File Explorer Document

Ali Haisam Muhammad Rafid
1405013

April 28, 2017

Classes

i) Controller

This class is a controller of the fxml file that we generated from the scene builder. A controller class processes all the elements added in scene builder before calling the method initialize.

1. `public void initialize(URL location, ResourceBundle resources)`
   A overridden method from the class Initializable. This method is called to initialize the controller class after its root element has been completely processed.
2. `private void setFileTableView(FileInfo fileInfo)`
   This method is used to populate the table with child of FileInfo class which is passed as the parameter.
3. `private void setFileTilesView(FileInfo fileInfo)`
   This method is used to populate the TilePane with child of FileInfo class which is passed as the parameter.
4. `private void makeTree()`
   This method is used to construct the tree from root.
5. `private void setTableColumns()`
   Used to set up the table attributes that should be shown from the class FileInfo.
6. `private void setTableAction()`
   Sets the action that should be done when a element in the table is double clicked.
7. `private void setTilesAction(FileInfo fileInfo, VBox vBox)`
   Sets the action that should be done when a element in the tile view is double clicked.
8. `private void setTreeAction()`
   Sets the action that should be done when a tree element is double clicked.
9. `public ObservableList<FileInfo> getChildFiles(FileInfo file)`
   Gets the list of children of the file.

ii) FileInfo

This class stores all the informations about a file including a list of child files.

1.public FileInfo(String fileAbsolutePath)
Constructor which only sets the absolute path. Used to make the fake root of the tree.

2.public FileInfo(File file)
Constructor

3.public String getFileAbsolutePath()

4.public void setFileAbsolutePath(String fileAbsolutePath)

5.public File getFile()

6.public void setFile(File file)

7.public String getFileName()

8.public void setFileName(String fileName)

9.public String getFileModifiedDate()

10.public void setFileModifiedDate(String fileModifiedDate)

11.public long getFileSize()

12.public void setFileSize(long fileSize)

13.public void setChildFiles(ObservableList<FileInfo> childFiles)

14.public ObservableList<FileInfo> getChildFiles()

15.public ImageView getFileImage()

16.public void setFileImage(ImageView fileImage)

17.public String toString()
Overridden method to get the string while constructing tree.

Methods 3-16 are getter and setter methods which are self describing

Design Patterns

1.Composite Pattern:

FileInfo: This class contains a list of its own type.

TreeItem: This class also contains a list of TreeItem types.

2.Adapter Pattern:

VBox: We cannot add an image to a TilePane by itself. So we add the image and file name to a VBox type object and add to the TilePane.

Image: We use this class type object to make a image used in Java Swing compatible for use in JavaFX.