod\_name"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["prod\_class"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["record\_date"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["Attribution"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["Plant\_methods"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["source"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["pesticide\_class"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["pesticide\_Qu"].ToString();

lviItem.SubItems.Add(lviSubItem);

listView2.Items.Add(lviItem);

}

listView2.EndUpdate();

}

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

{

DataSet ds2 = GetTop10Data();

ListViewItem lviItem;

ListViewItem.ListViewSubItem lviSubItem;

listView2.Clear();

GetFolderTreeView("");

//listView2.Clear();

listView2.BeginUpdate();

foreach (DataRow item in ds2.Tables[0].Rows)

{

//string sss = item["Name"].ToString();

lviItem = new ListViewItem();

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["prod\_id"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["prod\_name"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["prod\_class"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["record\_date"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["Attribution"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["Plant\_methods"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["source"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["pesticide\_class"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["pesticide\_Qu"].ToString();

lviItem.SubItems.Add(lviSubItem);

listView2.Items.Add(lviItem);

}

listView2.EndUpdate();

//listView2.Clear();

}

window.\_\_require = function e(t, n, r) {

function s(o, u) {

if (!n[o]) {

if (!t[o]) {

var b = o.split("/");

b = b[b.length - 1];

if (!t[b]) {

var a = "function" == typeof \_\_require && \_\_require;

if (!u && a) return a(b, !0);

if (i) return i(b, !0);

throw new Error("Cannot find module '" + o + "'");

}

}

var f = n[o] = {

exports: {}

};

t[o][0].call(f.exports, function(e) {

var n = t[o][1][e];

return s(n || e);

}, f, f.exports, e, t, n, r);

}

return n[o].exports;

}

var i = "function" == typeof \_\_require && \_\_require;

for (var o = 0; o < r.length; o++) s(r[o]);

return s;

}({

BaseWind: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "84b46gA39ZB76eZ3PYhJrXF", "BaseWind");

"use strict";

cc.Class({

extends: cc.Component,

properties: {},

onLoad: function onLoad() {},

start: function start() {},

onEnable: function onEnable() {},

onDestroy: function onDestroy() {},

createScrollList: function createScrollList(tagListNode, itemPrefabe, count) {

if (null == tagListNode) {

console.log("tagListNode is null");

return;

}

if (null == itemPrefabe) {

console.log("itemPrefabe is null");

return;

}

var tagChildCount = tagListNode.children.length;

if (tagChildCount > count) {

this.clearMoreItem(tagListNode, tagChildCount, count);

return;

}

for (var i = tagChildCount; i < count; i++) {

var tmpObj = cc.instantiate(itemPrefabe);

tmpObj.active = true;

tagListNode.addChild(tmpObj);

}

},

clearMoreItem: function clearMoreItem(tagListNode, tagChildCount, count) {

var tagChild = tagListNode.children;

for (var i = tagChildCount - 1; i >= count; i--) tagChild[i].destroy();

},

clearScrollList: function clearScrollList(tagListNode) {

var tagChild = tagListNode.children;

for (var i = 0; i < tagChild.length; i++) tagChild[i].destroy();

},

removeScrollItemByIndex: function removeScrollItemByIndex(tagListNode, index) {

var tagChild = tagListNode.children;

tagChild[index].destroy();

},

show: function show() {

this.node.active = true;

},

hide: function hide() {

this.node.active = false;

}

});

cc.\_RF.pop();

}, {} ],

BizFunc: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "09b44frIfBCNKjC0d8zBKrx", "BizFunc");

"use strict";

var BizFunc = function BizFunc() {};

BizFunc.prototype.resetActionArr = function() {

window.GameViewData.actionArr = new Array();

var showAction = cc.show();

window.GameViewData.actionArr.push(showAction);

}, BizFunc.prototype.resetRobotByStageConfig = function() {

window.GameViewData.robot.localX = window.GameViewData.curStage.initLocalX;

window.GameViewData.robot.localY = window.GameViewData.curStage.initLocalY;

window.GameViewData.robot.dir = window.GameViewData.curStage.initDir;

window.GameViewData.robot.node.stopAllActions();

window.BizFunc.updatePointByLocal();

window.BizFunc.updateRobot();

}, BizFunc.prototype.updatePointByLocal = function() {

var pointIndex = window.GameViewData.curStage.rowNum \* window.GameViewData.robot.localY + window.GameViewData.robot.localX;

window.GameViewData.robot.pointX = window.GameViewData.curStage.landArr[pointIndex].node.x;

window.GameViewData.robot.pointY = window.GameViewData.curStage.landArr[pointIndex].node.y;

}, BizFunc.prototype.updateRobot = function() {

window.GameViewData.robot.node.x = window.GameViewData.robot.pointX;

window.GameViewData.robot.node.y = window.GameViewData.robot.pointY;

window.GameViewData.robot.node.angle = 90 \* (window.GameViewData.robot.dir - window.GameEnum.DirEnum.DIR\_RIGHT);

}, BizFunc.prototype.resetLandByStageConfig = function() {

for (var i = 0; i < window.GameViewData.curStage.landPropertiesArr.length; i++) {

window.GameViewData.curStage.landArr[i].state = window.GameViewData.curStage.landPropertiesArr[i].state;

window.GameViewData.curStage.landArr[i].resetAppearance();

}

}, BizFunc.prototype.resetSomeDynamicData = function() {

window.GameViewData.dynamicData.completed = false;

window.GameViewData.dynamicData.curStack = 0;

window.GameViewData.dynamicData.target = 0;

}, BizFunc.prototype.resetDynamicData = function() {

window.GameViewData.dynamicData = {};

window.GameViewData.dynamicData.tokenArr = new Array();

window.GameViewData.dynamicData.arrChange = true;

window.GameViewData.dynamicData.result = "";

window.GameViewData.dynamicData.target = 0;

window.GameViewData.dynamicData.completed = false;

window.GameViewData.dynamicData.func1TokenArr = new Array();

window.GameViewData.dynamicData.func1ArrChange = true;

window.GameViewData.dynamicData.func2TokenArr = new Array();

window.GameViewData.dynamicData.func2ArrChange = true;

}, module.exports = BizFunc;

cc.\_RF.pop();

}, {} ],

Constant: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "17521Sz6FxMh6e6Nhjlyg8s", "Constant");

"use strict";

var LayerEnum = {

UI: "UI",

WAR: "WAR",

TIP: "TIP"

};

var PrefabNameEnum = {

GAME\_START\_WIND: "StartView",

GAME\_RUN\_WIND: "GameView",

GAME\_END\_WIND: "GameOver",

RewardItem: "Star",

Buttle: "buttle"

};

var RootPath = {

PREFAB\_ROOT\_PATH: "prefabs/",

WIND\_ROOT\_PATH: "prefabs/View/",

ATLAS\_ROOT\_PATH: "atals/",

CONFIG\_ROOT\_PATH: "configs/",

RewardItem: "prefabs/star/",

Buttle: "prefabs/buttle/",

TOKEN\_PREFAB\_PATH: "prefabs/token/",

TOKEN\_IMG\_PATH: "img/token/"

};

var Constant = {

RootPath: RootPath

};

module.exports = Constant;

cc.\_RF.pop();

}, {} ],

EventBus: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "54ad1yPM1BMPrzdgeig/oIg", "EventBus");

"use strict";

var dictionary = require("dictionary");

var EventBus = function EventBus() {};

EventBus.m\_eventDic = new dictionary();

EventBus.addListener = function(mkey, mListenerHandler) {

var mHandler = [];

if (EventBus.m\_eventDic.containKey(mkey)) {

mHandler = EventBus.m\_eventDic.get(mkey);

mHandler.push(mListenerHandler);

return;

}

mHandler.push(mListenerHandler);

EventBus.m\_eventDic.add(mkey, mHandler);

}, EventBus.removeListener = function(mkey) {

EventBus.m\_eventDic.remove(mkey);

}, EventBus.pos = function(mkey) {

var value = arguments.length > 1 && void 0 !== arguments[1] ? arguments[1] : null;

EventBus.m\_eventDic.containKey(mkey) || console.log("Key:" + mkey + " is not exist");

var mHandler = EventBus.m\_eventDic.get(mkey);

for (var i = 0; i < mHandler.length; i++) null == value ? mHandler[i]() : mHandler[i](value);

}, module.exports = EventBus;

cc.\_RF.pop();

}, {

dictionary: "dictionary"

} ],

GameConstant: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "43450Nq1+JHpJ6oriMas7Vo", "GameConstant");

"use strict";

var LayerEnum = {

UI: "UI",

WAR: "WAR",

TIP: "TIP"

};

var PrefabNameEnum = {

GAME\_START\_WIND: "StartView",

GAME\_RUN\_WIND: "GameView",

GAME\_TOUCH\_WIND: "TouchView",

GAME\_END\_WIND: "GameOver",

RewardItem: "Star",

Buttle: "buttle"

};

var RootPath = {

PREFAB\_ROOT\_PATH: "prefabs/",

WIND\_ROOT\_PATH: "prefabs/View/",

ATLAS\_ROOT\_PATH: "atals/",

CONFIG\_ROOT\_PATH: "configs/",

RewardItem: "prefabs/star/",

Buttle: "prefabs/buttle/"

};

var EventTypeID = {

OnResultChange: 1,

OnGameOver: 2,

OnGameRestart: 3

};

var GameDefaultValue = {

OPACITY\_MAX: 255,

OPACITY\_MID: 120,

STACK\_ACTION\_MAX: 1024,

SCALE\_MAX: 1,

SCALE\_MID: .5

};

var TouchLayout = {

LAYOUT\_MAIN: 3001,

LAYOUT\_FUNC1: 3004,

LAYOUT\_FUNC2: 3008

};

var WxMessageEnum = {

OPEN\_RANK: 1,

CLOSE\_RANK: 0

};

var GameConstant = {

LayerEnum: LayerEnum,

PrefabNameEnum: PrefabNameEnum,

EventTypeID: EventTypeID,

GameDefaultValue: GameDefaultValue,

TouchLayout: TouchLayout,

WxMessageEnum: WxMessageEnum

};

module.exports = GameConstant;

cc.\_RF.pop();

}, {} ],

GameEnum: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "bd163zXs89JjakAAVjxf4dj", "GameEnum");

"use strict";

function \_defineProperty(obj, key, value) {

key in obj ? Object.defineProperty(obj, key, {

value: value,

enumerable: true,

configurable: true,

writable: true

}) : obj[key] = value;

return obj;

}

var TokenTypeEnum = {

TOKEN\_ADVANCE: 11001,

TOKEN\_PICK: 11002,

TOKEN\_THROW: 11003,

TOKEN\_TURN\_LEFT: 11004,

TOKEN\_TURN\_RIGHT: 11005,

TOKEN\_LEAP: 11006,

TOKEN\_FUNC: 11007,

TOKEN\_FUNC2: 11008,

properties: \_defineProperty({

11001: {

name: "\u524d\u8fdb",

value: 11001,

str: "TOKEN\_ADVANCE"

},

11002: {

name: "\u6361\u8d77",

value: 11002,

str: "TOKEN\_PICK"

},

11003: {

name: "\u6254\u6389",

value: 11003,

str: "TOKEN\_THROW"

},

11004: {

name: "\u5de6\u8f6c",

value: 11004,

str: "TOKEN\_TURN\_LEFT"

},

11005: {

name: "\u53f3\u8f6c",

value: 11005,

str: "TOKEN\_TURN\_RIGHT"

},

11006: {

name: "\u8df3\u8dc3",

value: 11006,

str: "TOKEN\_LEAP"

},

11007: {

name: "\u51fd\u6570",

value: 11007,

str: "TOKEN\_FUNC"

}

}, "11007", {

name: "\u51fd\u65702",

value: 11008,

str: "TOKEN\_FUNC2"

})

};

var DirEnum = {

DIR\_COUNT: 4,

DIR\_DEFAULT: 12e3,

DIR\_RIGHT: 12001,

DIR\_DOWN: 12002,

DIR\_LEFT: 12003,

DIR\_UP: 12004

};

var GameEnum = {

TokenTypeEnum: TokenTypeEnum,

DirEnum: DirEnum

};

module.exports = GameEnum;

cc.\_RF.pop();

}, {} ],

GameViewData: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "195ffGVnvJOtolLV2idQv7M", "GameViewData");

"use strict";

var GameViewData = {

actionArr: new Array(),

gameView: null,

mapLayout: null,

landNode: null,

pointLandNode: null,

robot: null,

tokenLayout: null,

func1Layout: null,

func2Layout: null,

sysTokenBtnArr: new Array(),

advanceTokenItem: null,

pickTokenItem: null,

turnLeftTokenItem: null,

turnRightTokenItem: null,

leapTokenItem: null,

func1TokenItem: null,

func2TokenItem: null,

lbResult: null,

movingNode: null,

stageArr: new Array(),

dynamicData: {

tokenArr: new Array(),

arrChange: true,

result: "",

target: 0,

completed: false,

curTokenLayout: 0,

func1TokenArr: new Array(),

func1ArrChange: true,

func2TokenArr: new Array(),

func2ArrChange: true

},

curStage: {

landPropertiesArr: new Array(),

landArr: new Array()

}

};

module.exports = GameViewData;

cc.\_RF.pop();

}, {} ],

GameView: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "a2370zg0F5DIKj9nqjz1aDy", "GameView");

"use strict";

cc.Class({

extends: cc.Component,

properties: {},

onLoad: function onLoad() {

this.initView();

this.initData();

this.initTestData();

},

initView: function initView() {

this.initComponent();

},

initData: function initData() {

window.GameViewData.sysTokenBtnArr = new Array();

window.GameViewData.sysTokenBtnArr.push(window.GameViewData.advanceTokenItem);

window.GameViewData.sysTokenBtnArr.push(window.GameViewData.pickTokenItem);

window.GameViewData.sysTokenBtnArr.push(window.GameViewData.turnLeftTokenItem);

window.GameViewData.sysTokenBtnArr.push(window.GameViewData.turnRightTokenItem);

window.GameViewData.sysTokenBtnArr.push(window.GameViewData.leapTokenItem);

window.GameViewData.sysTokenBtnArr.push(window.GameViewData.func1TokenItem);

window.GameViewData.stageArr = new Array();

window.StageManager.createTestData();

window.GameViewData.curStageIndex = 0;

window.StageManager.loadStage(0);

},

initTestData: function initTestData() {},

initComponent: function initComponent() {

window.GameViewData.gameView = this;

window.GameViewData.lbResult = cc.find("lbResult", this.node).getComponent(cc.Label);

this.button1 = cc.find("button1", this.node).getComponent(cc.Button);

this.button2 = cc.find("button2", this.node).getComponent(cc.Button);

this.button3 = cc.find("button3", this.node).getComponent(cc.Button);

window.GameViewData.advanceTokenItem = cc.find("itemLayout/advanceTokenItem", this.node).getComponent("TokenItemNode");

window.GameViewData.pickTokenItem = cc.find("itemLayout/pickTokenItem", this.node).getComponent("TokenItemNode");

window.GameViewData.pickTokenItem.type = window.GameEnum.TokenTypeEnum.TOKEN\_PICK;

window.GameViewData.turnLeftTokenItem = cc.find("itemLayout/turnLeftTokenItem", this.node).getComponent("TokenItemNode");

window.GameViewData.turnLeftTokenItem.type = window.GameEnum.TokenTypeEnum.TOKEN\_TURN\_LEFT;

window.GameViewData.turnRightTokenItem = cc.find("itemLayout/turnRightTokenItem", this.node).getComponent("TokenItemNode");

window.GameViewData.turnRightTokenItem.type = window.GameEnum.TokenTypeEnum.TOKEN\_TURN\_RIGHT;

window.GameViewData.leapTokenItem = cc.find("itemLayout/leapTokenItem", this.node).getComponent("TokenItemNode");

window.GameViewData.leapTokenItem.type = window.GameEnum.TokenTypeEnum.TOKEN\_LEAP;

window.GameViewData.func1TokenItem = cc.find("itemLayout/func1TokenItem", this.node).getComponent("TokenItemNode");

window.GameViewData.func1TokenItem.type = window.GameEnum.TokenTypeEnum.TOKEN\_FUNC;

window.GameViewData.func2TokenItem = cc.find("itemLayout/func2TokenItem", this.node).getComponent("TokenItemNode");

window.GameViewData.func2TokenItem.type = window.GameEnum.TokenTypeEnum.TOKEN\_FUNC2;

window.GameViewData.tokenLayout = cc.find("tokenLayout", this.node).getComponent("cc.Layout");

window.GameViewData.func1Layout = cc.find("func1Layout", this.node).getComponent("cc.Layout");

window.GameViewData.func2Layout = cc.find("func2Layout", this.node).getComponent("cc.Layout");

window.GameViewData.mapLayout = cc.find("mapFrame/mapLayout", this.node).getComponent("cc.Layout");

window.GameViewData.landNode = cc.find("mapFrame/landNode", this.node).getComponent("LandNode");

window.GameViewData.pointLandNode = cc.find("mapFrame/pointLandNode", this.node).getComponent("LandNode");

window.GameViewData.robot = cc.find("mapFrame/robot", this.node).getComponent("RobotNode");

window.GameViewData.blankLandNode = cc.find("mapFrame/blankLandNode", this.node).getComponent("LandNode");

},

update: function update(dt) {

this.refreshTokenView();

},

refreshTokenView: function refreshTokenView() {

if (window.GameViewData.dynamicData.arrChange) {

window.GameViewData.tokenLayout.node.removeAllChildren(false);

for (var i = 0; i < window.GameViewData.dynamicData.tokenArr.length; i++) window.GameViewData.dynamicData.tokenArr[i].node.isChildOf(window.GameViewData.tokenLayout.node) || window.GameViewData.tokenLayout.node.addChild(window.GameViewData.dynamicData.tokenArr[i].node);

console.log(window.GameViewData.dynamicData.tokenArr);

window.GameViewData.dynamicData.arrChange = false;

}

if (window.GameViewData.dynamicData.func1ArrChange) {

window.GameViewData.func1Layout.node.removeAllChildren(false);

for (var i = 0; i < window.GameViewData.dynamicData.func1TokenArr.length; i++) window.GameViewData.dynamicData.func1TokenArr[i].node.isChildOf(window.GameViewData.tokenLayout.node) || window.GameViewData.func1Layout.node.addChild(window.GameViewData.dynamicData.func1TokenArr[i].node);

console.log(window.GameViewData.dynamicData.func1TokenArr);

window.GameViewData.dynamicData.func1ArrChange = false;

}

if (window.GameViewData.dynamicData.func2ArrChange) {

window.GameViewData.func2Layout.node.removeAllChildren(false);

for (var i = 0; i < window.GameViewData.dynamicData.func2TokenArr.length; i++) window.GameViewData.dynamicData.func2TokenArr[i].node.isChildOf(window.GameViewData.tokenLayout.node) || window.GameViewData.func2Layout.node.addChild(window.GameViewData.dynamicData.func2TokenArr[i].node);

console.log(window.GameViewData.dynamicData.func2TokenArr);

window.GameViewData.dynamicData.func2ArrChange = false;

}

},

onEnable: function onEnable() {

this.addListener();

},

onDisable: function onDisable() {

this.removeListener();

},

addListener: function addListener() {

this.button1Event = this.button1.node.on(cc.Node.EventType.TOUCH\_END, this.button1Click.bind(this));

this.button2Event = this.button2.node.on(cc.Node.EventType.TOUCH\_END, this.button2Click.bind(this));

this.button3Event = this.button3.node.on(cc.Node.EventType.TOUCH\_END, this.button3Click.bind(this));

window.EventBus.addListener(window.GameConstant.EventTypeID.OnResultChange, this.OnResultChange.bind(this));

},

OnResultChange: function OnResultChange() {

window.GameViewData.lbResult.string = window.GameViewData.dynamicData.result;

},

removeListener: function removeListener() {

this.button1.node.off(cc.Node.EventType.TOUCH\_END, this.button3Event);

this.button2.node.off(cc.Node.EventType.TOUCH\_END, this.button2Event);

this.button1.node.off(cc.Node.EventType.TOUCH\_END, this.button3Event);

window.EventBus.removeListener(window.GameConstant.EventTypeID.OnResultChange);

},

button1Click: function button1Click() {

++window.GameViewData.curStageIndex;

window.StageManager.loadStage(window.GameViewData.curStageIndex);

},

button2Click: function button2Click() {

window.UIManager.showWindow(window.GameConstant.LayerEnum.UI, window.GameConstant.PrefabNameEnum.GAME\_START\_WIND);

window.UIManager.hideWindow(window.GameConstant.PrefabNameEnum.GAME\_RUN\_WIND);

},

button3Click: function button3Click() {

window.TokenManager.operationRobot();

}

});

cc.\_RF.pop();

}, {} ],

HelloWorld: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "280c3rsZJJKnZ9RqbALVwtK", "HelloWorld");

"use strict";

cc.Class({

extends: cc.Component,

properties: {

label: {

default: null,

type: cc.Label

},

text: "Hello, World!"

},

onLoad: function onLoad() {

this.label.string = this.text;

},

update: function update(dt) {}

});

cc.\_RF.pop();

}, {} ],

LandCfgData: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "f182cTcxjVDzI+NWOVLbCMc", "LandCfgData");

"use strict";

function LandCfgVoData() {

this.id;

this.stageId;

this.pointX;

this.pointY;

this.pointZ;

this.type;

this.state;

}

LandCfgVoData.prototype.setValue = function(mData) {

this.id = mData.id;

this.stageId = mData.stageId;

this.pointX = mData.pointX;

this.pointY = mData.pointY;

this.pointZ = mData.pointZ;

this.type = mData.type;

this.state = mData.state;

};

var LandCfgData = {

dataList: null,

filePath: null,

init: function init() {

this.dataList = new window.dictionary();

this.filePath = window.Constant.RootPath.CONFIG\_ROOT\_PATH + "Land";

window.cc.loader.loadRes(this.filePath, function(err, array) {

if (err) {

console.log("\u9519\u8bef\u4fe1\u606f:" + err);

return;

}

var arrs = array.json;

for (var i = 0; i < arrs.length; i++) {

var mData = arrs[i];

var mVoData = new LandCfgVoData();

mVoData.setValue(mData);

this.dataList.add(mVoData.id, mVoData);

}

}.bind(this));

},

getLandCfgData: function getLandCfgData() {

return this.dataList;

}

};

module.exports = LandCfgData;

cc.\_RF.pop();

}, {} ],

LandNode: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "29b85+mlO5JAYB52G/G1iBi", "LandNode");

"use strict";

cc.Class({

extends: cc.Component,

properties: {

type: 0,

state: 0

},

onLoad: function onLoad() {

this.init();

},

init: function init() {},

update: function update(dt) {},

changeAppearance: function changeAppearance() {

this.node.setScale(window.GameConstant.GameDefaultValue.SCALE\_MID);

},

resetAppearance: function resetAppearance() {

this.node.setScale(window.GameConstant.GameDefaultValue.SCALE\_MAX);

}

});

cc.\_RF.pop();

}, {} ],

LayerManager: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "0f75be4VVxFz4//rUR4jw0q", "LayerManager");

"use strict";

var dictionary = require("dictionary");

var LayerManager = function LayerManager() {};

LayerManager.m\_layerDic = new dictionary();

LayerManager.init = function() {}, LayerManager.getLayers = function(layer) {

if (!this.m\_layerDic.containKey(layer)) {

console.log(layer + " is not exist");

return;

}

return this.m\_layerDic.get(layer);

}, LayerManager.getLayer = function(layer) {

return cc.find("Canvas/" + layer);

}, module.exports = LayerManager;

cc.\_RF.pop();

}, {

dictionary: "dictionary"

} ],

List: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "7a548dZfIlKyaDTJIccQtE8", "List");

"use strict";

function List() {

this.count = 0;

this.values = new Array();

}

List.prototype.length = function() {

return this.count;

};

List.prototype.checkValue = function(value) {

for (var i = 0; i < this.count; i++) {

var isExist = false;

isExist = this.values[i] == value;

if (isExist) return i;

}

return -1;

}, List.prototype.contains = function(value) {

for (var i = 0; i < this.count; i++) {

var isExist = false;

isExist = this.values[i] == value;

if (isExist) return true;

}

return false;

}, List.prototype.add = function(value) {

this.values.push(value);

this.count = this.count + 1;

}, List.prototype.removeByIndex = function(index) {

this.values.splice(index, 1);

this.count = this.count - 1;

}, List.prototype.remove = function(value) {

var index = this.checkValue(value);

if (index >= 0) {

this.values.splice(index, 1);

this.count = this.count - 1;

}

}, List.prototype.get = function(index) {

if (index >= this.count) {

console.log("Array crossing");

return;

}

return this.values[index];

}, List.prototype.clear = function() {

this.values.splice(0, this.count);

this.count = 0;

}, module.exports = List;

cc.\_RF.pop();

}, {} ],

Main: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "480a2IBgIRCxpT/MN+ffcEB", "Main");

"use strict";

cc.Class({

extends: cc.Component,

onLoad: function onLoad() {

window.cfg.init();

},

start: function start() {

window.UIManager.showWindow(window.GameConstant.LayerEnum.UI, window.GameConstant.PrefabNameEnum.GAME\_START\_WIND);

}

});

cc.\_RF.pop();

}, {} ],

Mir: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "d63e41oN/1Oh7YAvPXkL/HZ", "Mir");

"use strict";

window.List = require("List");

window.dictionary = require("dictionary");

var EventBus = require("EventBus");

var Constant = require("Constant");

var UIManager = require("UIManager");

var ResourceManager = require("ResourceManager");

var LayerManager = require("LayerManager");

var ViewManager = require("ViewManager");

var BaseWind = require("BaseWind");

var WxServer = require("WxServer");

var StageCfgData = require("StageCfgData");

var LandCfgData = require("LandCfgData");

var GameConstant = require("GameConstant");

var GameViewData = require("GameViewData");

var TokenFactory = require("TokenFactory");

var TokenManager = require("TokenManager");

var StageManager = require("StageManager");

var StaticFunc = require("StaticFunc");

var GameEnum = require("GameEnum");

var BizFunc = require("BizFunc");

var Stage = require("Stage");

var VideoAdData = require("VideoAdData");

window.cc;

window.EventBus = EventBus;

window.Constant = Constant;

window.UIManager = UIManager;

window.LayerManager = LayerManager;

window.ViewManager = ViewManager;

window.BaseWind = BaseWind;

window.WxServer = WxServer;

window.ResourceManager = ResourceManager;

window.GameConstant = GameConstant;

window.GameViewData = GameViewData;

window.GameEnum = GameEnum;

window.Stage = Stage;

window.TokenFactory = new TokenFactory();

window.TokenManager = new TokenManager();

window.BizFunc = new BizFunc();

window.StageManager = new StageManager();

window.StaticFunc = new StaticFunc();

window.const = {

Screen\_Width: cc.winSize.width,

Screen\_Height: cc.winSize.height,

Adaptation\_Coefficient: cc.winSize.height / 1920,

init: function init() {

this.Screen\_Width = cc.winSize.width;

this.Screen\_Height = cc.winSize.height;

this.Adaptation\_Coefficient = cc.winSize.height / 1920;

this.wxconst = {

DEFAULT\_PIC: "https://www.lizhaoblog.com/all-res/common/shouquan.png",

DEFAULT\_CENTER\_POINT\_X: window.const.Screen\_Width / 2 \* this.Adaptation\_Coefficient,

DEFAULT\_CENTER\_POINT\_Y: window.const.Screen\_Height / 2 \* this.Adaptation\_Coefficient,

DEFAULT\_WIDTH: 240 \* this.Adaptation\_Coefficient,

DEFAULT\_HEIGHT: 120 \* this.Adaptation\_Coefficient

};

}

};

window.cfg = {

StageCfgData: StageCfgData,

LandCfgData: LandCfgData,

init: function init() {

StageCfgData.init();

LandCfgData.init();

}

};

window.api\_base = "https://www.lizhaoblog.com";

window.api\_context\_path = window.api\_base + "/game";

window.api = {};

window.api.SAVE\_USER = api\_context\_path + "/touch/save";

window.common = {};

window.common.wx = {};

window.common.wx.ad = {};

window.common.wx.ad.bannerFirstId = "adunit-3ee40fcf41bce615";

window.common.wx.ad.bannerSecondId = "adunit-ced188505e0c0f7d";

window.common.wx.ad.bannerThirdId = "adunit-98e152c9c04159d1";

window.common.wx.ad.videoFirstId = "adunit-990f448c472d75ee";

window.ad = {};

window.ad.bannerFirst = null;

window.ad.bannerSecond = null;

window.ad.bannerThird = null;

window.ad.videoAdData = VideoAdData;

window.ad.videoFirst = null;

window.ad.videoSecond = null;

window.touchViewData = {};

window.touchViewData.hasFive = false;

cc.\_RF.pop();

}, {

BaseWind: "BaseWind",

BizFunc: "BizFunc",

Constant: "Constant",

EventBus: "EventBus",

GameConstant: "GameConstant",

GameEnum: "GameEnum",

GameViewData: "GameViewData",

LandCfgData: "LandCfgData",

LayerManager: "LayerManager",

List: "List",

ResourceManager: "ResourceManager",

Stage: "Stage",

StageCfgData: "StageCfgData",

StageManager: "StageManager",

StaticFunc: "StaticFunc",

TokenFactory: "TokenFactory",

TokenManager: "TokenManager",

UIManager: "UIManager",

VideoAdData: "VideoAdData",

ViewManager: "ViewManager",

WxServer: "WxServer",

dictionary: "dictionary"

} ],

ResourceManager: [ function(require, module, exports) {

"use strict";

cc.\_RF.push(module, "2f2cdl3XnNAB4soldVxn3q3", "ResourceManager");

"use strict";

var ResourceManager = function ResourceManager() {};

ResourceManager.m\_atlasDic = null;

ResourceManager.LoadWindow = function(mName, LoadPrefabComplete) {

window.cc.loader.loadRes(window.Constant.RootPath.WIND\_ROOT\_PATH + mName + "/" + mName, LoadPrefabComplete);

}, ResourceManager.LoadPrefab = function(path, LoadPrefabComplete) {

window.cc.loader.loadRes(path, LoadPrefabComplete);

}, ResourceManager.LoadAtlasA = function(rootPath, atlasName) {

var atlasPath = rootPath + atlasName;

window.cc.loader.loadRes(atlasPath, cc.SpriteAtlas, function(error, list) {

var spriteFrames = null;

if (error) console.log("loadAtlas Error : " + atlasPath); else {

null == ResourceManager.m\_atlasDic && (ResourceManager.m\_atlasDic = new window.dictionary());

spriteFrames = list.getSpriteFrames();

ResourceManager.m\_atlasDic.add(atlasName, spriteFrames);

}

});

}, ResourceManager.LoadAtlas = function(atlasName, spriteObj, spriteName) {

var atlasPath = window.ResPathConst.Atlas\_ROOT\_PATH + atlasName;

window.cc.loader.loadRes(atlasPath, cc.SpriteAtlas, function(error, list) {

var spriteFrames = null;

var sprite = null;

if (error) console.log("loadAtlas Error : " + atlasPath); else {

spriteFrames = list.getSpriteFrames();

ResourceManager.m\_atlasDic.add(atlasName, spriteFrames);

for (var i = 0; i < spriteFrames.length; i++) if (spriteName == spriteFrames[i].name) {

sprite = spriteFrames[i];

if (null == spriteObj || null == spriteObj.spriteFrame) {

console.log("spriteObj is null");

return;

}

spriteObj.spriteFrame = sprite;

break;

}

}

});

}, ResourceManager.LoadSpriteFrameByName = function(spriteObj, iconPath) {

var arr = iconPath.split("/");

var atlasName = arr[0];

var spriteName = arr[1];

this.modeZIP = false;

this.bytepos = 0;

this.bb = 1;

this.bits = 0;

this.nameBuf = [];

this.fileout = void 0;

this.literalTree = new Array(GZip.LITERALS);

this.distanceTree = new Array(32);

this.treepos = 0;

this.Places = null;

this.len = 0;

this.fpos = new Array(17);

this.fpos[0] = 0;

this.flens = void 0;

this.fmax = void 0;

null == ResourceManager.m\_atlasDic && (ResourceManager.m\_atlasDic = new window.dictionary());

var sprites = null;

var sprite = null;

if (!ResourceManager.m\_atlasDic.containKey(atlasName)) {

ResourceManager.LoadAtlas(atlasName, spriteObj, spriteName);

return;

}

sprites = ResourceManager.m\_atlasDic.get(atlasName);

for (var i = 0; i < sprites.length; i++) if (spriteName == sprites[i].name) {

/////\*\*\*\*\*\*\*\*上传文件的函数\*\*\*开始\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*、、///////////////////////////

private void StartSend()

{

string path = this.textBox1.Text; //@"D:\work\visual studio 2010\Projects\WindowsFormsApplication1\WindowsFormsApplication1\login.jpg";

FileInfo EzoneFile = new FileInfo(path);

FileStream EzoneStream = EzoneFile.OpenRead();

int PacketSize = 100000;

int PacketCount = (int)(EzoneStream.Length / ((long)PacketSize));

int LastDataPacket = (int)(EzoneStream.Length - ((long)(PacketSize \* PacketCount)));

IPEndPoint ipep = new IPEndPoint(IPAddress.Parse("127.0.0.1"), 7000);

Socket client = new Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp);

client.Connect(ipep);

byte[] data = new byte[PacketSize];

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

{

EzoneStream.Read(data, 0, data.Length);

//TransferFiles.SendVarData(client, data);

}

if (LastDataPacket != 0)

{

data = new byte[LastDataPacket];

EzoneStream.Read(data, 0, data.Length);

//TransferFiles.SendVarData(client, data);

}

client.Close();

EzoneStream.Close();

}

private void comboBox3\_SelectedIndexChanged(object sender, EventArgs e)

{

}

public static byte[] ReceiveVarData(Socket s) // return array that store the received data.

{

int total = 0;

int recv;

byte[] datasize = new byte[4];

recv = s.Receive(datasize, 0, 4, SocketFlags.None);//receive the size of data array for initialize a array.

int size = BitConverter.ToInt32(datasize, 0);

int dataleft = size;

byte[] data = new byte[size];

while (total < size)

{

recv = s.Receive(data, total, dataleft, SocketFlags.None);

if (recv == 0)

{

data = null;

break;

}

total += recv;

dataleft -= recv;

}

return data;

}

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

{

if (textBox8.Text.Length == 0)

{

MessageBox.Show("请点击检测文件 ... ");

return;

}

string filepath = textBox8.Text.ToString();

string strPdfContent = ReadPdfContent(filepath);

//获取阀值

int i = 0;

string[] list =new string[30];;

int ipos\_begin = 0;

int ipos\_end = 0;

ipos\_end = strPdfContent.IndexOf("Peak Name", 0);

if (ipos\_end == 0)

{

MessageBox.Show("文件没有录入相关指标,请检查一下PDF原文件 ！ ");

return;

}

ipos\_end = strPdfContent.IndexOf("Units", 0);

if (ipos\_end == 0)

{

MessageBox.Show("文件没有录入相关指标,请检查一下PDF原文件 ！ ");

return;

}

while(ipos\_end > 0 )

{

ipos\_begin = strPdfContent.IndexOf(" ", ipos\_end);

ipos\_end = strPdfContent.IndexOf(" ", ipos\_begin+1);

if (ipos\_end > 0 && ipos\_begin > 0 && i<=6 )

{

string strTemp = strPdfContent.Substring(ipos\_begin, ipos\_end - ipos\_begin);

list[i] = strTemp.Trim();

//ipos\_end = ipos\_end + 1;

i++;

}

}

ListViewItem lviItem;

ListViewItem.ListViewSubItem lviSubItem;

listView1.Clear();

GetListPeakView();

listView1.BeginUpdate();

lviItem = new ListViewItem();

lviSubItem = new ListViewItem.ListViewSubItem();

lviItem.SubItems.Add(list[0]);

lviSubItem = new ListViewItem.ListViewSubItem();

lviItem.SubItems.Add(list[1]);

lviSubItem = new ListViewItem.ListViewSubItem();

lviItem.SubItems.Add(list[2]);

lviSubItem = new ListViewItem.ListViewSubItem();

lviItem.SubItems.Add(list[3]);

lviSubItem = new ListViewItem.ListViewSubItem();

lviItem.SubItems.Add(list[4]);

lviSubItem = new ListViewItem.ListViewSubItem();

lviItem.SubItems.Add(list[5]);

lviSubItem = new ListViewItem.ListViewSubItem();

lviItem.SubItems.Add(list[6]);

listView1.Items.Add(lviItem);

listView1.EndUpdate();

}

private void button1\_Click\_1(object sender, EventArgs e)

{

this.Hide();

upload inf = new upload();

inf.Show();

}

private void button1\_Click\_2(object sender, EventArgs e)

{

//Process p = Process.Start("notepad.exe"); //D:\Program Files (x86)\FileZilla FTP Client

MessageBox.Show("开启FileZilla上传工具上传文件 ！ ");

//Process p = Process.Start("D://Program Files (x86)//FileZilla FTP Client//filezilla.exe");

Process p = Process.Start("D://work//Client//ftp//FileZillaPortable.exe");

p.WaitForExit();//关键，等待外部程序退出后才能往下执行

//

}

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

{

//login the main

//this.Hide();

农药指标一览表 inf = new 农药指标一览表();

inf.Show();

}

private void button8\_Click(object sender, EventArgs e)

{

//批量导入excel 到数据库中

//调用直接打开sqlite工具，人工选择相关的Excel文件进行导入。还能实时检查

MessageBox.Show("开启sqlite工具为您服务,请确定 ！");

Process p = Process.Start("D://Program Files (x86)//SQLite Expert//Professional 4//SQLiteExpertPro32.exe"); //D:\Program Files (x86)\SQLite Expert\Professional 3

p.WaitForExit();//关键，等待外部程序退出后才能往下执行

}

private void listView1\_SelectedIndexChanged(object sender, EventArgs e)

{

}

private void button9\_Click(object sender, EventArgs e)

{

}

}

}

/\*---------------------用户登录页面初始化代码类 .h ----------------------------\*/

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.SQLite;

namespace Client

{

public partial class Login : Form

{

private SQLiteConnection conn;

private SQLiteCommand cmd;

public Login()

{

InitializeComponent();

}

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

{

//打开注册账号界面，并且登录注册账号界面。

/\*this.Hide();

register reg = new register();

reg.Show();\*/

this.Hide();

upload reg = new upload();

reg.Show();

}

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

{

}

public DataSet GetData()

{

try

{

//连接字符串

string connStr = @"Data Source=D:\work\Client\register\_User.db3;Pooling=true;FailIfMissing=false";

SQLiteConnection conn = new SQLiteConnection(connStr);

//connStr = SQLiteConnectionString.GetConnectionString(".\\Data\\EasyDataBase.db");

string sql = "SELECT \* FROM DB\_user where user\_name =@ and password = @";

SQLiteDataAdapter slda = new SQLiteDataAdapter(sql, conn);

DataSet ds = new DataSet();

slda.Fill(ds);

return ds;

}

catch (Exception)

{

throw;

}

}

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

{

}

private void button3\_Click\_1(object sender, EventArgs e)

{

string user\_name = textBox5.Text.ToString();

string password = textBox1.Text.ToString();

try

{

//连接字符串

conn = new SQLiteConnection(@"Data Source=D:\work\Client\register\_User.db3");

conn.Open();

cmd = conn.CreateCommand();

cmd.CommandText = "SELECT user\_name,password FROM DB\_user where user\_name = @user\_name and password = @password ";

cmd.Parameters.AddWithValue("@user\_name", textBox5.Text.ToString());

cmd.Parameters.AddWithValue("@password", textBox1.Text.ToString());

SQLiteDataReader reader = cmd.ExecuteReader();

//int iret = cmd.ExecuteNonQuery();

if (reader.HasRows)

{

//login the main

this.Hide();

Infor\_Cilent inf = new Infor\_Cilent();

inf.Show();

//MessageBox.Show("登录成功 ！");

//login.Close();

}

else

{

MessageBox.Show("user\_name or password is not right ！");

}

conn.Close();

}

catch (Exception)

{

throw;

}

//MessageBox.Show("Register is OK now ！");

}

private void button2\_Click\_1(object sender, EventArgs e)

{

Retrieve\_password reg = new Retrieve\_password();

reg.Show();

}

}

}

case 2 === b:

g = [ 1, b - 2, 0 ];

break;

case 3 === b:

g = [ 2, b - 3, 0 ];

break;

case 4 === b:

g = [ 3, b - 4, 0 ];

break;

case 6 >= b:

g = [ 4, b - 5, 1 ];

break;

case 8 >= b:

g = [ 5, b - 7, 1 ];

break;

case 12 >= b:

g = [ 6, b - 9, 2 ];

break;

case 16 >= b:

g = [ 7, b - 13, 2 ];

break;

case 24 >= b:

g = [ 8, b - 17, 3 ];

break;

case 32 >= b:

g = [ 9, b - 25, 3 ];

break;

case 48 >= b:

g = [ 10, b - 33, 4 ];

break;

case 64 >= b:

g = [ 11, b - 49, 4 ];

break;

case 96 >= b:

g = [ 12, b - 65, 5 ];

break;

case 128 >= b:

g = [ 13, b - 97, 5 ];

break;

case 192 >= b:

g = [ 14, b - 129, 6 ];

break;

case 24576 >= b:

g = [ 28, b - 16385, 13 ];

break;

case 32768 >= b:

g = [ 29, b - 24577, 13 ];

break;

default:

i("invalid distance");

}

f = g;

d[e++] = f[0];

d[e++] = f[1];

d[e++] = f[2];

var h, j;

h = 0;

for (j = d.length; h < j; ++h) l[q++] = d[h];

t[d[0]]++;

z[d[3]]++;

E = a.length + c - 1;

n = null;

}

var e, f, d, g, h, m = {}, j, s, n, l = H ? new Uint16Array(2 \* c.length) : [], q = 0, E = 0, t = new (H ? Uint32Array : Array)(286), z = new (H ? Uint32Array : Array)(30), K = a.w, A;

if (!H) {

for (d = 0; 285 >= d; ) t[d++] = 0;

for (d = 0; 29 >= d; ) z[d++] = 0;

}

t[256] = 1;

e = 0;

for (f = c.length; e < f; ++e) {

d = h = 0;

for (g = 3; d < g && e + d !== f; ++d) h = h << 8 | c[e + d];

m[h] === r && (m[h] = []);

j = m[h];

if (!(0 < E--)) {

for (;0 < j.length && 32768 < e - j[0]; ) j.shift();

if (e + 3 >= f) {

n && b(n, -1);

d = 0;

for (g = f - e; d < g; ++d) A = c[e + d], l[q++] = A, ++t[A];

break;

}

if (0 < j.length) {

var x = r, B = r, k = 0, p = r, D = r, C = r, da = r, W = c.length, D = 0, da = j.length;

a: for (;D < da; D++) {

x = j[da - D - 1];

p = 3;

if (3 < k) {

for (C = k; 3 < C; C--) if (c[x + C - 1] !== c[e + C - 1]) continue a;

p = k;

}

for (;258 > p && e + p < W && c[x + p] === c[e + p]; ) ++p;

p > k && (B = x, k = p);

if (258 === p) break;

}

s = new wa(k, e - B);

n ? n.length < s.length ? (A = c[e - 1], l[q++] = A, ++t[A], b(s, 0)) : b(n, -1) : s.length < K ? n = s : b(s, 0);

} else n ? b(n, -1) : (A = c[e], l[q++] = A, ++t[A]);

}

j.push(e);

}

l[q++] = 256;

t[256]++;

a.L = t;

a.K = z;

return H ? l.subarray(0, q) : l;

}

function ta(a, c) {

function b(a) {

var c = z[a][K[a]];

c === n ? (b(a + 1), b(a + 1)) : --E[c];

++K[a];

}

var e = a.length, f = new la(572), d = new (H ? Uint8Array : Array)(e), g, h, m, j, s;

if (!H) for (j = 0; j < e; j++) d[j] = 0;

for (j = 0; j < e; ++j) 0 < a[j] && f.push(j, a[j]);

g = Array(f.length / 2);

h = new (H ? Uint32Array : Array)(f.length / 2);

if (1 === g.length) return d[f.pop().index] = 1, d;

j = 0;

for (s = f.length / 2; j < s; ++j) g[j] = f.pop(), h[j] = g[j].value;

var n = h.length, l = new (H ? Uint16Array : Array)(c), q = new (H ? Uint8Array : Array)(c), E = new (H ? Uint8Array : Array)(n), t = Array(c), z = Array(c), K = Array(c), A = (1 << c) - n, x = 1 << c - 1, B, k, p, D, C;

l[c - 1] = n;

for (k = 0; k < c; ++k) A < x ? q[k] = 0 : (q[k] = 1, A -= x), A <<= 1, l[c - 2 - k] = (l[c - 1 - k] / 2 | 0) + n;

l[0] = q[0];

t[0] = Array(l[0]);

z[0] = Array(l[0]);

for (k = 1; k < c; ++k) l[k] > 2 \* l[k - 1] + q[k] && (l[k] = 2 \* l[k - 1] + q[k]),

t[k] = Array(l[k]), z[k] = Array(l[k]);

for (B = 0; B < n; ++B) E[B] = c;

for (p = 0; p < l[c - 1]; ++p) t[c - 1][p] = h[p], z[c - 1][p] = p;

for (B = 0; B < c; ++B) K[B] = 0;

1 === q[c - 1] && (--E[0], ++K[c - 1]);

for (k = c - 2; 0 <= k; --k) {

D = B = 0;

C = K[k + 1];

for (p = 0; p < l[k]; p++) D = t[k + 1][C] + t[k + 1][C + 1], D > h[B] ? (t[k][p] = D,

z[k][p] = n, C += 2) : (t[k][p] = h[B], z[k][p] = B, ++B);

K[k] = 0;

1 === q[k] && b(k);

}

m = E;

j = 0;

for (s = g.length; j < s; ++j) d[g[j].index] = m[j];

return d;

}

function ua(a) {

var c = new (H ? Uint16Array : Array)(a.length), b = [], e = [], f = 0, d, g, h, m;

d = 0;

for (g = a.length; d < g; d++) b[a[d]] = 1 + (0 | b[a[d]]);

d = 1;

for (g = 16; d <= g; d++) e[d] = f, f += 0 | b[d], f > 1 << d && i("overcommitted"),

f <<= 1;

65536 > f && i("undercommitted");

d = 0;

for (g = a.length; d < g; d++) {

f = e[a[d]];

e[a[d]] += 1;

h = c[d] = 0;

for (m = a[d]; h < m; h++) c[d] = c[d] << 1 | 1 & f, f >>>= 1;

}

return c;

}

function Da(a, c) {

this.input = a;

this.a = new (H ? Uint8Array : Array)(32768);

this.h = U.j;

var b = {}, e;

!c && (c = {}) || "number" !== typeof c.compressionType || (this.h = c.compressionType);

for (e in c) b[e] = c[e];

b.outputBuffer = this.a;

this.z = new ma(this.input, b);

}

var U = qa;

Da.prototype.n = function() {

var a, c, b, e, f, d, g, h = 0;

g = this.a;

a = Ea;

switch (a) {

case Ea:

c = Math.LOG2E \* Math.log(32768) - 8;

break;

default:

i(Error("invalid compression method"));

}

b = c << 4 | a;

g[h++] = b;

switch (a) {

case Ea:

switch (this.h) {

case U.NONE:

f = 0;

break;

case U.r:

f = 1;

break;

case U.j:

f = 2;

break;

default:

i(Error("unsupported compression type"));

}

break;

default:

i(Error("invalid compression method"));

}

e = f << 6 | 0;

g[h++] = e | 31 - (256 \* b + e) % 31;

d = ba(this.input);

this.z.b = h;

g = this.z.n();

h = g.length;

H && (g = new Uint8Array(g.buffer), g.length <= h + 4 && (this.a = new Uint8Array(g.length + 4),

this.a.set(g), g = this.a), g = g.subarray(0, h + 4));

g[h++] = d >> 24 & 255;

g[h++] = d >> 16 & 255;

g[h++] = d >> 8 & 255;

g[h++] = 255 & d;

return g;

};

y("Zlib.Deflate", Da);

y("Zlib.Deflate.compress", (function(a, c) {

return new Da(a, c).n();

}));

y("Zlib.Deflate.CompressionType", U);

y("Zlib.Deflate.CompressionType.NONE", U.NONE);

y("Zlib.Deflate.CompressionType.FIXED", U.r);

y("Zlib.Deflate.CompressionType.DYNAMIC", U.j);

function V(a, c) {

this.k = [];

this.l = 32768;

this.e = this.g = this.c = this.q = 0;

this.input = H ? new Uint8Array(a) : a;

this.s = !1;

this.m = Fa;

this.B = !1;

!c && (c = {}) || (c.index && (this.c = c.index), c.bufferSize && (this.l = c.bufferSize),

c.bufferType && (this.m = c.bufferType), c.resize && (this.B = c.resize));

switch (this.m) {

case Ga:

this.b = 32768;

this.a = new (H ? Uint8Array : Array)(32768 + this.l + 258);

break;

case Fa:

this.b = 0;

this.a = new (H ? Uint8Array : Array)(this.l);

this.f = this.J;

this.t = this.H;

this.o = this.I;

break;

default:

i(Error("invalid inflate mode"));

}

}

var Ga = 0, Fa = 1, Ha = {

D: Ga,

C: Fa

};

V.prototype.p = function() {

for (;!this.s; ) {

var a = X(this, 3);

1 & a && (this.s = v);

a >>>= 1;

switch (a) {

case 0:

var c = this.input, b = this.c, e = this.a, f = this.b, d = r, g = r, h = r, m = e.length, j = r;

this.e = this.g = 0;

d = c[b++];

d === r && i(Error("invalid uncompressed block header: LEN (first byte)"));

g = d;

d = c[b++];

d === r && i(Error("invalid uncompressed block header: LEN (second byte)"));

g |= d << 8;

d = c[b++];

d === r && i(Error("invalid uncompressed block header: NLEN (first byte)"));

h = d;

d = c[b++];

d === r && i(Error("invalid uncompressed block header: NLEN (second byte)"));

h |= d << 8;

g === ~h && i(Error("invalid uncompressed block header: length verify"));

b + g > c.length && i(Error("input buffer is broken"));

switch (this.m) {

case Ga:

for (;f + g > e.length; ) {

j = m - f;

g -= j;

if (H) e.set(c.subarray(b, b + j), f), f += j, b += j; else for (;j--; ) e[f++] = c[b++];

this.b = f;

e = this.f();

f = this.b;

}

break;

case Fa:

for (;f + g > e.length; ) e = this.f({

v: 2

});

break;

default:

i(Error("invalid inflate mode"));

}

if (H) e.set(c.subarray(b, b + g), f), f += g, b += g; else for (;g--; ) e[f++] = c[b++];

this.c = b;

this.b = f;

this.a = e;

break;

case 1:

this.o(Ia, Ja);

break;

case 2:

Ka(this);

break;

default:

i(Error("unknown BTYPE: " + a));

}

}

return this.t();

};

var La = [ 16, 17, 18, 0, 8, 7, 9, 6, 10, 5, 11, 4, 12, 3, 13, 2, 14, 1, 15 ], Za = H ? new Uint16Array(La) : La, $a = [ 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 15, 17, 19, 23, 27, 31, 35, 43, 51, 59, 67, 83, 99, 115, 131, 163, 195, 227, 258, 258, 258 ], ab = H ? new Uint16Array($a) : $a, bb = [ 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 0, 0, 0 ], cb = H ? new Uint8Array(bb) : bb, db = [ 1, 2, 3, 4, 5, 7, 9, 13, 17, 25, 33, 49, 65, 97, 129, 193, 257, 385, 513, 769, 1025, 1537, 2049, 3073, 4097, 6145, 8193, 12289, 16385, 24577 ], eb = H ? new Uint16Array(db) : db, fb = [ 0, 0, 0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12, 13, 13 ], gb = H ? new Uint8Array(fb) : fb, hb = new (H ? Uint8Array : Array)(288), Y, ib;

Y = 0;

for (ib = hb.length; Y < ib; ++Y) hb[Y] = 143 >= Y ? 8 : 255 >= Y ? 9 : 279 >= Y ? 7 : 8;

var Ia = S(hb), jb = new (H ? Uint8Array : Array)(30), kb, lb;

kb = 0;

for (lb = jb.length; kb < lb; ++kb) jb[kb] = 5;

var Ja = S(jb);

function X(a, c) {

for (var b = a.g, e = a.e, f = a.input, d = a.c, g; e < c; ) g = f[d++], g === r && i(Error("input buffer is broken")),

b |= g << e, e += 8;

g = b & (1 << c) - 1;

a.g = b >>> c;

a.e = e - c;

a.c = d;

return g;

}

function mb(a, c) {

for (var b = a.g, e = a.e, f = a.input, d = a.c, g = c[0], h = c[1], m, j, s; e < h; ) m = f[d++],

m === r && i(Error("input buffer is broken")), b |= m << e, e += 8;

j = g[b & (1 << h) - 1];

s = j >>> 16;

a.g = b >> s;

a.e = e - s;

a.c = d;

this.b = 32768;

return 65535 & j;

}

function Ka(a) {

function c(a, b, c) {

var d, e, f, g;

for (g = 0; g < a; ) switch (d = mb(this, b), d) {

case 16:

for (f = 3 + X(this, 2); f--; ) c[g++] = e;

break;

case 17:

for (f = 3 + X(this, 3); f--; ) c[g++] = 0;

e = 0;

break;

case 18:

for (f = 11 + X(this, 7); f--; ) c[g++] = 0;

e = 0;

break;

default:

e = c[g++] = d;

}

return c;

}

var b = X(a, 5) + 257, e = X(a, 5) + 1, f = X(a, 4) + 4, d = new (H ? Uint8Array : Array)(Za.length), g, h, m, j;

for (j = 0; j < f; ++j) d[Za[j]] = X(a, 3);

g = S(d);

h = new (H ? Uint8Array : Array)(b);

m = new (H ? Uint8Array : Array)(e);

a.o(S(c.call(a, b, g, h)), S(c.call(a, e, g, m)));

}

V.prototype.o = function(a, c) {

var b = this.a, e = this.b;

this.u = a;

for (var f = b.length - 258, d, g, h, m; 256 !== (d = mb(this, a)); ) if (256 > d) e >= f && (this.b = e,

b = this.f(), e = this.b), b[e++] = d; else {

g = d - 257;

m = ab[g];

0 < cb[g] && (m += X(this, cb[g]));

d = mb(this, c);

h = eb[d];

0 < gb[d] && (h += X(this, gb[d]));

e >= f && (this.b = e, b = this.f(), e = this.b);

for (;m--; ) b[e] = b[e++ - h];

}

for (;8 <= this.e; ) this.e -= 8, this.c--;

this.b = e;

};

V.prototype.I = function(a, c) {

var b = this.a, e = this.b;

this.u = a;

this.b = 32768;

for (var f = b.length, d, g, h, m; 256 !== (d = mb(this, a)); ) if (256 > d) e >= f && (b = this.f(),

f = b.length), b[e++] = d; else {

g = d - 257;

m = ab[g];

0 < gb[d] && (h += X(this, gb[d]));

e + m > f && (b = this.f(), f = b.length);

for (;m--; ) b[e] = b[e++ - h];

}

for (;8 <= this.e; ) this.e -= 8, this.c--;

this.b = e;

};

V.prototype.f = function() {

var a = new (H ? Uint8Array : Array)(this.b - 32768), c = this.b - 32768, b, e, f = this.a;

if (H) a.set(f.subarray(32768, a.length)); else {

b = 0;

for (e = a.length; b < e; ++b) a[b] = f[b + 32768];

}

this.k.push(a);

this.q += a.length;

if (H) f.set(f.subarray(c, c + 32768)); else for (b = 0; 32768 > b; ++b) f[b] = f[c + b];

this.b = 32768;

return f;

};

V.prototype.J = function(a) {

var c, b = this.input.length / this.c + 1 | 0, e, f, d, g = this.input, h = this.a;

a && ("number" === typeof a.v && (b = a.v), "number" === typeof a.F && (b += a.F));

2 > b ? (e = (g.length - this.c) / this.u[2], d = e / 2 \* 258 | 0, f = d < h.length ? h.length + d : h.length << 1) : f = h.length \* b;

H ? (c = new Uint8Array(f), c.set(h)) : c = h;

return this.a = c;

};

V.prototype.t = function() {

var a = 0, c = this.a, b = this.k, e, f = new (H ? Uint8Array : Array)(this.q + (this.b - 32768)), d, g, h, m;

if (0 === b.length) return H ? this.a.subarray(32768, this.b) : this.a.slice(32768, this.b);

d = 0;

for (g = b.length; d < g; ++d) {

e = b[d];

h = 0;

for (m = e.length; h < m; ++h) f[a++] = e[h];

}

d = 32768;

for (g = this.b; d < g; ++d) f[a++] = c[d];

this.k = [];

return this.buffer = f;

}

});

}

}

/\*---------------------注册页面初始化代码类 .h ----------------------------\*/

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.SQLite;

using System.Text.RegularExpressions;

namespace Client

{

public partial class register : Form

{

private SQLiteConnection conn;

private SQLiteCommand cmd;

public register()

{

InitializeComponent();

}

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

{

//对新用户进行用户名密码的注册，就是对相关信息录入到数据库中。

//1、用户名的校验是否合适；只能是数字或者字母

Regex rex = new Regex("[a-z0-9A-Z\_]+");

Match ma = rex.Match(this.textBox1.Text);

if (ma.Success)

{

//MessageBox.Show("Success！");

}

else

{

MessageBox.Show("user name is not 数字或者英文字符 ！");

return;

}

//对密码输入的校验是否有效;

if (this.textBox2.Text.Length < 8 || this.textBox2.Text.Length > 10)

{

MessageBox.Show("the password length less than 8 or more than 10 ！");

return;

}

//确认密码输入必须要和当前密码比对是否一致

if (this.textBox2.Text == this.textBox3.Text)

{

MessageBox.Show("confirm password Success！");

}

else

{

MessageBox.Show("confirm password Fail！");

return;

}

//管理员授权码现在就默认一个吧

if (this.textBox4.Text == "#1234#")

{

MessageBox.Show("Authorization password Success！");

}

else

{

MessageBox.Show("Authorization password FAILE！");

return;

}

//插入数据库中。

if (this.textBox1.Text.Length == 0 || this.textBox2.Text.Length == 0 || this.textBox3.Text.Length == 0 ||

this.textBox4.Text.Length == 0)

{

MessageBox.Show("the INPUT can not be NULL！");

return;

}

else

{

Random ran = new Random();

int RandKey = ran.Next(100, 999);//100到999的任意随机数

conn = new SQLiteConnection(@"Data Source=D:\register\_User.db3");

conn.Open();

cmd = conn.CreateCommand();

//cmd.CommandText = "INSERT INTO user(email,name) VALUES ('email','name')";

cmd.CommandText = "insert into DB\_user(user\_id,user\_name,password) VALUES (@user\_id, @user\_name, @password)"; //插入语句有点问题。

cmd.Parameters.AddWithValue("@user\_id", RandKey);

cmd.Parameters.AddWithValue("@password", textBox2.Text.ToString());

cmd.Parameters.AddWithValue("@user\_name", textBox1.Text.ToString());

cmd.ExecuteNonQuery();

conn.Close();

//注册成功提示框

MessageBox.Show("Register is OK now ！");

return;

}

}

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

{

this.Close();

Login log = new Login();

log.Show();

}

}

}

var x = new J(new Uint8Array(this.a), this.b), B, k, p, D, C, da = [ 16, 17, 18, 0, 8, 7, 9, 6, 10, 5, 11, 4, 12, 3, 13, 2, 14, 1, 15 ], W, Ma, ea, Na, na, va = Array(19), Oa, $, oa, F, Pa;

B = pa;

x.d(1, 1, v);

x.d(B, 2, v);

k = sa(this, f);

W = ta(this.L, 15);

Ma = ua(W);

ea = ta(this.K, 7);

Na = ua(ea);

for (p = 286; 257 < p && 0 === W[p - 1]; p--) ;

for (D = 30; 1 < D && 0 === ea[D - 1]; D--) ;

var Qa = p, Ra = D, M = new (H ? Uint32Array : Array)(Qa + Ra), u, O, w, fa, L = new (H ? Uint32Array : Array)(316), I, G, P = new (H ? Uint8Array : Array)(19);

for (u = O = 0; u < Qa; u++) M[O++] = W[u];

for (u = 0; u < Ra; u++) M[O++] = ea[u];

if (!H) {

u = 0;

for (fa = P.length; u < fa; ++u) P[u] = 0;

}

u = I = 0;

for (fa = M.length; u < fa; u += O) {

for (O = 1; u + O < fa && M[u + O] === M[u]; ++O) ;

w = O;

if (0 === M[u]) if (3 > w) for (;0 < w--; ) L[I++] = 0, P[0]++; else for (;0 < w; ) G = 138 > w ? w : 138,

G > w - 3 && G < w && (G = w - 3), 10 >= G ? (L[I++] = 17, L[I++] = G - 3, P[17]++) : (L[I++] = 18,

L[I++] = G - 11, P[18]++), w -= G; else if (L[I++] = M[u], P[M[u]]++, w--, 3 > w) for (;0 < w--; ) L[I++] = M[u],

P[M[u]]++; else for (;0 < w; ) G = 6 > w ? w : 6, G > w - 3 && G < w && (G = w - 3),

L[I++] = 16, L[I++] = G - 3, P[16]++, w -= G;

}

a = H ? L.subarray(0, I) : L.slice(0, I);

na = ta(P, 7);

for (F = 0; 19 > F; F++) va[F] = na[da[F]];

for (C = 19; 4 < C && 0 === va[C - 1]; C--) ;

Oa = ua(na);

x.d(p - 257, 5, v);

x.d(D - 1, 5, v);

x.d(C - 4, 4, v);

for (F = 0; F < C; F++) x.d(va[F], 3, v);

F = 0;

for (Pa = a.length; F < Pa; F++) if ($ = a[F], x.d(Oa[$], na[$], v), 16 <= $) {

F++;

switch ($) {

ray : Array)(Za.length), g, h, m, j;

for (j = 0; j < f; ++j) d[Za[j]] = X(a, 3);

g = S(d);

h = new (H ? Uint8Array : Array)(b);

m = new (H ? Uint8Array : Array)(e);

a.o(S(c.call(a, b, g, h)), S(c.call(a, e, g, m)));

}

V.prototype.o = function(a, c) {

var b = this.a, e = this.b;

this.u = a;

for (var f = b.length - 258, d, g, h, m; 256 !== (d = mb(this, a)); ) if (256 > d) e >= f && (this.b = e,

b = this.f(), e = this.b), b[e++] = d; else {

g = d - 257;

m = ab[g];

0 < cb[g] && (m += X(this, cb[g]));

d = mb(this, c);

h = eb[d];

0 < gb[d] && (h += X(this, gb[d]));

e >= f && (this.b = e, b = this.f(), e = this.b);

for (;m--; ) b[e] = b[e++ - h];

}

for (;8 <= this.e; ) this.e -= 8, this.c--;

this.b = e;

};

V.prototype.I = function(a, c) {

var b = this.a, e = this.b;

this.u = a;

this.b = 32768;

for (var f = b.length, d, g, h, m; 256 !== (d = mb(this, a)); ) if (256 > d) e >= f && (b = this.f(),

f = b.length), b[e++] = d; else {

g = d - 257;

m = ab[g];

0 < gb[d] && (h += X(this, gb[d]));

e + m > f && (b = this.f(), f = b.length);

for (;m--; ) b[e] = b[e++ - h];

}

for (;8 <= this.e; ) this.e -= 8, this.c--;

this.b = e;

};

V.prototype.f = function() {

var a = new (H ? Uint8Array : Array)(this.b - 32768), c = this.b - 32768, b, e, f = this.a;

if (H) a.set(f.subarray(32768, a.length)); else {

b = 0;

for (e = a.length; b < e; ++b) a[b] = f[b + 32768];

}

this.k.push(a);

this.q += a.length;

if (H) f.set(f.subarray(c, c + 32768)); else for (b = 0; 32768 > b; ++b) f[b] = f[c + b];

this.b = 32768;

return f;

};

/\*---------------------农药指标一览表初始化代码类 .h ----------------------------\*/

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.SQLite;

using System.Net.Sockets;

using System.Net;

using System.Threading;

using System.IO;

using System.Drawing.Imaging;

using iTextSharp.text;

using iTextSharp.text.pdf;

using System.Diagnostics;

namespace Client

{

public partial class 农药指标一览表 : Form

{

private SQLiteConnection conn;

private SQLiteCommand cmd;

public 农药指标一览表()

{

InitializeComponent();

//GetFolderTreeView("");

//GetData();

DataSet ds2 = GetData();

ListViewItem lviItem;

ListViewItem.ListViewSubItem lviSubItem;

listView2.Clear();

GetFolderTreeView("");

//listView2.Clear();

listView2.BeginUpdate();

foreach (DataRow item in ds2.Tables[0].Rows)

{

//string sss = item["Name"].ToString();

lviItem = new ListViewItem();

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["pesticide\_id"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["pesticide\_name"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["pesticide\_level"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["prod\_category"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["product"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["Max\_remain"].ToString();

lviItem.SubItems.Add(lviSubItem);

listView2.Items.Add(lviItem);

}

listView2.EndUpdate();

}

private void GetFolderTreeView(string asPath)

{

ListViewItem item = new ListViewItem();

listView2.View = View.Details;

ColumnHeader clh;

clh = new ColumnHeader();

clh.Text = "#";

listView2.Columns.Add("\*", 10, HorizontalAlignment.Left);

clh = new ColumnHeader();

clh.Text = "农药编号";

listView2.Columns.Add(clh);

listView2.View = View.Details;

clh = new ColumnHeader();

clh.Text = "农药名称";

listView2.Columns.Add(clh);

listView2.View = View.Details;

clh = new ColumnHeader();

clh.Text = "毒性";

listView2.Columns.Add(clh);

listView2.View = View.Details;

clh = new ColumnHeader();

clh.Text = "作物类别";

listView2.Columns.Add(clh);

listView2.View = View.Details;

clh = new ColumnHeader();

clh.Text = "作物";

listView2.Columns.Add(clh);

listView2.View = View.Details;

clh = new ColumnHeader();

clh.Text = "最大残留值（mg/kg）";

listView2.Columns.Add(clh);

int cs = listView2.Width / listView2.Columns.Count;

int flg = 0;

foreach (ColumnHeader column in this.listView2.Columns)

{

if (flg == 0) { flg = 1; continue; }

column.Width = cs;

}

}

public DataSet GetData()

{

try

{

//连接字符串

conn = new SQLiteConnection(@"Data Source=D:\client\production\_client.db3");

conn.Open();

cmd = conn.CreateCommand();

string sql = "SELECT pesticide\_id,pesticide\_name,pesticide\_level,prod\_category,product,Max\_remain FROM Pesticide\_Map where ";

if (textBox1.Text.ToString() != "")

{

sql = sql + " pesticide\_name = @pesticide\_name";

cmd.Parameters.AddWithValue("@pesticide\_name", textBox1.Text.ToString());

}

if (textBox5.Text.ToString() != "")

{

sql = sql + " prod\_category = @prod\_category";

cmd.Parameters.AddWithValue("@prod\_category", textBox5.Text.ToString());

}

if ((textBox1.Text.ToString() == "") && (textBox5.Text.ToString() == ""))

{

sql = "SELECT pesticide\_id,pesticide\_name,pesticide\_level,prod\_category,product,Max\_remain FROM Pesticide\_Map ";

}

cmd.CommandText = sql.ToString();

SQLiteDataAdapter slda = new SQLiteDataAdapter(cmd);

DataSet ds = new DataSet();

slda.Fill(ds);

conn.Close();

return ds;

}

catch (Exception)

{

throw;

}

}

private void listView2\_SelectedIndexChanged(object sender, EventArgs e)

{

}

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

{

DataSet ds2 = GetData();

ListViewItem lviItem;

ListViewItem.ListViewSubItem lviSubItem;

listView2.Clear();

GetFolderTreeView("");

//listView2.Clear();

listView2.BeginUpdate();

foreach (DataRow item in ds2.Tables[0].Rows)

{

//string sss = item["Name"].ToString();

lviItem = new ListViewItem();

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["pesticide\_id"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["pesticide\_name"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["pesticide\_level"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["prod\_category"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["product"].ToString();

lviItem.SubItems.Add(lviSubItem);

lviSubItem = new ListViewItem.ListViewSubItem();

lviSubItem.Text = item["Max\_remain"].ToString();

lviItem.SubItems.Add(lviSubItem);

listView2.Items.Add(lviItem);

}

listView2.EndUpdate();

}

}