## Formative assessment – 2

## NAME:MUTHUWANA M.A.N.R

## INDEX:210400N

## CODE>

**Main Class**

public class Main {  
 public static void main(String[] args) {  
 MainArtist mainArtist = new MainArtist("Taylor Swift");  
 LivePerformance performance=new LivePerformance("Eras Tour",mainArtist,2023,"Glendale");  
 performance.setTrackList("Lavender Haze",10);  
 performance.setTrackList(" Lavender Haze",2);  
 performance.setTrackList("All Too Well",3);  
 performance.setTrackList("The lakes",5);  
 performance.setTrackList("The Man",60);  
 performance.setTrackList("Love Story",45);  
 performance.setBackupSinger("Jeslyn");  
 performance.setBackupSinger("Melanie");  
 performance.setBackupDancer("Stephanie");  
 performance.setBackupDancer("Jake");  
 ////////////////Performance starting//////////////  
 System.*out*.println("///////////////////////////////Performance starting//////////////////////////////////////////////////////");  
 performance.welcomeMessage();  
 performance.recode();  
 mainArtist.sing();  
 performance.backupSingers();  
 mainArtist.dance();  
 performance.backupDances();  
 performance.interact(mainArtist);  
 }  
}

**Performance Class**

import java.util.ArrayDeque;  
import java.util.ArrayList;  
public class Performance {  
 private MainArtist mainArtist;  
 private String performanceName;  
 private int year;  
 private String venue;  
 private ArrayList<MusicTrack> trackList=new ArrayList<>();  
 private ArrayList<Singer> backupSingers=new ArrayList<>();  
 private ArrayList<Dancer> backupDancers=new ArrayList<>();  
  
 Performance(String performanceName,MainArtist mainArtist,int year,String venue){  
 System.*out*.println("This is the constructor Performance() in the class Performance");  
 this.performanceName=performanceName;  
 this.mainArtist=mainArtist;  
 this.year=year;  
 this.venue=venue;  
 }  
 public void welcomeMessage(){  
 System.*out*.println("This is the method welcomeMessage() in the class Performance");  
 System.*out*.println("Welcome to the performance "+this.performanceName+" by "+this.mainArtist.getName());  
 }  
 public void recode(){  
 System.*out*.println("This is the method recode() in the class Performance.");  
 System.*out*.println("Recoding...");  
 }  
 public Artist getMainArtist(){  
 System.*out*.println("This is the method getMainArtist() in the Performance");  
 return this.mainArtist;  
 }  
 //performanceName is viewable and changeable  
 public void setPerformanceName(String performanceName){  
 System.*out*.println("This is the method setPerformanceName() in the class Performance");  
 this.performanceName = performanceName;  
 }  
 public String getPerformanceName(){  
 System.*out*.println("This is the method getPerformanceName() in the class Performance");  
 return this.performanceName;  
 }  
 //year and venue view by anyone but cannot be changed  
 public String getVenue(){  
 System.*out*.println("This is the method getVenue() in the class Performance");  
 return this.venue;  
 }  
 public int getYear(){  
 System.*out*.println("This is the method getYear() in the class Performance");  
 return this.year;  
 }  
 //Anyone can make changes to the trackList ,but the currently existing track list should not be visible to outsiders.  
 public void setTrackList(String name,float duration){  
 MusicTrack musicTrack=new MusicTrack(name,duration);  
 trackList.add(musicTrack);// you can add more methods to the TrackList  
 System.*out*.println("This is the method setTrackList() in the class Performance");  
 }  
 //The number of backup singers and backup dancers varies for each performance.  
 public void setBackupSinger(String name){  
 Singer backupSinger=new Singer(name);  
 backupSingers.add(backupSinger);  
 System.*out*.println("This is the method setBackupSinger() in the class Performance");  
 }  
 public void setBackupDancer(String name){  
 Dancer backupDancer=new Dancer(name);  
 backupDancers.add(backupDancer);  
 System.*out*.println("This is the method setBackupDancer() in the class Performance");  
 }  
 public void backupSingers(){  
 System.*out*.println("This is the method backupSingers() in the class Performance");  
 for(Singer singer:backupSingers){  
 singer.sing();  
 }  
 }  
 public void backupDances(){  
 System.*out*.println("This is the method backupSingers() in the class Performance");  
 for(Dancer dancer:backupDancers){  
 dancer.dance();  
 }  
 }  
}

**LivePerformance Class**

public class LivePerformance extends Performance{  
 LivePerformance(String name,MainArtist mainArtist,int year,String venue){  
 super(name,mainArtist,year,venue);  
 System.*out*.println("This is the constructor LivePerformance() in the class LivePerformance");  
 }  
 @Override  
 public void recode(){  
 System.*out*.println("This is the method recode() in the class LivePerformance.");  
 System.*out*.println("Recording LivePerformance...");  
 }  
 public void interact(Artist artist){  
 System.*out*.println("This is the method interact() in the class LivePerformance");  
 System.*out*.println(artist.getName()+" interact with audience");  
 }  
  
}

**Studio performance Class**

public class StudioPerformance extends Performance{  
 StudioPerformance(String name,MainArtist mainArtist,int year,String venue){  
 super(name,mainArtist,year,venue);  
 System.*out*.println("This is the constructor StudioPerformance() in the class StudioPerformance");  
 }  
 public void audioprocess(){  
 System.*out*.println("This is the method audioprocess in the class StudioPerformance.");  
 System.*out*.println("Audio processing...");  
 }  
 @Override  
 public void recode(){  
 System.*out*.println("This is the method recode() in the class StudioPerformance.");  
 System.*out*.println("Recoding StudioPerformance...");  
 }  
}

**MusicTrack Class**

public class MusicTrack{  
 private String name;  
 private float duration;  
  
 MusicTrack(String name,float duration){  
 System.*out*.println("This is the constructor MusicTrack() in the class Music Track");  
 this.name=name;  
 this.duration=duration;  
 }  
 public void setName(String name){  
 System.*out*.println("This is the method setName() in the class MusicTrack");  
 this.name=name;  
 }  
 public float getDuration(){  
 System.*out*.println("This is the method getDuration() in the class MusicTrack");  
 return duration;  
 }  
 public void setDuration(float duration){  
 System.*out*.println("This is the method setDuration() in the class MusicTrack");  
 this.duration=duration;  
 }  
}

**Ibackup Interface**

public interface IBackup{  
 public void sing();  
 public void dance();  
}

## 

**Artist Class**

public class Artist{  
 private String name;  
  
 Artist(String name){  
 System.*out*.println("This is the constructor Artist() in the class Artist");  
 this.name=name;  
 }  
 public String getName(){  
 System.*out*.println("This is the method getName() in the Artist");  
 return this.name;  
 }  
 void setName(String name){  
 System.*out*.println("This is the method getName() in the Artist");  
 this.name=name;  
 }  
}

**Main Artist Class**

public class MainArtist extends Artist implements IBackup{  
 MainArtist(String name){  
 super(name);  
 System.*out*.println("This is the constructor MainArtist() in the class MainArtist.");  
 }  
 @Override  
 public void sing() {  
 System.*out*.println("This is the method sing() in the class MainArtist.");  
 System.*out*.println(this.getName()+" is Singing");  
 }  
 @Override  
 public void dance() {  
 System.*out*.println("This is the method dance() in the class MainArtist.");  
 System.*out*.println(this.getName()+" is dancing");  
 }  
 @Override  
 public String getName() {  
 System.*out*.println("This is the method getName() in the class MainArtist.");  
 return super.getName();  
 }  
 @Override  
 void setName(String name) {  
 System.*out*.println("This is the method setName() in the class MainArtist.");  
 super.setName(name);  
 }  
}

**Singer Class**

public class Singer extends Artist{  
 Singer(String name){  
 super(name);  
 System.*out*.println("This is the constructor Singer() in the class Singer");  
 }  
 public void sing(){  
 System.*out*.println("This is the method sing() in the class Singer.");  
 System.*out*.println(this.getName()+" is Singing");  
 }  
}

**Dancer Class**

public class Dancer extends Artist{  
 Dancer(String name){  
 super(name);  
 System.*out*.println("This is the constructor Dancer() in the class Dancer");  
 }  
 public void dance(){  
 System.*out*.println("This is the method dance() in the class Dancer().");  
 System.*out*.println(this.getName()+" is Dancing");  
 }  
}