Lab Assignment 05

Inspiring Excellence

Course Code:	**CSE111**
Course Title:	**Programming Language II**
Topic:	**Multi-class Design**
Number of Tasks:	**10**

[You are not allowed to change the driver codes of any of the tasks]

After YouTube Music, Spotify has decided to redesign their Playlist system. However, they decided to **not use arrays** to store their music, instead, they will use OOP concepts to create the new Playlist system. You have been assigned to build the system by using 3 classes (**Song**, **Playlist,** and **SpotifyTester**).
Each song will have the *name of the song*, *artist name*, *length of the song in minutes* and *the next song*. Each playlist will have a *name* and the *first* song. Playlists can contain multiple songs. Both classes will have some features which will be demonstrated in each task.

[You are not allowed to use Array or any built-in libraries for this assignment]

Task 1

Design the **Song** class with *constructor* and *songInfo()* method along with necessary attributes in such a way that it produces the following output.

Driver Code	Output

`public class SpotifyTester {`
`  public static void main(String[] args) {`
`  Song s1 = new Song("Song-A", "Artist-A", 3);`
`  System.out.println("1==========");`
`  s1.songInfo();`
`  System.out.println("2==========");`
`// More lines will be added in this Tester class`
	`1==========`
	`Title: Song-A,  Artist: Artist-A`
	`Length: 3 minutes`
	`2==========`

Task 2

Design the **Playlist** class constructor along with necessary attributes in such a way that it produces the following output.

Driver Code	Output

`  System.out.println("2==========");`
`// Continuation from Task 1`
`  Playlist p1 = new Playlist("First Playlist");`
`  System.out.println("3==========");`
	`2==========`
	`First Playlist created.`
	`3==========`

Task 3

Create *info()* method inside the **Playlist** class to produce the following output.

Driver Code	Output

`System.out.println("3==========");`
`// Continuation from Task 2`
`p1.info();`
`System.out.println("4==========");`
`p1.start = s1;`
`p1.info();`
`System.out.println("5==========");`
`Song s2 = new Song("Song-B", "Artist-B", 4);`
`Song s3 = new Song("Song-C", "Artist-C", 2);`
`p1.start.next = s2;`
`p1.start.next.next = s3;`
`p1.info();`
`System.out.println("6==========");`
	3==========
	First Playlist has the following songs:
	No songs in First Playlist.
	4==========
	First Playlist has the following songs:
	Song-1
	Title: Song-A, Artist: Artist-A
	Length: 3 minutes
	5==========
	First Playlist has the following songs:
	Song-1
	Title: Song-A, Artist: Artist-A
	Length: 3 minutes
	Song-2
	Title: Song-B, Artist: Artist-B
	Length: 4 minutes
	Song-3
	Title: Song-C, Artist: Artist-C
	Length: 2 minutes
	6==========

Task 4

Create *addSong()* method inside the **Playlist** class to produce the following output.

Driver Code	Output

`System.out.println("6==========");`
`// Continuation from Task 3`
`Song s4 = new Song("Song-D","Artist-D",3);`
`Song s5 = new Song("Song-E","Artist-E",4);`
`p1.addSong(s4);`
`p1.addSong(s5);`
`System.out.println("7==========");`
`p1.info();`
`System.out.println("8==========");`
	`6==========`
	`Song-D added to First Playlist.`
	`Song-E added to First Playlist.`
	`7==========`
	`First Playlist has the following songs:`
	`Song-1`
	`Title: Song-A,  Artist: Artist-A`
	`Length: 3 minutes`
	`Song-2`
	`Title: Song-B,  Artist: Artist-B`
	`Length: 4 minutes`
	`Song-3`
	`Title: Song-C,  Artist: Artist-C`
	`Length: 2 minutes`
	`Song-4`
	`Title: Song-D,  Artist: Artist-D`
	`Length: 3 minutes`
	`Song-5`
	`Title: Song-E,  Artist: Artist-E`
	`Length: 4 minutes`
	`8==========`

Task 5

Create *addSong()* [overloaded] method inside the **Playlist** class to produce the following output.

Driver Code	Output

`System.out.println("8==========");`
`// Continuation from Task 4`
`Song s6 = new Song("Song-F","Artist-F",2);`
`Song s7 = new Song("Song-G","Artist-G",2);`
`Song s8 = new Song("Song-H","Artist-H",4);`
`Song s9 = new Song("Song-I","Artist-I",3);`
`p1.addSong(s6, 0);`
`p1.addSong(s7, 2);`
`p1.addSong(s8, 5);`
`p1.addSong(s9, 10);`
`System.out.println("9==========");`
`p1.info();`
`System.out.println("10==========");`
	8==========
	Song-F added to First Playlist.
	Song-G added to First Playlist.
	Song-H added to First Playlist.
	Cannot add song to Index 10.
	9==========
	First Playlist has the following songs:
	Song-1
	Title: Song-F, Artist: Artist-F
	Length: 2 minutes
	Song-2
	Title: Song-A, Artist: Artist-A
	Length: 3 minutes
	Song-3
	Title: Song-G, Artist: Artist-G
	Length: 2 minutes
	Song-4
	Title: Song-B, Artist: Artist-B
	Length: 4 minutes
	Song-5
	Title: Song-C, Artist: Artist-C
	Length: 2 minutes
	Song-6
	Title: Song-H, Artist: Artist-H
	Length: 4 minutes
	Song-7
	Title: Song-D, Artist: Artist-D
	Length: 3 minutes
	Song-8
	Title: Song-E, Artist: Artist-E
	Length: 4 minutes
	10==========

Task 6

Create *playSong()* method inside the **Playlist** class to produce the following output.

Driver Code	Output

`System.out.println("10==========");`
`//  Continuation from Task 5`
`p1.playSong("Song-F");`
`p1.playSong("Song-Z");`
`p1.playSong("Song-B");`
`System.out.println("11==========");`
	`10==========`
	`Playing Song-F by Artist-F.`
	`Song-Z not found in First Playlist.`
	`Playing Song-B by Artist-B.`
	`11==========`

Task 7

Create the *playSong()* [overloaded] method inside the **Playlist** class to produce the following output.

Driver Code	Output

`System.out.println("11==========");`
`// Continuation from Task 6`
`p1.playSong(0);`
`p1.playSong(4);`
`p1.playSong(10);`
`System.out.println("12==========");`
	`11==========`
	`Playing Song-F by Artist-F.`
	`Playing Song-C by Artist-C.`
	`Song at Index 10 not found in First Playlist.`
	`12==========`

Task 8

Create the *deleteSong()* method inside the **Playlist** class to produce the following output.

Driver Code	Output

`System.out.println("12==========");`
`//  Continuation from Task 7`
`p1.deleteSong("Song-D");`
`p1.deleteSong("Song-B");`
`p1.deleteSong("Song-F");`
`p1.deleteSong("Song-K");`
`System.out.println("13==========");`
`p1.info();`
`System.out.println("14==========");`
	12==========
	Song-D deleted from First Playlist.
	Song-B deleted from First Playlist.
	Song-F deleted from First Playlist.
	Song-K not found in First Playlist.
	13==========
	First Playlist has the following songs:
	Song-1
	Title: Song-A, Artist: Artist-A
	Length: 3 minutes
	Song-2
	Title: Song-G, Artist: Artist-G
	Length: 2 minutes
	Song-3
	Title: Song-C, Artist: Artist-C
	Length: 2 minutes
	Song-4
	Title: Song-H, Artist: Artist-H
	Length: 4 minutes
	Song-5
	Title: Song-E, Artist: Artist-E
	Length: 4 minutes
	14==========

Task 9

Create the *totalSong()* method inside the **Playlist** class to produce the following output.

Driver Code	Output

`System.out.println("14==========");`
`//  Continuation from Task 8`
`System.out.println(p1.name +" has "+p1.totalSong() +" songs");`
`System.out.println("15==========");`
	14==========
	First Playlist has 5 songs
	15==========

Task 10

Create the *merge()* method inside the **Playlist** class to produce the following output.

Driver Code	Output

```
   System.out.println("15==========");
// Continuation from Task 9
   Song ns1 = new Song("Song-Z", "Artist-Z", 3);
   Song ns2 = new Song("Song-Y", "Artist-Y", 4);
   Song ns3 = new Song("Song-X", "Artist-X", 2);
   System.out.println("16==========");
   Playlist p2 = new Playlist("Second Playlist");
   p2.addSong(ns1);
   p2.addSong(ns2);
   p2.addSong(ns3);
   System.out.println("17==========");
   p1.info();
   System.out.println("18==========");
   p2.info();
   System.out.println("19==========");
   p1.merge(p2);
   System.out.println("20==========");
   p1.info();
   System.out.println("21==========");
```

```
15==========
16==========
Second Playlist created.
Song-Z added to Second Playlist.
Song-Y added to Second Playlist.
Song-X added to Second Playlist.
17==========
First Playlist has the following
songs:
Song-1
Title: Song-A,  Artist: Artist-A
Length: 3 minutes
Song-2
Title: Song-G,  Artist: Artist-G
Length: 2 minutes
Song-3
Title: Song-C,  Artist: Artist-C
Length: 2 minutes
Song-4
Title: Song-H,  Artist: Artist-H
Length: 4 minutes
Song-5
Title: Song-E,  Artist: Artist-E
Length: 4 minutes
18==========
Second Playlist has the following
songs:
Song-1
Title: Song-Z,  Artist: Artist-Z
Length: 3 minutes
Song-2
Title: Song-Y,  Artist: Artist-Y
Length: 4 minutes
Song-3
Title: Song-X,  Artist: Artist-X
Length: 2 minutes
19==========
Merge Completed!
20==========
First Playlist has the following
songs:
Song-1
Title: Song-A,  Artist: Artist-A
```

	Length: 3 minutes Song-2 Title: Song-G, Artist: Artist-G Length: 2 minutes Song-3 Title: Song-C, Artist: Artist-C Length: 2 minutes Song-4 Title: Song-H, Artist: Artist-H Length: 4 minutes Song-5 Title: Song-E, Artist: Artist-E Length: 4 minutes Song-6 Title: Song-Z, Artist: Artist-Z Length: 3 minutes Song-7 Title: Song-Y, Artist: Artist-Y Length: 4 minutes Song-8 Title: Song-X, Artist: Artist-X Length: 2 minutes 21==========

Task 11 [Ungraded]

Create the *showHistory()* method inside the **Playlist** class to produce the following output. [Hint: showHistory() only shows the songs which were played from the playlist. So you might need to update the method which is used to play Songs.]

Driver Code	Output

`System.out.println("21==========");`
`//  Continuation from Task 10`
`p1.showHistory();`
`System.out.println("22==========");`
`p2.showHistory();`
`System.out.println("23==========");`
`}`
`}`
	`21==========`
	`History of First Playlist:`
	`Title: Song-F,  Artist: Artist-F`
	`Length: 2 minutes`
	`Title: Song-B,  Artist: Artist-B`
	`Length: 4 minutes`
	`Title: Song-F,  Artist: Artist-F`
	`Length: 2 minutes`
	`Title: Song-C,  Artist: Artist-C`
	`Length: 2 minutes`
	`22==========`
	`History of Second Playlist:`
	`No songs played in Second Playlist.`
	`23==========`

Complete driver code and expected output:

Driver Code	Output

public class SpotifyTester {
  public static void main(String[] args) {
  Song s1 = new Song("Song-A", "Artist-A", 3);
  System.out.println("1==========");
  s1.songInfo();
  System.out.println("2==========");
  Playlist p1 = new Playlist("First Playlist");
  System.out.println("3==========");
  p1.info();
  System.out.println("4==========");
  p1.start = s1;
  p1.info();
  System.out.println("5==========");
  Song s2 = new Song("Song-B", "Artist-B", 4);
  Song s3 = new Song("Song-C", "Artist-C", 2);
  p1.start.next = s2;
  p1.start.next.next = s3;
  p1.info();
  System.out.println("6==========");
  Song s4 = new Song("Song-D", "Artist-D", 3);
  Song s5 = new Song("Song-E", "Artist-E", 4);
  p1.addSong(s4);
  p1.addSong(s5);
  System.out.println("7==========");
  p1.info();
  System.out.println("8==========");
  Song s6 = new Song("Song-F", "Artist-F", 2);
  Song s7 = new Song("Song-G", "Artist-G", 2);
  Song s8 = new Song("Song-H", "Artist-H", 4);
  Song s9 = new Song("Song-I", "Artist-I", 3);
  p1.addSong(s6, 0);
  p1.addSong(s7, 2);
  p1.addSong(s8, 5);
  p1.addSong(s9, 10);
  System.out.println("9==========");
  p1.info();
  System.out.println("10==========");
  p1.playSong("Song-F");
  p1.playSong("Song-Z");
  p1.playSong("Song-B");
  System.out.println("11==========");
  p1.playSong(0);
  p1.playSong(4);
  p1.playSong(10);
	1==========
	Title: Song-A, Artist: Artist-A
	Length: 3 minutes
	2==========
	First Playlist created.
	3==========
	First Playlist has the following songs:
	No songs in First Playlist.
	4==========
	First Playlist has the following songs:
	Song-1
	Title: Song-A, Artist: Artist-A
	Length: 3 minutes
	5==========
	First Playlist has the following songs:
	Song-1
	Title: Song-A, Artist: Artist-A
	Length: 3 minutes
	Song-2
	Title: Song-B, Artist: Artist-B
	Length: 4 minutes
	Song-3
	Title: Song-C, Artist: Artist-C
	Length: 2 minutes
	6==========
	Song-D added to First Playlist.
	Song-E added to First Playlist.
	7==========
	First Playlist has the following songs:
	Song-1
	Title: Song-A, Artist: Artist-A
	Length: 3 minutes
	Song-2
	Title: Song-B, Artist: Artist-B
	Length: 4 minutes
	Song-3
	Title: Song-C, Artist: Artist-C
	Length: 2 minutes
	Song-4

```
    System.out.println("12==========");
    p1.deleteSong("Song-D");
    p1.deleteSong("Song-B");
    p1.deleteSong("Song-F");
    p1.deleteSong("Song-K");
    System.out.println("13==========");
    p1.info();
    System.out.println("14==========");
    System.out.println(p1.name + " has " +
p1.totalSong() + " songs");
    System.out.println("15==========");
    Song ns1 = new Song("Song-Z", "Artist-Z", 3);
    Song ns2 = new Song("Song-Y", "Artist-Y", 4);
    Song ns3 = new Song("Song-X", "Artist-X", 2);
    System.out.println("16==========");
    Playlist p2 = new Playlist("Second Playlist");
    p2.addSong(ns1);
    p2.addSong(ns2);
    p2.addSong(ns3);
    System.out.println("17==========");
    p1.info();
    System.out.println("18==========");
    p2.info();
    System.out.println("19==========");
    p1.merge(p2);
    System.out.println("20==========");
    p1.info();
    System.out.println("21==========");
    //Ungraded Task
    p1.showHistory();
    System.out.println("22==========");
    p2.showHistory();
    System.out.println("23==========");
  }
}
```

```
Title: Song-D,  Artist: Artist-D
Length: 3 minutes
Song-5
Title: Song-E,  Artist: Artist-E
Length: 4 minutes
8==========
Song-F added to First Playlist.
Song-G added to First Playlist.
Song-H added to First Playlist.
Cannot add song to Index 10.
9==========
First Playlist has the following songs:
Song-1
Title: Song-F,  Artist: Artist-F
Length: 2 minutes
Song-2
Title: Song-A,  Artist: Artist-A
Length: 3 minutes
Song-3
Title: Song-G,  Artist: Artist-G
Length: 2 minutes
Song-4
Title: Song-B,  Artist: Artist-B
Length: 4 minutes
Song-5
Title: Song-C,  Artist: Artist-C
Length: 2 minutes
Song-6
Title: Song-H,  Artist: Artist-H
Length: 4 minutes
Song-7
Title: Song-D,  Artist: Artist-D
Length: 3 minutes
Song-8
Title: Song-E,  Artist: Artist-E
Length: 4 minutes
10==========
Playing Song-F by Artist-F.
Song-Z not found in First Playlist.
Playing Song-B by Artist-B.
11==========
Playing Song-F by Artist-F.
```

	Playing Song-C by Artist-C. Song at Index 10 not found in First Playlist. 12========== Song-D deleted from First Playlist. Song-B deleted from First Playlist. Song-F deleted from First Playlist. Song-K not found in First Playlist. 13========== First Playlist has the following songs: Song-1 Title: Song-A, Artist: Artist-A Length: 3 minutes Song-2 Title: Song-G, Artist: Artist-G Length: 2 minutes Song-3 Title: Song-C, Artist: Artist-C Length: 2 minutes Song-4 Title: Song-H, Artist: Artist-H Length: 4 minutes Song-5 Title: Song-E, Artist: Artist-E Length: 4 minutes 14========== First Playlist has 5 songs 15========== 16========== Second Playlist created. Song-Z added to Second Playlist. Song-Y added to Second Playlist. Song-X added to Second Playlist. 17========== First Playlist has the following songs: Song-1 Title: Song-A, Artist: Artist-A Length: 3 minutes Song-2 Title: Song-G, Artist: Artist-G Length: 2 minutes Song-3

	Title: Song-C, Artist: Artist-C Length: 2 minutes Song-4 Title: Song-H, Artist: Artist-H Length: 4 minutes Song-5 Title: Song-E, Artist: Artist-E Length: 4 minutes 18========== Second Playlist has the following songs: Song-1 Title: Song-Z, Artist: Artist-Z Length: 3 minutes Song-2 Title: Song-Y, Artist: Artist-Y Length: 4 minutes Song-3 Title: Song-X, Artist: Artist-X Length: 2 minutes 19========== Merge Completed! 20========== First Playlist has the following songs: Song-1 Title: Song-A, Artist: Artist-A Length: 3 minutes Song-2 Title: Song-G, Artist: Artist-G Length: 2 minutes Song-3 Title: Song-C, Artist: Artist-C Length: 2 minutes Song-4 Title: Song-H, Artist: Artist-H Length: 4 minutes Song-5 Title: Song-E, Artist: Artist-E Length: 4 minutes Song-6 Title: Song-Z, Artist: Artist-Z Length: 3 minutes Song-7

	Title: Song-Y, Artist: Artist-Y Length: 4 minutes Song-8 Title: Song-X, Artist: Artist-X Length: 2 minutes 21========== History of First Playlist: Title: Song-F, Artist: Artist-F Length: 2 minutes Title: Song-B, Artist: Artist-B Length: 4 minutes Title: Song-F, Artist: Artist-F Length: 2 minutes Title: Song-C, Artist: Artist-C Length: 2 minutes 22========== History of Second Playlist: No songs played in Second Playlist. 23==========