```
1  package edu.cpp.cs5800.proxy;
2
3  import org.junit.jupiter.api.Test;
4
5  import static org.junit.jupiter.api.Assertions.assertEquals;
6
7  public class SongTest {
8      Song song = new Song("Blinding Lights", "The Weeknd", "After Hours", 200);
9
10     @Test
11     public void testGetTitle() {
12         assertEquals("Blinding Lights", song.getTitle());
13     }
14
15     @Test
16     public void testGetArtist() {
17         assertEquals("The Weeknd", song.getArtist());
18     }
19
20     @Test
21     public void testGetAlbum() {
22         assertEquals("After Hours", song.getAlbum());
23     }
24
25     @Test
26     public void testGetDuration() {
27         assertEquals(200, song.getDuration());
28     }
29 }
30
```

```
1  package edu.cpp.cs5800.proxy;
2
3  import org.junit.jupiter.api.BeforeAll;
4  import org.junit.jupiter.api.BeforeEach;
5  import org.junit.jupiter.api.Test;
6
7  import static org.junit.jupiter.api.Assertions.assertEquals;
8
9  public class RealSongServiceTest {
10     private SongService songService;
11
12     @BeforeEach
13     void setup() {
14         Song.incrementId = 1;
15         songService = new RealSongService();
16     }
17     @Test
18     public void testSearchById() {
19         String expected = "Song(2, Blinding Lights, The Weeknd, After Hours, 200)";
20         assertEquals(expected, songService.searchById(2).toString());
21     }
22
23     @Test
24     public void testSearchByTitle() {
25         String expected = "[Song(6, Rockstar, Post Malone, Beerbongs & Bentleys, 218)]";
26         assertEquals(expected, songService.searchByTitle("Rockstar").toString());
27     }
28
29     @Test
30     public void testSearchByAlbum() {
31         String expected = "[Song(1, Shape of You, Ed Sheeran, Divide, 240)]";
32         assertEquals(expected, songService.searchByAlbum("Divide").toString());
33     }
34 }
35
```

```java
1  package edu.cpp.cs5800.proxy;
2
3  import org.junit.jupiter.api.BeforeEach;
4  import org.junit.jupiter.api.Test;
5
6  import static org.junit.jupiter.api.Assertions.assertEquals;
7
8  public class ProxySongServiceTest {
9      private SongService songService = new ProxySongService();
10
11     @BeforeEach
12     void setup() {
13         Song.incrementId = 1;
14         songService = new RealSongService();
15     }
16
17     @Test
18     public void testSearchById() {
19         String expected = "Song(2, Blinding Lights, The Weeknd, After Hours, 200)";
20         assertEquals(expected, songService.searchById(2).toString());
21     }
22
23     @Test
24     public void testSearchByTitle() {
25         String expected = "[Song(6, Rockstar, Post Malone, Beerbongs & Bentleys, 218)]";
26         assertEquals(expected, songService.searchByTitle("Rockstar").toString());
27     }
28
29     @Test
30     public void testSearchByAlbum() {
31         String expected = "[Song(1, Shape of You, Ed Sheeran, Divide, 240)]";
32         assertEquals(expected, songService.searchByAlbum("Divide").toString());
33     }
34 }
35
```