1. Imagine you were a programmer writing code for a MP3 player. Among other classes, you would likely need some variation of the following classes.
2. An artist class – to store information about the artist who’s songs we will be playing. We will simply store a (String) name for each artist.
3. A song class - where every song has a (String) name and a (String) lyrics, and an (Artist) artist. In a real mp3 program it would have an actual mp3 file associated with it - but we will ignore that part for now and just print lyrics instead of trying to play actual music.

We would need a way to display information about the currently playing song to the user. We will support this by adding a “display” method to the song class. When display is called the song should print out its name and the name of the artist.

Obviously, there would need to be a way to play songs. To support this, we will add a method “play” to our song class. When play is called, the song will be played – which in our context means print the lyrics to the screen.

Create some "Artist"s, and some "Song"s. "Display" the songs and "play" them.

E.g. Write code that will make the following code work as expected

// create an instance of the Artist class passing it a string to use for // the name

Artist bennyF = new Artist("Benny Friedman");

// create an instance of the Song class passing it a string for the name, // a string for the lyrics, and an Artist instance for the Artist

Song toda = new Song("Toda", "Toda...", bennyF (Note: not a string, the Artist bennyF));

calling

toda.display();

would print out

Toda by Benny Friedman

calling

toda.play()

would print out

Toda...

Hint - no static variables or methods (other than main) are appropriate for this exercise. Since we want each artist and song to have its own information all variables should be instance variables and then it would follow that we would have instance methods operating on those instance variables.