OOLE:Project 5

Project Name: Painting Manager using C++

Developer: Ashwath Narayanan Sundaresan

Purpose: A painting manager application which is used to manage and maintain the paintings in an art galary.

**Youtube Demo link: <http://youtu.be/7Z4o_9OIZ-w>**

**Project description:**

**Files included:**

Header Files:

String.h - Declares the String class instance variables, default constructor, copy constructor, parameter constructor, destructor and the instance methods for accessing.

String.cpp – Provides the definition of the String.h.

Node.h - Declares the Node class instance variables, default constructor, copy constructor, parameter constructor, destructor and the instance methods for accessing.

Node.cpp – Provides the definition of the Node.h.

Linkedlist.h - Declares the Linkedlist class instance variables, default constructor, copy constructor, parameter constructor, destructor and the instance methods for accessing.

Linkedlist.cpp – Provides the definition of the Linkedlist.h. Logic for creating a linkedlist for each of the artist.

Painting.cpp – Is an abstract class which specifies the member variables and member functions which is used by the sub class which inherits this class.

Portrait – Is a sub class of the paining class which specifies the portrait details.

Landscape – Is a sub class of the painting class which specifies the Landscape details.

Stilllife – Is a sub class of the painting class which specifies the Stilllife details.

**---------------------------------------------------------------------------------------------------------------------------**

**Using the painting manager:**

**Compile the Main.cpp :**

**g++ PaintingApplication.cpp**

**Executing the painting manager:**

**./a.out**

Once the user executes the application he is prompted with options:

Selecting option 'a':

**Enter the artist first name and last name.**

Selecting option 'p':

**Enter the title without any space,** if the user enters with space the user is prompted again until he enters the title without any space.

Enter other details and finally the painting title is added.

Selecting option 'r':

**Remove a title from the painting manager,** the user needs to provide the id of the painting.

Selecting option 'd':

**Delete an artist from the painting manager,** the user provides the artist first and last name and the entire linked list for the artist is deleted.

Selecting option 'c':

**Copy a title,** the user needs to provide the title and the artist name, a copy of the particular title is created and appended at the last with version:'version number'. The version number is incremented each time when a new copy of the title is created for that title in the artist.

Selecting option 'l':

**List the painting details in the painting manager application.**

Selecting option 'q':

**Quit the painting manager.** Invokes the destructor for all the classes.

Selecting option 'z':

**Clear the console.**

**----------------------------------------------------------------------------------------------------------------------------**