1. Solve following Hackerrank challenges:
    a. https://www.hackerrank.com/challenges/java-inheritance-1/problem
    b. https://www.hackerrank.com/challenges/java-inheritance-2/problem
    c. https://www.hackerrank.com/challenges/java-abstract-class/problem
    d. https://www.hackerrank.com/challenges/java-interface/problem
    e. https://www.hackerrank.com/challenges/java-method-overriding/problem

NOTE: For this exercise I suggest to solve the exercise in IntelliJ and then copy solution code into the Hackerrank platform to check it as their site isn't the most developer friendly environment.
My requirement for all **a** to **e** points is to add a printscreen for each bullet with **Solved** status like below.

2*. Write an app for a movie rental inventory system
The goal of this exercise is to design and implement a simple inventory control system for a small video rental store.
Define least two classes: a class Video to model a video and a class VideoStore to model the actual store.
Assume that an object of class Video has the following attributes:

- a title;
- a flag to say whether it is checked out or not; and
- an average user rating.

In addition, you will need to add methods corresponding to the following:

- being checked out;
- being returned; and
- receiving a rating.

The VideoStore class will contain at least an instance variable that references an array of videos (say of length 10).
The VideoStore class has to have the following methods:

- addVideo(String): add a new video (by title) to the inventory;
- checkOut(String): check out a video (by title);
- returnVideo(String): return a video to the store;
- receiveRating(String, int) : take a user's rating for a video; and
- listInventory(): list the whole inventory of videos in the store.

Finally, create a VideoStoreLauncher class with a main() method which will test the functionality of your other two classes. It should allow the following.

Add at least 3 videos: "The Matrix", "Godfather II", "Star Wars Episode IV: A New Hope";
Some testing steps

- Give several ratings to each video.
- Rent each video out once and return it.
- List the inventory after "Godfather II" has been rented out out.

Summary of design specs:

- Store a library of videos identified by title.
- Allow a video to indicate whether it is currently rented out.
- Allow users to rate a video and display the percentage of users that liked the video.
- Print the store's inventory, listing for each video:
  - its title,
  - the average rating, and
  - whether it is checked out or on the shelves.

Note: For storing the videos in the VideoStore class you may use an array of Video (Video[]) or (if you are more courageous) a list of Video (List<Video> videosList = new ArrayList<>(); check the java API for List: https://docs.oracle.com/javase/10/docs/api/java/util/List.html

NOTE: * The marked exercises are not mandatory for this homework, only for those who want to work additionally.