Music Library Organizer(---RETIRED---)

Grade settings: Maximum grade: 100
Disable external file upload, paste and drop external content: Yes
Based on: Music Library Organizer(---RETIRED---)
Run: Yes **Evaluate**: Yes
Automatic grade: Yes

A music enthusiast has a large collection of songs in various formats, such as CDs, digital files, and streaming services. They want to be able to organize their collection in one place and easily search and play their favourite songs. However, their current system of keeping track of their music is disorganized and time-consuming. He approached us to design a software to meet his requirements.

As a software developer, help him automate the above process by writing a Java program.

Component Specification: MusicalLibrary Class

Type (Class)	Attributes	Methods
MusicalLibrary	Map<String,String> songMap	Getter and setter methods for the attribute are included in the code skeleton.

***Note**: Here the songMap, holds the Key as songTitle and Value as artistName.*

Requirement 1: Find the Count of song title for a given artist name

Type (Class)	Methods	Responsibilities
MusicalLibrary	public int **getSongCountByArtistName**(String artistName)	This method accepts **artistName** as an argument and returns the count of song title for that artistName in the songMap. If the artistName is not available in the songMap, then it returns 0. ***Condition**:* **artistName** is case in-sensitive

Requirement 2: Filter the list of song title for the given artist name

Type (Class)	Methods	Responsibilities
MusicalLibrary	public List<String> **filterSongTitleByArtistName**(String artistName)	This method accepts artistName as argument and returns a list of all the songTitle in the music library that are performed by the specified artist. ***Condition:*** **artistName** is case in-sensitive

You are provided with the main method as code template and it is excluded from evaluation.

Note:

- In the Sample Input / Output provided, the highlighted text in bold corresponds to the input given by the user, and the rest of the text represents the output.
- Ensure to follow the object-oriented specifications provided in the question description.
- Ensure to provide the names for the classes, attributes, and methods as specified in the question description.
- Adhere to the code template, if provided.

Sample Input/Output 1:

Enter number of music records to be added:

5

Enter the music records (Song Title: Artist Name):

Thriller:Michael Jackson

The Beatles:The Beatles

Purple Rain:Prince

Abbey Road:The Beatles

Bad:Michael Jackson
Enter the artist name

Michael jackson

The count of songs by Michael jackson is 2

The list of songs by Michael jackson is:

Bad

Thriller

Sample Input/Output 2:

Enter number of music records to be added:

3

Enter the music records (Song Title: Artist Name):

Nevermind:Nirvana

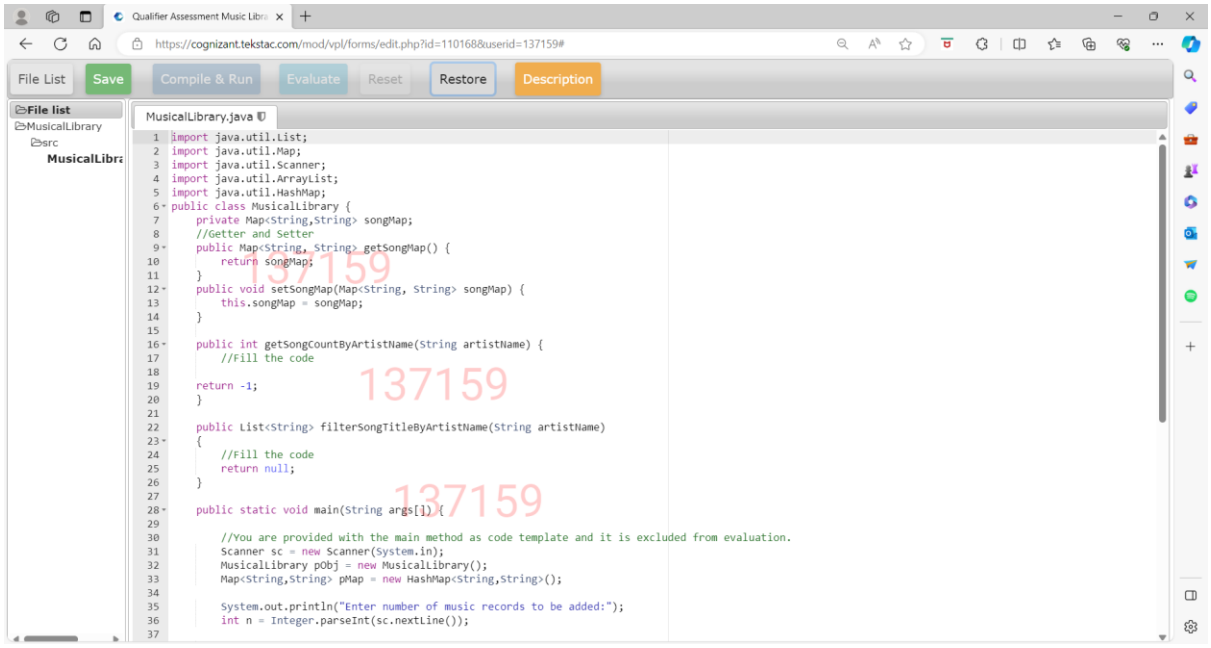
Rumours:Fleetwood Mac

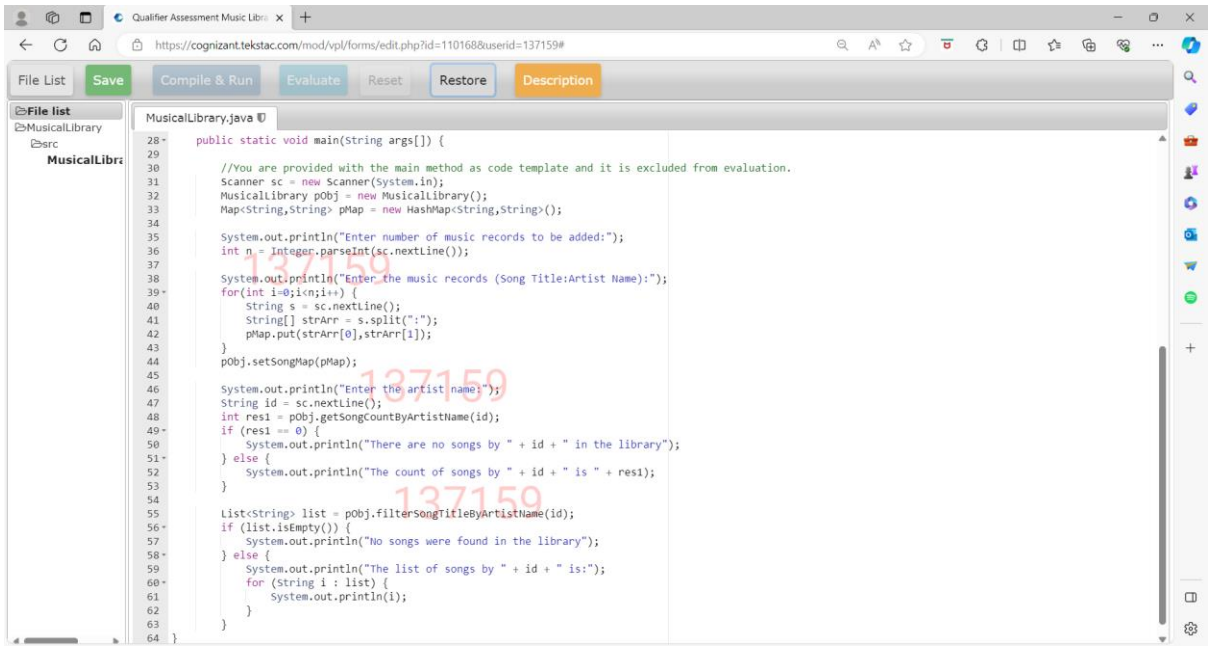
Abbey Road:The Beatles
Enter the artist name

Michael Jackson

There are no songs by Michael Jackson in the library

No songs were found in the library


Qualifier Assessment Music Libra
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=110168&userid=137159#
File List
Save
Compile & Run
Evaluate
Reset
Restore
Description
File list
MusicalLibrary
src
MusicalLibra
MusicalLibrary.java
import java.util.List;
import java.util.Map;
import java.util.Scanner;
import java.util.ArrayList;
import java.util.HashMap;
public class MusicalLibrary {
private Map<String,String> songMap;
//Getter and Setter
public Map<String, String> getSongMap() {
return songMap;
}
public void setSongMap(Map<String, String> songMap) {
this.songMap = songMap;
}
public int getSongCountByArtistName(String artistName) {
//Fill the code
return -1;
}
public List<String> filterSongTitleByArtistName(String artistName)
{
//Fill the code
return null;
}
public static void main(String args[]) {
//You are provided with the main method as code template and it is excluded from evaluation.
Scanner sc = new Scanner(System.in);
MusicalLibrary pObj = new MusicalLibrary();
Map<String,String> pMap = new HashMap<String,String>();
System.out.println("Enter number of music records to be added:");
int n = Integer.parseInt(sc.nextLine());
137159
137159
137159


Qualifier Assessment Music Libra
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=110168&userid=137159#
File List
Save
Compile & Run
Evaluate
Reset
Restore
Description
File list
MusicalLibrary
src
MusicalLibra
MusicalLibrary.java
public static void main(String args[]) {
//You are provided with the main method as code template and it is excluded from evaluation.
Scanner sc = new Scanner(System.in);
MusicalLibrary pObj = new MusicalLibrary();
Map<String,String> pMap = new HashMap<String,String>();
System.out.println("Enter number of music records to be added:");
int n = Integer.parseInt(sc.nextLine());
System.out.println("Enter the music records (Song Title:Artist Name):");
for(int i=0;i<n;i++) {
String s = sc.nextLine();
String[] strArr = s.split(":");
pMap.put(strArr[0],strArr[1]);
}
pObj.setSongMap(pMap);
System.out.println("Enter the artist name:");
String id = sc.nextLine();
int res1 = pObj.getSongCountByArtistName(id);
if (res1 == 0) {
System.out.println("There are no songs by " + id + " in the library");
} else {
System.out.println("The count of songs by " + id + " is " + res1);
}
List<String> list = pObj.filterSongTitleByArtistName(id);
if (list.isEmpty()) {
System.out.println("No songs were found in the library");
} else {
System.out.println("The list of songs by " + id + " is:");
for (String i : list) {
System.out.println(i);
}
}
}
137159
137159
137159