Music Management System(MMS)

Description.

I will Create Music Management System (MMS) in Java. MMS is widely used software. It can be any complexity. Our example is basic one, which have the following features:

1. **MusicTrack** - Represents a music track with fields like title, author, and other relevant information.
2. **MMS** - The main management system class with methods to add, remove, and print information about music tracks.
3. **MMSTester** - A tester class to create music tracks, add them to MMS, and test the functionality.

Here is an outline of the classes:

The class MusicTrack has several fields, including title and author. This class can be implemented in the following way:

package music;

public class MusicTrack {

private String title;

private String author;

public String getTitle() {

return title;

}

public void setTitle(String title) {

this.title = title;

}

public String getAuthor() {

return author;

}

public void setAuthor(String author) {

this.author = author;

}

}

MMS class: The Music management system should have an inner structure for storing music tracks. The management system should have methods for adding the new tracks and removing the old ones. It should have the ability to print the entire content when needed. The class can be implemented in the following way:

package library;

import java.util.ArrayList;

import java.util.List;

public class MMS {

private List<MusicTrack> storage = new ArrayList<>();

public void addMusicTrack(MusicTrack musicTrack) {

storage.add(musicTrack);

}

public boolean removeMusicTrack(MusicTrack musicTrack) {

boolean removed = false;

for (int i = 0; i < storage.size(); i++) {

MusicTrack track = storage.get(i);

if (track.getTitle().equals(musicTrack.getTitle()) && track.getAuthor().equals(musicTrack.getAuthor())) {

storage.remove(i);

removed = true;

break;

}

}

return removed;

}

public void printStorage() {

if (storage.isEmpty()) {

System.out.println("The storage is empty");

} else {

for (MusicTrack track : storage) {

System.out.println(track.getAuthor() + ", " + track.getTitle());

System.out.println();

}

}

}

}

Now let’s test our management system. First, create some music. Then add those tracks to the library. Then try to remove some of them.

MMSTester class:

package music;

public class MMSTester {

public static void main(String[] args) {

MusicTrack track1 = new MusicTrack();

track1.setTitle("Intro");

track1.setAuthor("Psychonaut 4");

MusicTrack track2 = new MusicTrack();

track2.setTitle("Enjoy The Silence");

track2.setAuthor("Depeche Mode");

MMS mms = new MMS();

mms.addMusicTrack(track1);

mms.addMusicTrack(track2);

mms.removeMusicTrack(track1);

mms.printStorage();

}

}

We print the state of the library to check if all the methods are working properly.