classAttribute[name\_String, type\_String,units\_String] := (\* does "\_String" actually do anything here vs. "\_" alone...not sure, to be det. \*)

Module[{objectAttribute},

objectAttribute["name"] = name;

objectAttribute["type"] = type;

objectAttribute["units"] = units;

getName[classAttribute[objectAttribute]] := objectAttribute["name"];

getType[classAttribute[objectAttribute]] := objectAttribute["type"];

getUnits[classAttribute[objectAttribute]] := objectAttribute["units"];

Return[classAttribute[objectAttribute]]

];

classDataSet[importedData\_, name\_String, sourceFile\_String, id\_] :=

Module[{objectDataSet, attributeVector, attributeList = {} },

objectDataSet["id"] = id;

objectDataSet["name"] = name;

objectDataSet["sourceFile"] = sourceFile;

attributeVector = First[importedData] ;

Module[{i, attributeInfo},

For[i = 1, i  Length[attributeVector], i++,

attributeInfo = StringSplit[attributeVector[[i]],"\_"];

AppendTo[attributeList, classAttribute[attributeInfo[[1]],attributeInfo[[2]], attributeInfo[[3]]]]

]

];

objectDataSet["attributeList"] = attributeList;objectDataSet["instanceList"] = Drop[importedData, 1];

getID[classDataSet[objectDataSet]] := objectDataSet["id"];

getName[classDataSet[objectDataSet]] := objectDataSet["name"];

getSourceFile[classDataSet[objectDataSet]] := objectDataSet["sourceFile"];

getAttributeList[classDataSet[objectDataSet]] := objectDataSet["attributeList"];

getInstanceList[classDataSet[objectDataSet]] := objectDataSet["instanceList"];

getNumAttributes[classDataSet[objectDataSet]] := Length[objectDataSet["attributeList"]];

getNumInstances[classDataSet[objectDataSet]] := Length[objectDataSet["instanceList"]];

getAttribute[classDataSet[objectDataSet], index\_] :=

If[index > 0 &&index  getNumAttributes[classDataSet[objectDataSet]],

objectDataSet["attributeList"][[index]]

];

getInstance[classDataSet[objectDataSet], index\_] :=

If[index > 0 &&index  getNumInstances[classDataSet[objectDataSet]],

objectDataSet["instanceList"][[index]]

];

Return[classDataSet[objectDataSet]]

];

classMainWindow[] :=

Module[{objectMainWindow, w, h, dataSetList, dataSetTableData, plotViewerList, ds1},

(\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*)

(\*\*\*\* Initialize variables \*\*\*\*)

(\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*)

w = Dynamic[CurrentValue[Notebook, WindowSize][[1]]];

h = Dynamic[CurrentValue[Notebook, WindowSize][[2]]];

dataSetList = {};

dataSetTableData ={};

plotViewerList ={};

(\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*)

(\*\*\*\* Method Definitions \*\*\*\*)

(\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*)

createDefaultMenuBar[] :=

Panel[

Row[{

ActionMenu["File", {"Load Data..."  loadDataSet[],Delimiter, "Exit"  Print[11]}, Appearance "Button", ImageSize{80,25}],

ActionMenu["Plot", {"New 2D Cartesian..."  create2DPlot[], "New 3D Cartesian..."  Print[11]}, Appearance "Button",ImageSize{80,25}]

}, ImageSize Scaled[0.5]

], ImageSizeFull, BackgroundGray

];

createMainMetadataSection[] :=

Panel[

Column[{

createLoadDataButton[],

createDataSetTable["Loaded Data Sets", {"ID", "Name", "Source File", "# Attributes", "# Instances"}, dataSetList],

createPlotViewerTable["Current Plot Viewer Windows", {"ID", "Dataset Name", "Plot Type"}, objectMainWindow["plotViewerList"]]

}, Left,Spacings{Automatic, 33}

], ImageSize{0.9\*w, 0.9\*h}, Appearance {"Frameless"}, BackgroundLightGray

];

createLoadDataButton[] :=

Row[{

Button[Labeled[, "Load Data",Right], (loadDataSet[]), ImageSize{130,45}]

}];

createDataSetTable[tableName\_, columnNames\_, rowData\_] :=

Module[{},

AppendTo[dataSetTableData, getTableHeaders[columnNames]];

Row[{

Panel[

Labeled[

Pane[

Dynamic[Grid[dataSetTableData, FrameAll, ItemSizeScaled[0.247]

]], ScrollbarsTrue, ImageSize {0.85\*w, 0.2\*h}

], getStyledTableName[tableName], Top

], ImageSize{0.87\*w, 0.3\*h}

]

}]

];

createPlotViewerTable[tableName\_, columnNames\_, rowData\_] :=

Module[{tableData},

tableData = {getTableHeaders[columnNames]};

tableData= AppendTo[tableData, getPlotViewerTableInfo[rowData]];

Row[{

Panel[

Labeled[

Pane[

Dynamic[Grid[tableData, FrameAll, ItemSizeScaled[0.247]

]], ScrollbarsTrue, ImageSize{0.85\*w, 0.2\*h}

],getStyledTableName[tableName], Top

], ImageSize{0.87\*w, 0.3\*h}

]

}]

];

getStyledTableName[name\_] :=

Text[name, BaseStyle{Large, Bold}];

getTableHeaders[columnNames\_] :=

Module[{i, headers = {} },

For[i = 1, i  Length[columnNames], i++,

AppendTo[headers, Text[columnNames[[i]], BaseStyle{Bold, Larger}]]

];

Return[headers]

];

getDataSetTableInfo[dataSet\_] :=

Module[{},

AppendTo[dataSetTableData, {getID[dataSet], getName[dataSet], getSourceFile[dataSet], getNumAttributes[dataSet], getNumInstances[dataSet]}]

];

(\* To be completed once further along with PlotViewer module \*)

getPlotViewerTableInfo[dataSetList\_] :=

Return[{}];

loadDataSet[] :=

Module[{sourceFile, dataSetName, dataSet, data},

sourceFile = chooseFile[];

dataSetName = inputDataSetName[Length[dataSetList]+1]; (\* Length[dataSetList] parameter used atm b/c of temp. fix to InputString issue \*)

data = Import[sourceFile];

dataSet = classDataSet[data, dataSetName, sourceFile, Length[dataSetList]+1];

AppendTo[dataSetList, dataSet];

AppendTo[dataSetTableData, getDataSetTableInfo[dataSet]];

];

(\* Need to diagnose why SystemDialogInput["FileOpen"] doesn't work correctly, or find alternative method \*)

chooseFile[] :=

Module[{file},

(\* below line should work but doesn't why ???? \*)

(\* file = SystemDialogInput["FileOpen"]; \*)

file = "C:\\Users\\Matthew\\Documents\\Spring 2014 Classes\\CSCE 470\\Software Development Project Stuff\\Mathematica Stuff\\Test Format CSV Files\\Test2.csv";

Return[file]

];

(\* Need to diagnose why InputString[] doesn't work correctly, or find alternative method \*)

inputDataSetName[id\_] :=

Module[{},

(\* below line doesn't work for some reason \*)

(\* Return[InputString["Enter data set name:"]]; \*)

Return["dataSet" <> ToString[id]]

];

(\* This will create an object2DPlotViewer when that module is complete \*)

create2DPlot[] :=

Module[{},

CreateWindow[DocumentNotebook[Panel["coming soon..."]]]

];

(\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*)

(\*\*\*\* "Main"/Initialization Code \*\*\*\*)

(\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*)

Deploy[

Panel[

Column[{

createDefaultMenuBar[],

createMainMetadataSection[]

},ItemSizeScaled[1.0]

], ImageSize{0.9\*w, 0.9\*h}, BackgroundLightGray

]

]

];

(\* then “run”/“make an instance of” classMainWindow[] like this: \*)

**x = classMainWindow[]**