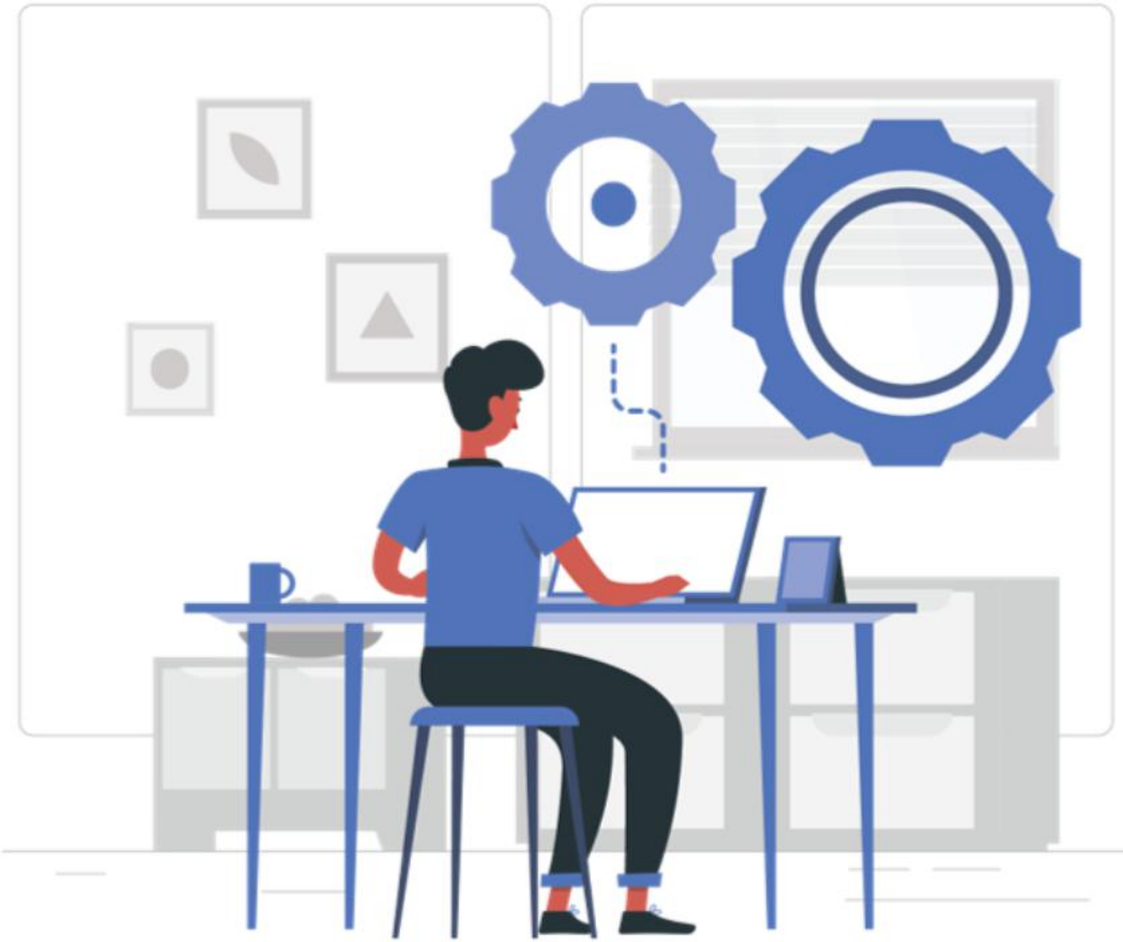
Challenge

Manipulate Objects Using Unordered Collections and Construct Objects as a Key Value Pair



Movie Database

Karry wants to create an application to store movies in a collection object and retrieve movies from the same collection object.
She can also manipulate the movie object.
Help Karry to accomplish the task mentioned in the next slide.

CHALLENGE

Instructions for the Challenge

- Click on the boilerplate.
- Fork the boilerplate using the fork button 

- Select your namespace to fork the project.
- Clone the project into your local system.
- Open the project in the IntelliJ IDE.
- Work on the solution.
- Execute the test cases given in the test folder.
- Push the solution to git.

Tasks

- The class movie has the following attributes
  - movieId, movieName, genre, and releaseDate
- Generate the getter and setter for all the attributes.
- Create a parameterized constructor that has all the attributes.
- Override the toString() method.
- The MovieService class has the following functionalities:
  - All the movies with their ratings will get stored in a Map<Movie,Integer>. The ratings are stored as an integer and must be the value of the map.
  - Find all the movies with a rating of 4 and sort the movie names in alphabetical order.
  - Find all the movies with the highest rating and return the movie object in a list.
  - Return a map of movies with key as movie name and value as releaseDate for all the movies with the genre "comedy“.

Tasks (cont'd)

- Inside the MovieService:
  - Write logic to add all movie objects inside the map object in the method given below and return the map containing the movies and their ratings:
    - Create a movie object by calling the parameterized constructor.
    - Create a map object and store movie objects as keys and ratings of any integer value.
    - The rating can be any number from 1 to 5.

public Map<Movie,Integer> getMovieWithRating()

Tasks (cont'd)

- Inside the MovieService class:
  - Write logic to retrieve all the movies with a rating of four from the given map object.
  - Create a TreeSet object that will store only the movie names with ratings of four.
  - Return the above-created TreeSet object.

```
public Set<String> getMovieNameWithRating4(Map<Movie,Integer> map,int rating){
```

Tasks (cont'd)

- Inside the MovieService class:
  - Write the logic to get all the movies with the highest ratings from the map provided.
  - Store the movie names with the highest ratings in the list and return the list.
  - Use the entrySet() method to retrieve a key and value pair from the map.

```
public List<String> getMovieWithHighestRating(Map<Movie,Integer> map){
```

Tasks (cont'd)

- Inside the MovieService class:
  - Write the logic to get all the movies that are of the “comedy” genre from the map provided.
  - Store the name of the movies and release date where the genre is "comedy" as a key-value pair in the HashMap object.

```
public Map<String,String> getAllMoviesWithComedy(Map<Movie,Integer> map){}
```