

Overview

In this task, you will have to simulate a **Trip Advisor** like app's backend. There will be a few Cities with a list of TouristPlaces in each.

Defining TouristPlace Class

- Define a **class TouristPlace** with the following data members :
 - name (String)
 - workingHour (String like "10:00AM to 5:00PM")
 - entryTicketPrice (int)
 - noOfRatings (int)
 - starRating (float)
 - famousFor (List<String>)

Defining Constructor & Methods

- Define a **parameterized constructor** passing name, workingHour and entryTicketPrice to it
 - Also, initialize famousFor to new ArrayList<>() in it
 - noOfRatings & starRating will by default be 0.
 - Define a method
void rate(int stars)
and update the noOfRatings & starRating.
 - Define a method
TouristPlace addFamousFor(String s)
and add s to famousFor list. return this; at the end.
 - Just Define an empty method
boolean isPlaceOpen()
We will see the implementation in the next session. return true; for now.
-

Defining City Class

- Define another **class City** with the following data members :
 - name (String)
 - state (String)
 - touristPlaces (List<TouristPlace>)
- Define a parameterized Constructor for **class City** by passing name & state to it.
 - Also, initialize touristPlaces to new ArrayList<>() in it

Driver Code

- In the **Main class**,
 - Create 1 TouristPlace of your choice
 - Invoke touristPlace.addFamousFor()
 - Invoke touristPlace.rate() multiple times
 - Print touristPlace.starRating
-

Submission

- No need to submit, we will code & check LIVE during the sessions from now on.