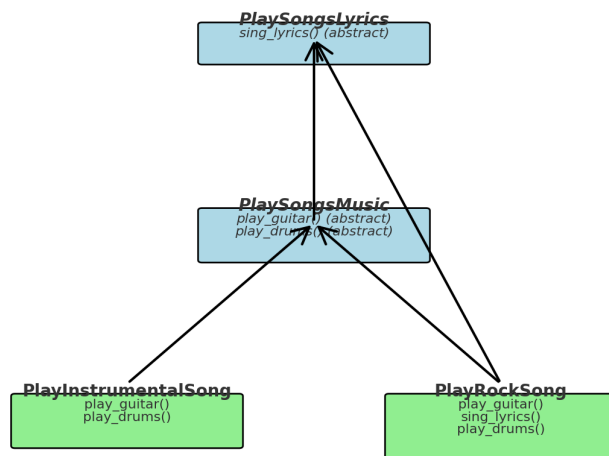


CREATING UML DIAGRAMS: Using code block from OOP lesson 2

Class Diagram is one of the important UML diagrams for software development which shows the object classes in the system and the association between these classes.

Class diagrams is a static type structure diagram that describes the structure of a system by showing the systems classes each class attributes and also the relationship among the objects.



PlaySongsLyrics (Abstract Class, cannot be instantiated on its own. It serves as a blueprint for other classes.)

- Method: `sing_lyrics()` (abstract)

PlaySongsMusic (Abstract Class, the purpose of PlaySongMusic is to serve as a base class for other classes that will implement its abstract methods)

- Methods:
 - `play_guitar()` (abstract)
 - `play_drums()` (abstract)

PlayInstrumentalSong (Implements `PlaySongsMusic`)**

- Methods:
 - `play_guitar()` (implementation)
 - `play_drums()` (implementation)

4. ****PlayRockSong** (Implements `PlaySongsMusic` and `PlaySongsLyrics`)**

- Methods:
 - `play_guitar()` (implementation)
 - `sing_lyrics()` (implementation)
 - `play_drums()` (implementation)

Relationships between these classes

PlayInstrumentalSong and PlayRockSong are subclasses of PlaySongsMusic

PlayRockSong is also a subclass of PlaySongsLyrics

PlaySongsLyrics and PlaySongsMusic are abstract classes, indicating that they only provide method declarations (abstract methods)

UML Diagram Structure

Classes are represented with rectangles divided into three parts: the top for the class name, and the bottom for methods.

Abstract classes and methods are italicized.

Arrows represent inheritance, pointing from the subclass to the superclass.

Here is the UML diagram visualising the class relationships, attributes, and methods based on the provided code: