

CS255 – Project 02 – Playlist

Description and Overview

Write a program that simulates a playlist of songs. This will be implemented by a singly-linked list.

The user will manage the playlist from a menu of options shown below.

```
***** PLAYLIST MENU *****
a - Add song
d - Remove song
p - Change position of song
s - Output songs by specific artist
t - Output total time of playlist (in seconds)
o - Output full playlist
c - Clear playlist
q - Quit

Choose option: █
```

The user should then select an option from the menu. The program should verify that a correct menu option has been selected. Once this has been done, the program should process the selection. The operation of the menu is similar to what was done for Project 01 Shopping Cart.

Significant starter code has been provided for this project. Make sure you review this before starting to code.

The following C++ files will be part of this project. Some are complete and some need modification. Please make sure you review all starter code before beginning this project. We will do an overview of these specs during our scheduled class time.

- **PlaylistDriver.cpp**
 - File that contains the `main()` function
 - The only function that needs to be completed is **processMenuChoice**.
 - *Do not modify any other function or includes in this file.*
- **SongNode.h**
 - *This file is COMPLETE – make no modifications to this file.*
 - This file is the declaration and definition of the `SongNode` class.
 - This class creates and accesses `SongNode` objects.
 - The `Playlist`, which is the linked list, will create the linked list of `SongNode` objects.
- **Playlist.h**
 - *This file is COMPLETE – make no modifications to this file.*
 - Class declaration of the `Playlist`, which is the linked-list.
 - Read comments included in this file to for an overview.
- **Playlist.cpp**
 - Starter code has been provided for this.
 - This is primarily the implementation of the `Playlist` class
 - *Some of the member functions have been completed – make sure you review all comments in this file in detail to determine what you are responsible for.*

Submission Requirements:

- UML Diagram for `SongNode` and `Playlist` named **P02UMLLastName**
- Sample runs of your program named **P02SampleRunsLastName**
- Zipped C++ project file named **P02255LastName.zip**
- Put all of this into one .zip file named **P02LastName.zip**

Sample Runs: See “P02 Playlist Sample Runs.docx”

Due Date: See Programming Projects on Course Blackboard Site.