

SWINBURNE UNIVERSITY OF TECHNOLOGY

2021 SEM 1 INTRODUCTION TO PROGRAMMING

DOUBTFIRE SUBMISSION

Album File Handling

Submitted By:

Hong Que VO

102240620

2021/05/12 09:52

Tutor:

Tuan Dung LAI

May 12, 2021



```
1  require './input_functions'
2
3  # Task 6.2 T - use the code from 5.1 and 5.2 T to help with this
4
5  module Genre
6      POP, CLASSIC, JAZZ, ROCK = *1..4
7  end
8
9  $genre_names = ['Null', 'Pop', 'Classic', 'Jazz', 'Rock']
10
11 class Album
12     # you will need to add tracks to the following and the initialize()
13     attr_accessor :title, :artist, :genre, :tracks
14
15     # complete the missing code:
16     def initialize (title, artist, genre, tracks)
17         # insert lines here
18         @title = title
19         @artist = artist
20         @genre = genre
21         @tracks = tracks
22     end
23 end
24
25 class Track
26     attr_accessor :name, :location
27
28     def initialize (name, location)
29         @name = name
30         @location = location
31     end
32 end
33
34 # Reads in and returns a single track from the given file
35
36 def read_track(music_file)
37     track_name = music_file.gets.chomp
38     track_location = music_file.gets.chomp
39     returning_track = Track.new(track_name, track_location)
40     return returning_track
41 end
42
43 # Returns an array of tracks read from the given file
44
45 def read_tracks(music_file)
46     tracks = Array.new()
47     count = music_file.gets().to_i()
48
49     # Put a while loop here which increments an index to read the tracks
50
51     while count > 0 do
52         track = read_track(music_file)
53     end
54 end
```

```
54     tracks << track
55     count -= 1
56 end
57     return tracks
58 end
59
60 # Takes an array of tracks and prints them to the terminal
61
62 def print_tracks(tracks)
63
64     index = 0
65     while index < tracks.length
66         print_track(tracks[index])
67         index +=1
68     end
69
70     # print all the tracks use: tracks[x] to access each track.
71 end
72
73
74 def print_track(track)
75     puts('Track title is: ' + track.name)
76     puts('Track file location is: ' + track.location)
77 end
78
79
80 def read_album(music_file)
81
82     # insert lines here - use read_integer_in_range to get a genre
83     puts("Enter Album")
84     album_title = read_string("Enter album name:")
85     album_artist = read_string("Enter artist name:")
86     album_genre = read_integer_in_range("Enter Genre between 1 - 4: ", 1, 4)
87     album_track = read_tracks(music_file)
88     album = Album.new(album_title, album_artist, album_genre, album_track)
89     album.title = album_title
90     album.artist = album_artist
91     album.genre = album_genre
92     album.tracks = album_track
93     return album
94 end
95
96 # Takes a single album and prints it to the terminal
97 # complete the missing lines:
98
99 def print_album(album)
100     puts('Album information is: ')
101     # insert lines here
102     puts(album.title)
103     puts(album.artist)
104     puts('Genre is ' + album.genre.to_s)
105     puts($genre_names[album.genre])
106     print_tracks(album.tracks)
```

```
107 end
108
109 # Takes a single track and prints it to the terminal
110
111
112
113 # Reads in an Album from a file and then prints all the album
114 # to the terminal
115
116 def main()
117     music_file = File.new("album.txt", "r")
118     album = read_album(music_file)
119     #tracks =read_tracks(music_file)
120     music_file.close()
121     print_album(album)
122     #print_tracks(tracks)
123 end
124
125 main()
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
PS D:\COS10009\CODE\LAB6> ruby 6_2P.rb
Enter Album
Enter album name:
hakuta
Enter artist name:
hakata
Enter Genre between 1 - 4:
2
Album information is:
hakuta
hakata
Genre is 2
Classic
Track title is: Crackling Rose
Track file location is: sounds/01-Cracklin-rose.wav
Track title is: Soolaimon
Track file location is: sounds/06-Soolaimon.wav
Track title is: Sweet Caroline
Track file location is: sounds/20-Sweet_Caroline.wav
PS D:\COS10009\CODE\LAB6> █
```