

# Comercia - Java Challenge

## Objective

Evaluate your OO analysis and modeling skills, developing skills, how you structure your code and design an API.

You should not need to spend more than 1 day on the task, if the solution provided is incomplete, explain which parts you decided to focus on and why they are relevant.

Keep the objective in mind, feel free to use any tool, libraries or frameworks, include a README file about the decisions you made, what you focused on and what you did not focus and why, as well as how to run and use the program.

Upload the code to GitHub and send an email with the link to [juan.bastias@globalpay.com](mailto:juan.bastias@globalpay.com)

## The problem - Video game rental system

For a video game rental system we want to create a piece of software for managing the rental administration with three primary functions:

- \* Have an inventory of games
- \* Calculate the price for rental
- \* Keep the track of the customer loyalty points

## Price

The price of rentals is based on the type of the game and how many days it has been rented for.

The users will say when renting the game for how many days they want to rent it and they will pay up front.

If the game is returned late, then rent for extra days will be charged when returning.

The system has three type of games:

- \* New releases - Price is <premium price> times number of days rented.
- \* Standard games - Price is <basic price> for the first 3 days and then <basic price> times the number of days over 3.
- \* Classic game - Price is <basic price> for the first 5 days and then <basic price> times the number of days over 5.

<premium price> 4€

<basic price> 3€

The program should expose a rest-api HTTP API (you could use any framework, if you don't have a personal preference you could use Spring Boot)

The API should (at least) expose the following operations:

- \* Rent one or several games and calculate the price.
- \* Return a game and calculate surcharges (if exist)

## Example of price calculations

No Man's Sky (New release) 1 day -> 4€

Resident Evil 6 (Standard game) 5 days -> 9€

Fallout 4 (Standard game) 2 days -> 3€

Fallout 3 (Classic game) 7 days -> 9€

When returning games late:

No Man's Sky (New release) 2 extra days -> 8€

Resident Evil 6 (Standard game) 1 day extra -> 3€

Total late charge: 11€

## Loyalty points.

Customers get loyalty points when renting a game, a new release gives 2 points, and the other ones 1 points, regardless of the time rented.