

Nama : Muhammad Abdul Aziz

NIM : 2211104026

Kelas : SE0601

Tugas W04 Praktikum Refactoring

Soal

Berdasarkan class Song.Java yang tercantum pada assignment ini lakukan hal-hal berikut ini:

1. Identifikasi bad smell yang ada pada class tersebut!
2. Lakukan refactoring terhadap class Song.Java sesuai dengan bad smell yang dilakukan, silahkan buat method dan class baru jika dibutuhkan!

Jawaban :

1. Primitive Obsession

Penggunaan int untuk genre, yang menyebabkan kode sulit dibaca dan rawan kesalahan karena harus menghafal angka tertentu untuk setiap genre.

```
private int genre;

/**
 * Set the genre of this song
 *
 * 0 = undefined
 * 1 = pop
 * 2 = rock
 * 3 = hip hop
 * 4 = RnB
 * 5 = jazz
 * 6 = instrumentals
 * 7 = cLowncore
 *
 * @param genre
 */
public void setGenre(int genre) {
    this.genre = genre;
}
```

2. Long Parameter List

Konstruktur Song dan metode setAlbum serta setArtist memiliki terlalu banyak parameter.

```
public Song(String id, String title, String releaseYear, String musicFileURL) {  
    this.id = id;  
    this.title = title;  
    this.releaseYear = releaseYear;  
    this.musicFileURL = musicFileURL;  
}  
  
public void setAlbum(String albumName, String albumCoverURL) {  
    this.albumName = albumName;  
    this.albumCoverURL = albumCoverURL;  
}  
  
public void setArtist(String artistName, String artistAlias, String artistImageURL) {  
    this.artistName = artistName;  
    this.artistAlias = artistAlias;  
    this.artistImageURL = artistImageURL;  
}
```

3. Large Class

Kelas Song terlalu besar karena menangani terlalu banyak tanggung jawab (lagu, album, artis).

```
private String albumName;  
private String albumCoverURL;  
private String artistName;  
private String artistAlias;  
private String artistImageURL;
```

4. Long Method

Metode `printInfo` terlalu panjang dan memiliki banyak kondisi `if-else`, membuatnya sulit dibaca dan dipelihara.

```
public void printInfo(int detailLevel) {  
    if (detailLevel == 0) { Replace chain of ifs with switch  
        System.out.println("song title: " + title);  
        System.out.println("release year: " + releaseYear);  
        if (genre > 0) {  
            System.out.println("genre: " + genre);  
        }  
    } else if (detailLevel == 1) {  
        System.out.println("song title: " + title);  
        System.out.println("release year: " + releaseYear);  
        if (genre > 0) {  
            System.out.println("genre: " + genre);  
        }  
        if (!artistName.equals(anObject: "")) {  
            System.out.println("artist name: " + artistName);  
        }  
        if (!artistAlias.equals(anObject: "")) {  
            System.out.println("artist also known as: " + artistAlias);  
        }  
    } else if (detailLevel == 2) {  
        System.out.println("song title: " + title);  
        System.out.println("release year: " + releaseYear);  
        if (genre > 0) {  
            System.out.println("genre: " + genre);  
        }  
        if (!albumName.equals(anObject: "")) {  
            System.out.println("album title: " + albumName);  
        }  
    } else if (detailLevel == 3) {  
        System.out.println("song title: " + title);  
        System.out.println("release year: " + releaseYear);  
        if (genre > 0) {  
            System.out.println("genre: " + genre);  
        }  
        if (!artistName.equals(anObject: "")) {  
            System.out.println("artist name: " + artistName);  
        }  
        if (!artistAlias.equals(anObject: "")) {  
            System.out.println("artist also known as: " + artistAlias);  
        }  
        if (!albumName.equals(anObject: "")) {  
            System.out.println("album title: " + albumName);  
        }  
    }  
}
```

5. Link Github

<https://github.com/akuazizz/MKEPL-PRAKTIKUM-REFACTORING>