

```

1  classAttribute[name_String, type_String,units_String] := (* does "_String" actually do anything here vs. "_" alone...not sure, to be det. *)
2  Module[{objectAttribute},
3    objectAttribute["name"] = name;
4    objectAttribute["type"] = type;
5    objectAttribute["units"] = units;
6
7    getName[classAttribute[objectAttribute]] := objectAttribute["name"];
8    getType[classAttribute[objectAttribute]] := objectAttribute["type"];
9    getUnits[classAttribute[objectAttribute]] := objectAttribute["units"];
10
11    Return[classAttribute[objectAttribute]]
12  ];
13
14
15
16  classDataSet[importedData_, name_String, sourceFile_String, id_] :=
17    Module[{objectDataSet, attributeVector, attributeList = {} },
18
19      objectDataSet["id"] = id;
20      objectDataSet["name"] = name;
21      objectDataSet["sourceFile"] = sourceFile;
22
23      attributeVector = First[importedData] ;
24      Module[{i, attributeInfo},
25        For[i = 1, i ≤ Length[attributeVector], i++,
26          attributeInfo = StringSplit[attributeVector[[i]], "_"];
27          AppendTo[attributeList, classAttribute[attributeInfo[[1]],attributeInfo[[2]], attributeInfo[[3]]]]
28        ]
29      ];
30
31      objectDataSet["attributeList"] = attributeList;objectDataSet["instanceList"] = Drop[importedData, 1];
32
33      getID[classDataSet[objectDataSet]] := objectDataSet["id"];
34      getName[classDataSet[objectDataSet]] := objectDataSet["name"];
35      getSourceFile[classDataSet[objectDataSet]] := objectDataSet["sourceFile"];
36      getAttributeList[classDataSet[objectDataSet]] := objectDataSet["attributeList"];
37      getInstanceList[classDataSet[objectDataSet]] := objectDataSet["instanceList"];
38      getNumAttributes[classDataSet[objectDataSet]] := Length[objectDataSet["attributeList"]];
39      getNumInstances[classDataSet[objectDataSet]] := Length[objectDataSet["instanceList"]];
40      getAttribute[classDataSet[objectDataSet], index_] :=
41        If[index > 0 &&index ≤ getNumAttributes[classDataSet[objectDataSet]],
42          objectDataSet["attributeList"][[index]]
43        ];
44      getInstance[classDataSet[objectDataSet], index_] :=
45        If[index > 0 &&index ≤ getNumInstances[classDataSet[objectDataSet]],
46          objectDataSet["instanceList"][[index]]
47        ];
48
49      Return[classDataSet[objectDataSet]]
50    ];
51
52
53
54  classMainWindow[] :=
55    Module[{objectMainWindow, w, h, dataSetList, dataSetTableData, plotViewerList, ds1},
56
57      (*****
58      (**** Initialize variables ****)
59      (*****
60
61      w = Dynamic[CurrentValue[Notebook, WindowSize][[1]]];
62      h = Dynamic[CurrentValue[Notebook, WindowSize][[2]]];
63      dataSetList = {};
64      dataSetTableData ={};
65      plotViewerList ={};
66
67
68      (*****
69      (**** Method Definitions ****)
70      (*****
71
72      createDefaultMenuBar[] :=
73        Panel[
74          Row[{
75            ActionMenu["File", {"Load Data..." => loadDataSet[],Delimiter, "Exit" => Print[11]}, Appearance ->"Button", ImageSize->{80,25}],
76            ActionMenu["Plot", {"New 2D Cartesian..." => create2DPlot[], "New 3D Cartesian..." => Print[11]}, Appearance ->"Button",ImageSize->{80,25}]
77          }, ImageSize-> Scaled[0.5]
78        ], ImageSize->Full, Background->Gray
79      ];
80
81
82      createMainMetadataSection[] :=
83        Panel[
84          Column[{
85            createLoadDataButton[],
86            createDataSetTable["Loaded Data Sets", {"ID", "Name", "Source File", "# Attributes", "# Instances"}, dataSetList],
87            createPlotViewerTable["Current Plot Viewer Windows", {"ID", "Dataset Name", "Plot Type"}, objectMainWindow["plotViewerList"]]
88          }, Left,Spacings->{Automatic, 3->3}
89        ], ImageSize->{0.9*w, 0.9*h}, Appearance ->{"Frameless"}, Background->LightGray
90      ];
91
92
93      createLoadDataButton[] :=
94        Row[{
95          Button[Labeled[, "Load Data",Right], {loadDataSet[]}, ImageSize->{130,45}]
96        ]};
97

```

```

98
99 createDataSetTable[tableNames_, columnNames_, rowData_] :=
100 Module[{},
101   AppendTo[dataSetTableData, getTableHeaders[columnNames]];
102
103   Row[{
104     Panel[
105       Labeled[
106         Pane[
107           Dynamic[Grid[dataSetTableData, Frame→All, ItemSize→Scaled[0.247]
108             ], Scrollbars→True, ImageSize→ {0.85*w, 0.2*h}
109           ], getStyledTableName[tableNames], Top
110         ], ImageSize→{0.87*w, 0.3*h}
111       ]
112     }]
113 ];
114
115
116 createPlotViewerTable[tableNames_, columnNames_, rowData_] :=
117 Module[{tableData},
118   tableData = {getTableHeaders[columnNames]};
119   tableData= AppendTo[tableData, getPlotViewerTableInfo[rowData]];
120
121   Row[{
122     Panel[
123       Labeled[
124         Pane[
125           Dynamic[Grid[tableData, Frame→All, ItemSize→Scaled[0.247]
126             ], Scrollbars→True, ImageSize→{0.85*w, 0.2*h}
127           ],getStyledTableName[tableNames], Top
128         ], ImageSize→{0.87*w, 0.3*h}
129       ]
130     }]
131 ];
132
133
134 getStyledTableName[name_] :=
135 Text[name, BaseStyle→{Large, Bold}];
136
137
138 getTableHeaders[columnNames_] :=
139 Module[{i, headers = {} },
140   For[i = 1, i ≤ Length[columnNames], i++,
141     AppendTo[headers, Text[columnNames[[i]], BaseStyle→{Bold, Larger}]]
142   ];
143   Return[headers]
144 ];
145
146
147 getDataSetTableInfo[dataSet_] :=
148 Module[{},
149   AppendTo[dataSetTableData, {getID[dataSet], getName[dataSet], getSourceFile[dataSet], getNumAttributes[dataSet], getNumInstances[dataSet]}]
150 ];
151
152
153 (* To be completed once further along with PlotViewer module *)
154 getPlotViewerTableInfo[dataSetList_] :=
155 Return[{}];
156
157
158 loadDataSet[] :=
159 Module[{sourceFile, dataSetName, dataSet, data},
160   sourceFile = chooseFile[];
161   dataSetName = inputDataSetName[Length[dataSetList]+1]; (* Length[dataSetList] parameter used atm b/c of temp. fix to InputString issue *)
162   data = Import[sourceFile];
163   dataSet = classDataSet[data, dataSetName, sourceFile, Length[dataSetList]+1];
164   AppendTo[dataSetList, dataSet];
165
166   AppendTo[dataSetTableData, getDataSetTableInfo[dataSet]];
167 ];
168
169
170 (* Need to diagnose why SystemDialogInput["FileOpen"] doesn't work correctly, or find alternative method *)
171 chooseFile[] :=
172 Module[{file},
173   (* below line should work but doesn't why ??? *)
174   (* file = SystemDialogInput["FileOpen"]; *)
175   file = "C:\\Users\\Matthew\\Documents\\Spring 2014 Classes\\CSCE 470\\Software Development Project Stuff\\Mathematica Stuff\\Test Format CSV Files\\Test2.csv";
176   Return[file]
177 ];
178
179
180 (* Need to diagnose why InputString[] doesn't work correctly, or find alternative method *)
181 inputDataSetName[id_] :=
182 Module[{},
183   (* below line doesn't work for some reason *)
184   (* Return[InputString["Enter data set name:"]]; *)
185   Return["dataSet" <> ToString[id]]
186 ];
187
188
189 (* This will create an object2DPlotViewer when that module is complete *)
190 create2DPlot[] :=
191 Module[{},
192   CreateWindow[DocumentNotebook[Panel["coming soon..."]]]
193 ];
194

```

```
195
196      (*****)
197      (**** "Main"/Initialization Code ****)
198      (*****)
199
200      Deploy[
201        Panel[
202          Column[{
203            createDefaultMenuBar[],
204            createMainMetadataSection[]
205          },ItemSize→Scaled[1.0]
206        ], ImageSize→{0.9*w, 0.9*h}, Background→LightGray
207      ]
208    ]
209
210
211  ];
212
213  (* then "run"/"make an instance of" classMainWindow[] like this: *)
214  x = classMainWindow[]
```