rem @echo off

set CLASSPATH=.

REM \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

REM \* ADD DATE to files

set THIS\_RUN=C:\000\_aaa\_stab\STB\_MAIN

cd %THIS\_RUN%

DateAdder

REM \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

REM \* ratio\_SS\_LIVE

set THIS\_RUN=C:\000\_aaa\_stab\ratio\_SS\_LIVE

set ACTIVE\_PAIRS\_MAIN=10

cd %THIS\_RUN%

copy /Y "c:\upcomingsplits.txt" .

java splits

REM java SA\_SetWTBParams 0.9 0.85 0.7 0.5 6 3 0.75 .75 .6 %ACTIVE\_PAIRS\_MAIN% '23/11/2006' '23/11/2006' "ENDOFDAY"

REM java CopyNewGens\_to\_HIST (mod TD on GEN 1st then copy)

copy /Y "c:\\*NASDAQ Financial\*" %THIS\_RUN%\hist\"FlashQuotes - NASDAQ Financial-100.txt"

copy /Y "c:\\*NASDAQ 100\*" %THIS\_RUN%\hist\"Nasdaq 100 Flash Quotes.txt"

rem date format in nxt 2 needs to be Jan 25, 2008 after nasdaq.com

java SA\_todays\_NASDAQ\_prices

java SA\_todays\_NASDAQ\_FIN\_prices

java ss\_param\_based\_wtb3

java wtc\_param\_based

REM the new c sharp reporter

justgetdataSS %THIS\_RUN%\DailyReports\

REM summary rep

reportSPASS %THIS\_RUN%\DailyReports\

java SA\_EOD

REM eod report

reportSPASS %THIS\_RUN%\DailyReports\

rem REGRESSIONSS C:\REG\_base\_2.xls

rem ++++++++++++++++++++++++++++++++++++++++++++++++++++++

move /Y "c:\\*NASDAQ Financial\*" "C:\000\_aaa\_stab\STB\_MAIN\pricearch"

move /Y "c:\\*NASDAQ 100\*" "C:\000\_aaa\_stab\STB\_MAIN\pricearch"

REGRESSIONSS

using System;

using System.Collections.Generic;

using System.Text;

using System.Data;

using System.IO;

using Excel = Microsoft.Office.Interop.Excel;

using Office = Microsoft.Office.Core;

//using System.Runtime.InteropServices;

//using VBIDE = Microsoft.Vbe.Interop;

using System.Diagnostics;

using System.Reflection; // tis has defs for values like Missing.\* in the excel sub calls

//using System.Runtime.InteropServices;

//using System.Threading;

//using System.ComponentModel;

using VBIDE = Microsoft.Vbe.Interop; // C:\Program Files\Microsoft Office\Excel.Net\Microsoft.Vbe.Interop.dll

using System.Data.SqlClient;

using System.Xml;

namespace REGRESSIONSS

{

class Program

{

//string FILENAME = "LOG\_REGRESSIONSS.txt";

StreamWriter o;

StreamWriter w;

void loginit()

{

o = File.CreateText(@"C:\000\_aaa\_stab\ratio\_SS\_LIVE\LOG\_REGRESSIONSS.txt");

o.Close();

}

void log(string s)

{

w = File.AppendText(@"C:\000\_aaa\_stab\ratio\_SS\_LIVE\LOG\_REGRESSIONSS.txt");

w.WriteLine(DateTime.Now.ToString() + " " + "LOG REGRESSTION() " + s);

w.Close();

}

static void Main(string[] args)

{

Program me = new Program();

// which base spreadsheeet depends on scatter or ratio and where u r

string REG\_SPREADSHEET;

if (args.Length != 1)

{

Console.WriteLine("usage: REGRESSION <REG\_SPREADSHEET>");

Console.WriteLine(@"eg: JustGetData 'C:\REG\_base\_2.xls\'");

Console.WriteLine(@"or");

Console.WriteLine(@"eg: JustGetData 'C:\REG\_base\_2\_scatter.xls\'");

Console.WriteLine(@" ");

}

REG\_SPREADSHEET = args[0];

#region log setup

me.loginit();

me.log("Start");

#endregion log setup

#region delete old run

DataSet1.last\_loadDataTable last\_load\_dt = new DataSet1.last\_loadDataTable();

DataSet1TableAdapters.GEN\_TRD\_basisTableAdapter GEN\_TRD\_basis\_ta = new REGRESSIONSS.DataSet1TableAdapters.GEN\_TRD\_basisTableAdapter();

GEN\_TRD\_basis\_ta.DeleteQuery();

#endregion delete old run

#region loop round doing regression

#region get last load values for loop

DataSet1TableAdapters.last\_loadTableAdapter last\_load\_ta = new REGRESSIONSS.DataSet1TableAdapters.last\_loadTableAdapter();

last\_load\_ta.Fill(last\_load\_dt);

string regDt2; // date for regression file saved later

#region variables

// do the new Run

String X\_ticker, Y\_ticker;

int MAX\_PER;

double m, c, r, X\_currentPrice, X\_pctChange, X\_Change, X\_volume, Y\_currentPrice;

double X\_prvPrice, Y\_prvPrice;

//String TD;

double PER\_LOW\_m, PER\_LOW\_c, PER\_LOW\_r;

double PER\_MED\_m, PER\_MED\_c, PER\_MED\_r;

double PER\_HIGH\_m, PER\_HIGH\_c, PER\_HIGH\_r;

System.DateTime last\_OVERNIGHT\_DT;

double lastL, stdevp;

double stdevp220, stdevp130, stdevp40, stdevp10;

//double currentL220, currentL130, currentL40;

#endregion variables

//foreach (DataSet1.last\_loadRow last\_load\_dr in last\_load\_dt.Rows)

int cnt = 0;

// this loops round the rows to get the x tickers in the reg calc

for (int i = 0; i < last\_load\_dt.Rows.Count; i++)

{

DataSet1.last\_loadRow x\_row = (DataSet1.last\_loadRow)last\_load\_dt.Rows[i];

X\_ticker = x\_row["ticker"].ToString();

regDt2 = x\_row["dt"].ToString(); // date for regression file saved later

DateTime mdt = Convert.ToDateTime(regDt2);

regDt2 = mdt.ToString("yyyyMMdd"); // it is a bit of a jiggle to get into right format

// this loops round the rows to get the y tickers in the reg calc

for (int j = i + 1; j < last\_load\_dt.Rows.Count; j++)

{

cnt++;

DataSet1.last\_loadRow y\_row = (DataSet1.last\_loadRow)last\_load\_dt.Rows[j];

Y\_ticker = y\_row["ticker"].ToString();

//uint need args log FileStyleUriParser

me.log(" ");

double pct = 0.0;

int totcalcs = (last\_load\_dt.Rows.Count \* last\_load\_dt.Rows.Count - last\_load\_dt.Rows.Count) / 2;

pct = Math.Round(100.0 \* cnt / totcalcs, 4);

Console.WriteLine(X\_ticker + " " + Y\_ticker + " i " + i + " j " + j + " " + cnt + " of " + totcalcs + " pct complete " + pct + " %");

me.log(X\_ticker + " " + Y\_ticker + " i " + i + " j " + j + " " + cnt + " of " + totcalcs + " pct complete " + pct + " %");

#region child data

// get child data

DataSet1.testdatahistoryfrorregbasisDataTable history\_dt = new DataSet1.testdatahistoryfrorregbasisDataTable();

DataSet1TableAdapters.testdatahistoryfrorregbasisTableAdapter history\_ta = new REGRESSIONSS.DataSet1TableAdapters.testdatahistoryfrorregbasisTableAdapter();

history\_ta.FillBy(history\_dt, X\_ticker, Y\_ticker);

//history\_ta.GetDataBy(Y\_ticker, Y\_ticker);

me.log(" Open c1 to write " + history\_dt.Rows.Count + " rows");

if (history\_dt.Rows.Count < 223)

{

me.log(" SKIP PAIR " + Y\_ticker + " " + Y\_ticker + " break loop as not enough values");

//break;

}

# region C1 file setup

// write C1.txt as j and C2.txt as i

using (StreamWriter w = File.CreateText(@"C:\000\_aaa\_stab\ratio\_SS\_LIVE\C1.txt"))

{

w.WriteLine(X\_ticker);

int c1\_cnt = 0;

foreach (DataRow rowP in history\_dt.Rows)

{

c1\_cnt++;

if (c1\_cnt <= 250)

{

String price;

price = rowP.ItemArray[1].ToString(); // Xclose

w.WriteLine(price);

}

}

w.Close();

}

me.log(" Close c1");

# endregion C1 file setup

# region C2 file setup

// write C2.txt as j and C2.txt as i

me.log(" Open c2 to write " + history\_dt.Rows.Count + " rows");

using (StreamWriter w = File.CreateText(@"C:\000\_aaa\_stab\ratio\_SS\_LIVE\C2.txt"))

{

w.WriteLine(Y\_ticker);

int c2\_cnt = 0;

foreach (DataRow rowP in history\_dt.Rows)

{

c2\_cnt++;

if (c2\_cnt <= 250)

{

String price;

price = rowP.ItemArray[3].ToString(); // Yclose

w.WriteLine(price);

}

}

w.Close();

}

me.log(" Close c2");

# endregion C2 file setup

#region excel

Excel.Application excelApp = new Excel.ApplicationClass(); // Creates a new Excel Application

excelApp.Visible = true; // Makes Excel visible to the user.

excelApp.DisplayAlerts = false;

// The following line adds a new workbook

//Excel.Workbook newWorkbook = excelApp.Workbooks.Add(XlWBATemplate.xlWBATWorksheet);

// The following code opens an existing workbook

// not used as command line : string workbookPath = @"C:/REG\_base\_2.xls"; // Add your own path here

Excel.Workbook excelWorkbook = excelApp.Workbooks.Open(REG\_SPREADSHEET, 0,

false, 5, "", "", false, Excel.XlPlatform.xlWindows, "", true,

false, 0, true, false, false);

// The following gets the Worksheets collection

Excel.Sheets excelSheets = excelWorkbook.Worksheets;

// The following gets Sheet1 for editing

string currentSheet = "Sheet1";

Excel.Worksheet excelWorksheet = (Excel.Worksheet)excelSheets.get\_Item(currentSheet);

// The following gets cell A1 for editing

//Excel.Range excelCell = (Excel.Range)excelWorksheet.get\_Range("A4", "A4");

// The following sets cell A1's value to "Hi There"

//excelCell.Value2 = "Hi There";

//20080214 dont need 2 call reg1 this as its now al in he excwl in the following macro

// that just lads prices sheet does rest

// excelWorksheet.Application.Run("Module3.LoadCFiles", Missing.Value

// excelWorksheet.Application.Run("Module4.vista\_load", Missing.Value

// excelWorksheet.Application.Run("Module6.load\_c2", Missing.Value

try

{

excelWorksheet.Application.Run("Module7.loadem", Missing.Value

, Missing.Value

, Missing.Value

, Missing.Value

, Missing.Value

, Missing.Value

, Missing.Value

, Missing.Value

, Missing.Value

, Missing.Value

, Missing.Value

, Missing.Value, Missing.Value, Missing.Value, Missing.Value,

Missing.Value, Missing.Value, Missing.Value, Missing.Value,

Missing.Value, Missing.Value, Missing.Value, Missing.Value,

Missing.Value, Missing.Value, Missing.Value, Missing.Value,

Missing.Value, Missing.Value, Missing.Value);

}

catch (Exception e)

{

Console.WriteLine("[whoops] {0}", e);

}

#region 220

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

// 220

//

Excel.Range EC\_m220 = (Excel.Range)excelWorksheet.get\_Range("D3", "D3");

String m220 = EC\_m220.Value2.ToString();

Excel.Range EC\_c220 = (Excel.Range)excelWorksheet.get\_Range("E3", "E3");

String c220 = EC\_c220.Value2.ToString();

Excel.Range EC\_r220 = (Excel.Range)excelWorksheet.get\_Range("D5", "D5");

String r220 = EC\_r220.Value2.ToString();

Excel.Range avg220ratio = (Excel.Range)excelWorksheet.get\_Range("D14", "D14");

String average220ratio = avg220ratio.Value2.ToString();

//

// 220

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

#endregion 220

#region 130

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

// 130

//

Excel.Range EC\_m130 = (Excel.Range)excelWorksheet.get\_Range("G3", "G3");

String m130 = EC\_m130.Value2.ToString();

Excel.Range EC\_c130 = (Excel.Range)excelWorksheet.get\_Range("H3", "H3");

String c130 = EC\_c130.Value2.ToString();

Excel.Range EC\_r130 = (Excel.Range)excelWorksheet.get\_Range("G5", "G5");

String r130 = EC\_r130.Value2.ToString();

Excel.Range avg130ratio = (Excel.Range)excelWorksheet.get\_Range("G14", "G14");

String average130ratio = avg220ratio.Value2.ToString();

//

// 130

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

#endregion 130

#region 40

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

// 40

//

Excel.Range EC\_m40 = (Excel.Range)excelWorksheet.get\_Range("J3", "J3");

String m40 = EC\_m40.Value2.ToString();

Excel.Range EC\_c40 = (Excel.Range)excelWorksheet.get\_Range("K3", "K3");

String c40 = EC\_c40.Value2.ToString();

Excel.Range EC\_r40 = (Excel.Range)excelWorksheet.get\_Range("J5", "J5");

String r40 = EC\_r40.Value2.ToString();

Excel.Range avg40ratio = (Excel.Range)excelWorksheet.get\_Range("J14", "J14");

String average40ratio = avg40ratio.Value2.ToString();

//

// 40

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

#endregion 40

#region 10

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

// 10

// INTRODUCED TO CAT IT AS IT HAPPENS one cell less that above cos i did it manual

//

Excel.Range EC\_m10 = (Excel.Range)excelWorksheet.get\_Range("M2", "M2");

String m10 = EC\_m10.Value2.ToString();

Excel.Range EC\_c10 = (Excel.Range)excelWorksheet.get\_Range("N2", "N2");

String c10 = EC\_c10.Value2.ToString();

Excel.Range EC\_r10 = (Excel.Range)excelWorksheet.get\_Range("M4", "M4");

String r10 = EC\_r10.Value2.ToString();

Excel.Range avg10ratio = (Excel.Range)excelWorksheet.get\_Range("M14", "M14");

String average10ratio = avg10ratio.Value2.ToString();

//

// 10

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

#endregion 10

#region stdevp's

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

// stdevps

//

Excel.Range EC\_stdevp220 = (Excel.Range)excelWorksheet.get\_Range("D11", "D11");

String stdevp220S = EC\_stdevp220.Value2.ToString();

Excel.Range EC\_stdevp130 = (Excel.Range)excelWorksheet.get\_Range("G11", "G11");

String stdevp130S = EC\_stdevp130.Value2.ToString();

Excel.Range EC\_stdevp40 = (Excel.Range)excelWorksheet.get\_Range("J11", "J11");

String stdevp40S = EC\_stdevp40.Value2.ToString();

Excel.Range EC\_stdevp10 = (Excel.Range)excelWorksheet.get\_Range("J11", "J11");

String stdevp10S = EC\_stdevp10.Value2.ToString();

stdevp220 = Convert.ToDouble(stdevp220S);

stdevp130 = Convert.ToDouble(stdevp130S);

stdevp40 = Convert.ToDouble(stdevp40S);

stdevp10 = Convert.ToDouble(stdevp10S);

//

// stdevp's

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

#endregion stdevp's

// got all the vals now stickem in a db

#region store in db

MAX\_PER = 220; // if max per < num in array then 0

X\_currentPrice = 0;

X\_pctChange = 0;

X\_Change = 0;

X\_volume = 0;

Y\_currentPrice = 0;

X\_prvPrice = 0;

Y\_prvPrice = 0;

// TD = @"20/4/2007";

PER\_LOW\_m = Convert.ToDouble(m40);

PER\_LOW\_c = Convert.ToDouble(c40);

PER\_LOW\_r = Convert.ToDouble(r40);

PER\_MED\_m = Convert.ToDouble(m130);

PER\_MED\_c = Convert.ToDouble(c130);

PER\_MED\_r = Convert.ToDouble(r130);

PER\_HIGH\_m = Convert.ToDouble(m220);

PER\_HIGH\_c = Convert.ToDouble(c220);

PER\_HIGH\_r = Convert.ToDouble(r220);

last\_OVERNIGHT\_DT = DateTime.Now;

lastL = 0.22;

// 20070817 the period 10 things

/\* WAS

stdevp = 0.876;

m = 0.0;

c = 0.0;

r = 0.0;

\* \*/

stdevp = stdevp10;

m = Convert.ToDouble(m10);

c = Convert.ToDouble(c10);

r = Convert.ToDouble(r10);

System.DateTime TD = new DateTime(2002, 1, 3);

TD = DateTime.Today;

GEN\_TRD\_basis\_ta.Insert(X\_ticker, Y\_ticker, MAX\_PER, m, c, r, X\_currentPrice, X\_pctChange, X\_Change, X\_volume, Y\_currentPrice, X\_prvPrice, Y\_prvPrice, TD, PER\_LOW\_m, PER\_LOW\_c, PER\_LOW\_r, PER\_MED\_m, PER\_MED\_c, PER\_MED\_r, PER\_HIGH\_m, PER\_HIGH\_c, PER\_HIGH\_r, last\_OVERNIGHT\_DT, lastL, stdevp, stdevp220, stdevp130, stdevp40, Convert.ToDouble(average220ratio), Convert.ToDouble(average130ratio), Convert.ToDouble(average40ratio), Convert.ToDouble(average10ratio));

//this.gEN\_TRD\_basisTableAdapter.Insert(X\_ticker, Y\_ticker, MAX\_PER, m, c, r, X\_currentPrice, X\_pctChange, X\_Change, X\_volume, Y\_currentPrice, X\_prvPrice, Y\_prvPrice, TD, PER\_LOW\_m, PER\_LOW\_c, PER\_LOW\_r, PER\_MED\_m, PER\_MED\_c, PER\_MED\_r, PER\_HIGH\_m, PER\_HIGH\_c, PER\_HIGH\_r, last\_OVERNIGHT\_DT, lastL, stdevp, stdevp220, stdevp130, stdevp40);

me.log(@" Excel being stored " + X\_ticker + @" \*\*\* Y\_ticker=\*\*\* " + Y\_ticker + @" with lastovernightdt " + last\_OVERNIGHT\_DT + " for TD=" + TD);

#endregion store in db

#region store spreadsheet if needed

//used to use date t instead of regDt2

//String t = DateTime.Now.ToString("yyyyMMdd") ;

//excelWorkbook.SaveAs((object)@"c:\temp\REG " + X\_ticker + " " + Y\_ticker +" " + regDt2 + ".xls", Missing.Value, Missing.Value

// , Missing.Value, Missing.Value, Missing.Value

// , Microsoft.Office.Interop.Excel.XlSaveAsAccessMode.xlShared

// , Microsoft.Office.Interop.Excel.XlSaveConflictResolution.xlLocalSessionChanges

// , Missing.Value

// , Missing.Value

// , Missing.Value

// , Missing.Value

// );

#endregion store spreadsheet

try

{

excelApp.Sheets.Application.Quit();

}

catch (Exception whathappened)

{

Debug.Print(whathappened.ToString());

}

try

{

excelApp.Workbooks.Close();

}

catch (Exception whathappened)

{

Debug.Print(whathappened.ToString());

}

excelApp.Quit();

#endregion excel

#endregion child data

}

// useful: foreach (DataColumn column in SPADataTable.Columns)

}

//Console.ReadLine();

// for (int i=0; i< dataSet21.last\_load.Rows.Count; i++) {

#endregion get last load values for loop

#endregion loop round doing regression

}

}

}

JUSTGETDATA

using System;

using System.Collections.Generic;

using System.Text;

using System.Data;

using System.Diagnostics;

using System.IO;

using System.Reflection; // for Missing.value

using Excel = Microsoft.Office.Interop.Excel;

using Office = Microsoft.Office.Core;

//using System.Reflection;

using System.Runtime.InteropServices;

//using System.Threading;

using VBIDE = Microsoft.Vbe.Interop; // C:\Program Files\Microsoft Office\Excel.Net\Microsoft.Vbe.Interop.dll

namespace JustGetData

{

class Program

{

static void Main(string[] args)

{

string todaysDate4Reports="";

string BASE\_REPORT\_DIR;

if (args.Length != 1 )

{

Console.WriteLine("usage: JustGetData <baseDirName>");

Console.WriteLine(@"eg: JustGetData 'C:\000\_aaa\_stabs\STB\_MAIN\'");

Console.WriteLine(@"dont forget the final slash !");

}

BASE\_REPORT\_DIR = args[0];

// if u wanna add the next line u have to add a settings.settings file from the new items for the project

string wifeName = global::JustGetData.Settings1.Default.wifesName;

#region get today for filename

// this dataset is for all tables in dataset declared here

// now the specifics for the today table

DataSet1.TODAYDataTable todayDataTable ;

todayDataTable = new DataSet1.TODAYDataTable();

DataSet1TableAdapters.TODAYTableAdapter myTodayTableAdapter =

new DataSet1TableAdapters.TODAYTableAdapter();

myTodayTableAdapter.Fill(todayDataTable);

DataSet1.TODAYRow todayRow;

foreach (DataSet1.TODAYRow x in todayDataTable.Rows)

{

todaysDate4Reports = x.today\_dt.ToString();

// dd/mm/yyyy

todaysDate4Reports = todaysDate4Reports.Substring(6, 4) +

todaysDate4Reports.Substring(3, 2) +

todaysDate4Reports.Substring(0, 2);

}

// this is wot i tried but failed - dont know why

// had to do the preceeding loop to get round it

//todaysDate4Reports = todayDataTable.Rows[0].ToString();// todayDataTable.Rows[0]; // there is only one so dont need an array

#endregion get today for filename

# region parent\_setup

DataSet1 mydataset = new DataSet1();

DataSet1.SuggestedPositionArchiveDataTable dataTable;

dataTable = new DataSet1.SuggestedPositionArchiveDataTable();

DataSet1TableAdapters.SuggestedPositionArchiveTableAdapter mytableadapter =

new DataSet1TableAdapters.SuggestedPositionArchiveTableAdapter();

mydataset.EnforceConstraints = false;

mytableadapter.Fill(dataTable);

#endregion parent\_setup

# region clear previous run excel files

// base dir for reports

BASE\_REPORT\_DIR = BASE\_REPORT\_DIR + todaysDate4Reports + @"\";

//BASE\_REPORT\_DIR = @"C:\000\_aaa\_stabs\stb\_main\DailyReports" ;

DirectoryInfo di=new DirectoryInfo(BASE\_REPORT\_DIR);

// if there isnt a dir then create one

if (!di.Exists )

{

di.Create();

}

else

{

// delete any files in it

FileInfo[] fii = di.GetFiles("\*.xls");

if (fii.GetLength(0) > 0)

{

foreach (FileInfo f in fii)

{

f.Delete();

}

}

FileInfo[] fiii = di.GetFiles("\*.html");

if (fiii.GetLength(0) > 0)

{

foreach (FileInfo f in fiii)

{

f.Delete();

}

}

}

# endregion clear previous run excel files

# region child\_setup

DataSet1.testdata\_historyfrorregbasisDataTable childTable;

childTable=new DataSet1.testdata\_historyfrorregbasisDataTable();

DataSet1 mychildset = new DataSet1();

DataSet1TableAdapters.testdata\_historyfrorregbasisTableAdapter mychildadapter =

new DataSet1TableAdapters.testdata\_historyfrorregbasisTableAdapter();

//mydataset.EnforceConstraints = false;

//mytableadapter.Fill(dataTable);

#endregion child\_setup

StreamWriter sw = new StreamWriter(@"c:\TestFileSW.txt");

// Add some text to the file.

sw.Write("This is the ");

sw.WriteLine("header for the file.");

sw.WriteLine("-------------------");

// Arbitrary objects can also be written to the file.

sw.Write("The date is: ");

sw.WriteLine(DateTime.Now);

Stream myFile = File.Create(@"c:\TestTLFile.txt");

/\* Create a new text writer using the output stream, and add it to

\* the trace listeners. \*/

TextWriterTraceListener myTextListener = new TextWriterTraceListener(myFile);

Trace.Listeners.Add(myTextListener);

// Write output to the file.

Trace.Write("Test output ");

sw.WriteLine("Writing reports to " + @"C:\000\_aaa\_stabs\stb\_testdata");

// Flush the output.

Trace.Flush();

foreach (DataSet1.SuggestedPositionArchiveRow dr in dataTable.Rows)

{

Console.WriteLine(dr.x\_ticker + " " + dr.y\_ticker);

// THE VIEW RETRIEVES EM IN DATE DESC ORDER - CURRENT PRICE IS THEREFORE 1ST

mychildadapter.FillBy(childTable, dr.x\_ticker, dr.y\_ticker);

#region excel

// INSERT

Excel.Application excelApp = new Excel.ApplicationClass(); // Creates a new Excel Application

excelApp.Visible = true; // Makes Excel visible to the user.

// The following line adds a new workbook

//Excel.Workbook newWorkbook = excelApp.Workbooks.Add(XlWBATemplate.xlWBATWorksheet);

// The following code opens an existing workbook

string workbookPath = @"C:/WTB\_report\_template.xls"; // Add your own path here

Excel.Workbook excelWorkbook = excelApp.Workbooks.Open(workbookPath, 0,

false, 5, "", "", false, Excel.XlPlatform.xlWindows, "", true,

false, 0, true, false, false);

//// The following gets the Worksheets collection

Excel.Sheets excelSheets = excelWorkbook.Worksheets;

//// The following gets Sheet1 for editing

string currentSheet = "frontend";

Excel.Worksheet excelWorksheet = (Excel.Worksheet)excelSheets.get\_Item(currentSheet);

//// The following gets cell A1 for editing

Excel.Range excelCell = (Excel.Range)excelWorksheet.get\_Range("A4", "A4");

//// The following sets cell A1's value to "Hi There"

//excelCell.Value2 = "Hi There";

//excelCell.Font.Italic = true;

//System.DateTime TD = new DateTime(2002, 1, 3);

#region excel\_header\_details

excelCell = (Excel.Range)excelWorksheet.get\_Range("A1", "A1");

excelCell.Value2 = dr.x\_ticker;

excelCell = (Excel.Range)excelWorksheet.get\_Range("B1", "B1");

excelCell.Value2 = dr.y\_ticker;

excelCell = (Excel.Range)excelWorksheet.get\_Range("F5", "F5");

excelCell.Value2 = dr.PER\_HIGH\_m;

excelCell = (Excel.Range)excelWorksheet.get\_Range("F6", "F6");

excelCell.Value2 = dr.PER\_MED\_m;

excelCell = (Excel.Range)excelWorksheet.get\_Range("F7", "F7");

excelCell.Value2 = dr.PER\_LOW\_m;

excelCell = (Excel.Range)excelWorksheet.get\_Range("G5", "G5");

excelCell.Value2 = dr.PER\_HIGH\_c;

excelCell = (Excel.Range)excelWorksheet.get\_Range("G6", "G6");

excelCell.Value2 = dr.PER\_MED\_c;

excelCell = (Excel.Range)excelWorksheet.get\_Range("G7", "G7");

excelCell.Value2 = dr.PER\_LOW\_c;

excelCell = (Excel.Range)excelWorksheet.get\_Range("H5", "H5");

excelCell.Value2 = dr.PER\_high\_r;

excelCell = (Excel.Range)excelWorksheet.get\_Range("H6", "H6");

excelCell.Value2 = dr.PER\_med\_r;

excelCell = (Excel.Range)excelWorksheet.get\_Range("H7", "H7");

excelCell.Value2 = dr.PER\_low\_r;

excelCell = (Excel.Range)excelWorksheet.get\_Range("I5", "I5");

excelCell.Value2 = dr.stdevp220;

excelCell = (Excel.Range)excelWorksheet.get\_Range("I6", "I6");

excelCell.Value2 = dr.stdevp130;

excelCell = (Excel.Range)excelWorksheet.get\_Range("I7", "I7");

excelCell.Value2 = dr.stdevp40;

#endregion excel\_header\_details

//INSERT END

#endregion excel

int noOfPricesRequired = 250;

int noRetrievedSoFar = 0;

foreach (DataSet1.testdata\_historyfrorregbasisRow cr in childTable.Rows)

{

// Console.WriteLine(" " + cr.X\_ticker + " " + cr.Xclose

// +" " + cr.Y\_ticker + " " + cr.Yclose);

if (noRetrievedSoFar<=noOfPricesRequired)

{

excelCell = (Excel.Range)excelWorksheet.get\_Range("A" + (noRetrievedSoFar + 2), "A" + (noRetrievedSoFar + 2));

excelCell.Value2 = cr.Xclose;

excelCell = (Excel.Range)excelWorksheet.get\_Range("B" + (noRetrievedSoFar + 2), "B" + (noRetrievedSoFar + 2));

excelCell.Value2 = cr.Yclose;

sw.WriteLine("Prices put in excel for " + cr.Y\_ticker + " " + cr.Yclose);

noRetrievedSoFar++;

}

}

//}

#region formReportDir

// save reports locally depending if it is closed or new

string saveItDir = BASE\_REPORT\_DIR; // @"C:\000\_aaa\_stabs\stb\_main\DailyReports\SuggestedPositions";

//string saveItDir = @".\DailyReports\SuggestedPositions";

string closeDt = "";

if ( dr.IscloseDtNull() == false )

{//if there is a date convert it otherwise it is open

DateTime dtf;

DateTime.TryParse(dr.closeDt.ToString(), out dtf);

closeDt = dtf.ToString("yyyyMMdd");

}

#endregion formReportDir

// commented out the print it bit

#region print it

//try

//{

// excelWorksheet.Application.Run("smry.PrintGraphs", Missing.Value

//, Missing.Value

//, Missing.Value

//, Missing.Value

//, Missing.Value

//, Missing.Value

//, Missing.Value

//, Missing.Value

//, Missing.Value

//, Missing.Value

//, Missing.Value

//, Missing.Value, Missing.Value, Missing.Value, Missing.Value,

//Missing.Value, Missing.Value, Missing.Value, Missing.Value,

//Missing.Value, Missing.Value, Missing.Value, Missing.Value,

//Missing.Value, Missing.Value, Missing.Value, Missing.Value,

//Missing.Value, Missing.Value, Missing.Value);

//}

//catch (Exception e1)

//{

// Debug.Print(e1.ToString());

//}

#endregion print it

# region work out position type NEW OPEN CLOSED

// closed if there is a closed date

// new if not closed and open day is today (as per last load)

// open if not closed and not new

// open by default and amended in this region

string positionType = " pos-OPEN ";

if (dr.openDt.ToString("yyyMMdd").Equals(todaysDate4Reports))

{

Console.WriteLine("its new today");

positionType = " pos-NEW";

}

if (closeDt.CompareTo(todaysDate4Reports)<=0 )

{

if (!(closeDt.CompareTo("") == 0))

{

positionType = " pos-CLOSED ";

}

}

#endregion

#region store spreadsheet if needed

//todaysDate4Reports contains todays date from last load for filename

string saveAsFileName = @"test graphi 3 copy" + positionType + " seq-" + dr.seq + " report date-" +

todaysDate4Reports + " close date=" + closeDt + " ticker pairs " +

dr.x\_ticker + " " + dr.y\_ticker + ".xls";

string fullFilePath = saveItDir.ToString() + saveAsFileName;

Console.WriteLine("Saving ss to " + fullFilePath);

excelWorkbook.SaveAs(fullFilePath, Missing.Value, Missing.Value

, Missing.Value, Missing.Value, Missing.Value

, Microsoft.Office.Interop.Excel.XlSaveAsAccessMode.xlShared

, Microsoft.Office.Interop.Excel.XlSaveConflictResolution.xlLocalSessionChanges

, Missing.Value

, Missing.Value

, Missing.Value

, Missing.Value

);

#endregion store spreadsheet

#region save the associated yahoo finance html pages

//

// keep the same name so when u dir the by name they r together

string saveAsHTMLFileName = saveItDir + @"\test graphi 3 copy" + positionType +

" seq-" + dr.seq + " report date-" +

todaysDate4Reports + " close date=" + closeDt +

" ticker pairs " + dr.x\_ticker + " " + dr.y\_ticker + ".html";

StreamWriter sw\_html = new StreamWriter(saveAsHTMLFileName);

# region buy or sell string x\_op y\_op

//x\_op and y\_op is BUY or SELL depending on the sign of the quantity traded

string x\_op, y\_op;

if (dr.x\_quantity > 0)

{

x\_op = "BUY";

}

else

{

x\_op = "SELL";

}

if (dr.y\_quantity > 0)

{

y\_op = "BUY";

}

else

{

y\_op = "SELL";

}

# endregion buy or sell string x\_op y\_op

#region write the html output

//http://finance.yahoo.com/q/bc?t=1y&s=CHKP&l=off&z=m&q=l&c=teva

sw\_html.WriteLine("<html>");

sw\_html.WriteLine(" <head>");

sw\_html.WriteLine(" <style type='text/css'> ");

sw\_html.WriteLine(" P { page-break-after: always } ");

sw\_html.WriteLine("</style> ");

sw\_html.WriteLine(" </head>");

sw\_html.WriteLine(" ");

sw\_html.WriteLine(" <body>");

sw\_html.WriteLine(" ");

sw\_html.WriteLine(" <table>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" <h2>");

sw\_html.WriteLine(" Pair " + dr.x\_ticker + " (" + x\_op + ") vs " + dr.y\_ticker + " (" + y\_op + ") ");

sw\_html.WriteLine(" </h2>");

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" One year comparison ADJUSTED");

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" <iframe width=900 height=800 src='http://finance.yahoo.com/q/bc?t=1y&s=" + dr.x\_ticker + "&l=off&z=m&q=l&c=" + dr.y\_ticker + "'> </iframe>");

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr><td><P></td></tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" 3 Month comparison ADJUSTED");

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" <iframe width=900 height=800 src='http://finance.yahoo.com/q/bc?t=3m&s=" + dr.x\_ticker + "&l=off&z=m&q=l&c=" + dr.y\_ticker + "'> </iframe>");

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr><td><P></td></tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" 5 Day comparison ADJUSTED");

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" <iframe width=900 height=800 src='http://finance.yahoo.com/q/bc?t=5d&s=" + dr.x\_ticker + "&l=off&z=m&q=l&c=" + dr.y\_ticker + "'> </iframe>");

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr><td><P></td></tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" 5 Day " + dr.x\_ticker + " ( " + x\_op + " ) ADJUSTED" );

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" <iframe width=900 height=800 src='http://finance.yahoo.com/q/bc?s="+dr.x\_ticker+"&t=3m&l=off&z=m&q=l&c='> </iframe>");

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr><td><P></td></tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" 5 Day " + dr.y\_ticker + " ( " + y\_op + " ) ADJUSTED" );

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" <iframe width=900 height=800 src='http://finance.yahoo.com/q/bc?s="+dr.y\_ticker+"&t=3m&l=off&z=m&q=l&c='> </iframe>");

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr><td><P></td></tr>");

// this BIT SHOWS UNADJUSTED. ITS BETTER LOOKING BUT WRONG.

// sw\_html.WriteLine(" <tr>");

// sw\_html.WriteLine(" <td>");

// sw\_html.WriteLine(" One month comparison");

// sw\_html.WriteLine(" </td>");

// sw\_html.WriteLine(" </tr>");

// sw\_html.WriteLine(" <tr>");

// sw\_html.WriteLine(" <td>");

// sw\_html.WriteLine(" <img border=0 width=600 src='http://cchart.finance.ukl.yahoo.com/z?s=" + dr.x\_ticker + "&c=" + dr.y\_ticker + "&t=1m&a=ss' alt='Chart'> ");

// sw\_html.WriteLine(" </td>");

// sw\_html.WriteLine(" </tr>");

// sw\_html.WriteLine(" <tr>");

// sw\_html.WriteLine(" <td>");

// sw\_html.WriteLine(" One year comparison");

// sw\_html.WriteLine(" </td>");

// sw\_html.WriteLine(" </tr>");

// sw\_html.WriteLine(" <tr>");

// sw\_html.WriteLine(" <td>");

// sw\_html.WriteLine(" <img border=0 width=600 src='http://cchart.finance.ukl.yahoo.com/z?s=" + dr.x\_ticker + "&c=" + dr.y\_ticker + "&t=1y&a=ss' alt='Chart'> ");

// sw\_html.WriteLine(" </td>");

// sw\_html.WriteLine(" </tr>");

// sw\_html.WriteLine(" <tr><td><P></td></tr>");

// sw\_html.WriteLine(" ");

// sw\_html.WriteLine(" <tr>");

// sw\_html.WriteLine(" <td>");

// sw\_html.WriteLine(" 1st stock");

// sw\_html.WriteLine(" </td>");

// sw\_html.WriteLine(" </tr>");

// sw\_html.WriteLine(" <tr>");

// sw\_html.WriteLine(" <td>");

// sw\_html.WriteLine(" <img border=0 width=600 src='http://cchart.finance.ukl.yahoo.com/z?s=" + dr.x\_ticker + "&t=1y&a=ss' alt='Chart'> ");

// sw\_html.WriteLine(" </td>");

// sw\_html.WriteLine(" </tr>");

// sw\_html.WriteLine(" <tr>");

// sw\_html.WriteLine(" <td>");

// sw\_html.WriteLine(" 2nd stock");

// sw\_html.WriteLine(" </td>");

// sw\_html.WriteLine(" </tr>");

// sw\_html.WriteLine(" <td>");

// sw\_html.WriteLine(" <tr>");

// sw\_html.WriteLine(" <td>");

// sw\_html.WriteLine(" <img border=0 width=600 src='http://cchart.finance.ukl.yahoo.com/z?s=" + dr.y\_ticker + "&t=1y&a=ss' alt='Chart'> ");

// sw\_html.WriteLine(" </td>");

// sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" stock news " + dr.x\_ticker + " ( " + x\_op + " )" );

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" <iframe width=600 height=400 src='http://finance.yahoo.com/q/h?s=" + dr.x\_ticker + "'></iframe> ");

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" stock news " + dr.y\_ticker + " ( " + y\_op + " )" );

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" <tr>");

sw\_html.WriteLine(" <td>");

sw\_html.WriteLine(" <iframe width=600 height=400 src='http://finance.yahoo.com/q/h?s=" + dr.y\_ticker + "'></iframe> ");

sw\_html.WriteLine(" </td>");

sw\_html.WriteLine(" </tr>");

sw\_html.WriteLine(" </table>");

sw\_html.WriteLine(" ");

sw\_html.WriteLine(" </body>");

sw\_html.WriteLine("</html>");

#endregion write the html output

sw\_html.Close();

//

#endregion save the associated yahoo finance html pages

#region close excel sheet

excelApp.Sheets.Application.Quit();

try

{

excelApp.Workbooks.Close();

}

catch (Exception whathappened)

{

Debug.Print(whathappened.ToString());

}

excelApp.Quit();

#endregion close excel sheet

}

// this is here to pause the program - can remove for final version

// Console.ReadLine();

}

}

}

REM set THIS\_RUN=C:\000\_aaa\_stab\ratio\_SS\_LIVE

REM set ACTIVE\_PAIRS\_MAIN=10

REM cd %THIS\_RUN%

REM REGRESSIONSS C:\REG\_base\_2.xls

REM java SA\_CopyRegAcross

REPORTSPASS

using System;

using System.Collections.Generic;

using System.Text;

using System.Data;

using System.Diagnostics;

using System.IO;

using System.Reflection; // for Missing.value

using Excel = Microsoft.Office.Interop.Excel;

using Office = Microsoft.Office.Core;

//using System.Reflection;

using System.Runtime.InteropServices;

//using System.Threading;

using VBIDE = Microsoft.Vbe.Interop; // C:\Program Files\Microsoft Office\Excel.Net\Microsoft.Vbe.Interop.dll

namespace reportSPASS

{

class Program

{

static void Main(string[] args)

{

string BASE\_REPORT\_DIR;

string FULL\_FILE\_PATH;

if (args.Length != 1)

{

Console.WriteLine("usage: reportSPA <baseDirName>");

Console.WriteLine(@"eg: reportSPA 'C:\000\_aaa\_stabs\STB\_MAIN\DailyReports\'");

Console.WriteLine(@"dont forget the final slash !");

}

BASE\_REPORT\_DIR = args[0];

//need to save in today dir

#region get today for filename

string todaysDate4Reports = "";

// this dataset is for all tables in dataset declared here

// now the specifics for the today table

DataSet1.TODAYDataTable todayDataTable;

todayDataTable = new DataSet1.TODAYDataTable();

DataSet1TableAdapters.TODAYTableAdapter myTodayTableAdapter =

new DataSet1TableAdapters.TODAYTableAdapter();

myTodayTableAdapter.Fill(todayDataTable);

DataSet1.TODAYRow todayRow;

foreach (DataSet1.TODAYRow x in todayDataTable.Rows)

{

todaysDate4Reports = x.today\_dt.ToString();

// dd/mm/yyyy

todaysDate4Reports = todaysDate4Reports.Substring(6, 4) +

todaysDate4Reports.Substring(3, 2) +

todaysDate4Reports.Substring(0, 2);

}

// this is wot i tried but failed - dont know why

// had to do the preceeding loop to get round it

//todaysDate4Reports = todayDataTable.Rows[0].ToString();// todayDataTable.Rows[0]; // there is only one so dont need an array

#endregion get today for filename

FULL\_FILE\_PATH = BASE\_REPORT\_DIR + todaysDate4Reports + @"\REPORT\_SPA";

//filename is pandl

#region tablevars

DataSet1.SPADataTable SPADataTable;

SPADataTable = new DataSet1.SPADataTable();

DataSet1TableAdapters.SPATableAdapter mySPATableAdapter =

new reportSPASS.DataSet1TableAdapters.SPATableAdapter();

mySPATableAdapter.Fill(SPADataTable);

DataSet1.SPADataTable SPARow;

#endregion tablevars

Excel.Application excelApp = new Excel.ApplicationClass(); // Creates a new Excel Application

excelApp.Visible = true; // Makes Excel visible to the user.

// The following code opens an existing workbook

string workbookPath = @"C:/blank.xls"; // Add your own path here

Excel.Workbook excelWorkbook = excelApp.Workbooks.Open(workbookPath, 0,

false, 5, "", "", false, Excel.XlPlatform.xlWindows, "", true,

false, 0, true, false, false);

Excel.Worksheet excelWorksheet = new Microsoft.Office.Interop.Excel.Worksheet();

//// The following gets the Worksheets collection

Excel.Sheets excelSheets = excelWorkbook.Worksheets;

//// The following gets Sheet1 for editing

string currentSheet = "Sheet1";

excelWorksheet = (Excel.Worksheet)excelSheets.get\_Item(currentSheet);

//// The following gets cell A1 for editing

Excel.Range excelCell = (Excel.Range)excelWorksheet.get\_Range("A4", "A4");

string[] rows = { "A", "B", "C" };

int rowCounter = 1;

int columnCounter = 1;

#region write all rows

foreach (DataColumn column in SPADataTable.Columns)

{

((Excel.Range)excelWorksheet.Cells[rowCounter, columnCounter]).Value2 =

column.ToString(); // "fred";

// bold row please may have to have a template ss

columnCounter++;

}

columnCounter = 1;

rowCounter++;

foreach (DataSet1.SPARow row in SPADataTable.Rows)

{

foreach (DataColumn column in SPADataTable.Columns)

{

// string rr = SPADataTable.Columns.IndexOf(row[column]).ToString();

//string rr = SPADataTable.Columns.IndexOf(column); //(row[column]).ToString() ; //column. .ToString();

//string cr = SPADataTable.Rows.IndexOf(row).ToString();

((Excel.Range)excelWorksheet.Cells[rowCounter, columnCounter]).Value2 =

row[column].ToString(); // "fred";

Console.WriteLine(row[column] + " row " + " col " + row[column].ToString());

columnCounter++;

}

columnCounter = 1;

rowCounter++;

}

#endregion write all rows

# region clear previous run excel files

// base dir for reports

BASE\_REPORT\_DIR = BASE\_REPORT\_DIR + todaysDate4Reports + @"\";

//BASE\_REPORT\_DIR = @"C:\000\_aaa\_stabs\stb\_main\DailyReports" ;

DirectoryInfo di = new DirectoryInfo(BASE\_REPORT\_DIR);

// if there isnt a dir then create one

Console.WriteLine("checking rep dir");

if (!di.Exists)

{

Console.WriteLine("didnt exist creating" + di.ToString());

di.Create();

}

else

{

// code from earlier

// FULL\_FILE\_PATH = BASE\_REPORT\_DIR + todaysDate4Reports + "REPORT\_SPA";

// delete any files in it

Console.WriteLine("dir " + di.ToString() + " existed deleting prv run");

FileInfo[] fii = di.GetFiles("REPORT\_SPA.xls");

if (fii.GetLength(0) > 0)

{

foreach (FileInfo f in fii)

{

f.Delete();

}

}

}

# endregion clear previous run excel files

#region store spreadsheet if needed

//todaysDate4Reports contains todays date from last load for filename

//string saveAsFileName = @"test graphi 3 copy" + positionType + " seq-" + dr.seq + " report date-" +

// todaysDate4Reports + " close date=" + closeDt + " ticker pairs " +

// dr.x\_ticker + " " + dr.y\_ticker + ".xls";

//string fullFilePath = saveItDir.ToString() + saveAsFileName;

//Console.WriteLine("Saving ss to " + fullFilePath);

///////was excelWorkbook.Save();

excelWorkbook.SaveAs(FULL\_FILE\_PATH, Missing.Value, Missing.Value

, Missing.Value, Missing.Value, Missing.Value

, Microsoft.Office.Interop.Excel.XlSaveAsAccessMode.xlShared

, Microsoft.Office.Interop.Excel.XlSaveConflictResolution.xlLocalSessionChanges

, Missing.Value

, Missing.Value

, Missing.Value

, Missing.Value

);

#endregion store spreadsheet

#region close excel sheet

excelApp.Sheets.Application.Quit();

try

{

excelApp.Workbooks.Close();

}

catch (Exception whathappened)

{

Debug.Print(whathappened.ToString());

}

excelApp.Quit();

#endregion close excel sheet

// string s = Console.ReadLine();

}

}

}