

La empresa de video juegos online “VideOL SRL” lo contrata a usted para desarrollar un programa que le permita procesar los archivos que genera con los usuarios gammer’s y los juegos y tiempo de juego que se conectan al sistema de juegos online.

De los gammer’s se registra en un archivo con extensión .csv, denominado “gammers.csv”, los siguientes datos: idJugador, dni, nombre, apellido, alias, plan (Basico, Completo, Extendido), importe base, tiempo limite (en horas) de conexiones según plan.

De las veces que los gammer’s se conectan a algún juego, en el mismo día, se almacenan en un archivo denominado “conexiones.csv”, los siguientes datos: idJugador, dirección IP de conexión, nombre del juego, fecha, hora de inicio, hora de finalización. Tanto la hora de inicio como de finalización, son números enteros de 1 a 24 (corresponden a horas de un mismo día).

Se le pide a usted, que genere las clases:

1. Clase Gammer, que permita registrar el estado de cada objeto de dicha clase, con los atributos correspondientes.
2. Clase Conexion, que permita registrar el estado de cada objeto de dicha clase, con los atributos correspondientes.
3. Una clase Gestor de Gammer, basada en lista de Python, que permita administrar los objetos de la clase Gammer.
4. Una clase Gestor de Conexiones, basada en un arreglo Numpy, que permita administrar los objetos de la clase Conexión.
5. Un menú de opciones que permita:

- a. En el programa principal, leer por teclado el DNI de un Gammer, si no existe emitir un mensaje que advierta de tal situación, si existe, mostrar un listado de las conexiones, con el siguiente formato.

DNI: xx.xxx.xxx

Nombre y apellido: xxxxxxxxxxxxxxxxx

Alias: xxxxxxxx

Plan: xxxxxxxx

Importe base: xxxxx

Ip de conexión	Juego	Fecha	hora inicio	hora fin
xxx.xxx.xxx.xxx	xxxxxxxxxxxxxx	xx/xx/xxxx	xx	xx
xxx.xxx.xxx.xxx	xxxxxxxxxxxxxx	xx/xx/xxxx	xx	xx
xxx.xxx.xxx.xxx	xxxxxxxxxxxxxx	xx/xx/xxxx	xx	xx
xxx.xxx.xxx.xxx	xxxxxxxxxxxxxx	xx/xx/xxxx	xx	xx
xxx.xxx.xxx.xxx	xxxxxxxxxxxxxx	xx/xx/xxxx	xx	xx

Total de horas: xxxx

Horas en exceso: xxxx

Importe a facturar: XXXXXXXX

**Regla de Negocio para el cálculo del importe a facturar:**

Cada hora en exceso, se factura según el plan: 25% el importe base para plan Básico, 30% del importe para plan Completo y 40% para el plan Extendido.

- b. En el programa principal, leer por teclado el nombre de un juego, si no existe, mostrar un mensaje que advierta de tal situación, si existe, mostrar: dirección IP de conexión, nombre y apellido, alias, tipo de plan de los usuarios que han jugado dicho juego.
- c. La empresa, quiere advertir a los gammer’s que utilizan el servicio Basico que no pueden conectarse simultáneamente a los juegos en más de una dirección IP, en la misma fecha y hora de inicio. Para ello le solicita a usted que emita un listado de jugadores con servicio Basico, que se conectan en IP’s distintas, simultáneamente.

**Regla de negocio:** para implementar esta funcionalidad, debe sobrecargar el operador == (\_\_eq\_\_), en la clase Conexión. Se considera que dos conexiones son simultáneas, cuando siendo el mismo idJugador, tienen la misma fecha, la misma hora de inicio y la dirección IP de conexión es distinta.

Para optimizar la funcionalidad también debe sobrecargar el operador mayor que (\_\_lt\_\_) en la clase Conexión, a efectos de ordenar el gestor de Conexiones de menor a mayor por idJugador, fecha, hora de inicio e IP de conexión.

**NOTA:**

**LAS CLASES NO DEBEN ROMPER EL ENCAPSULAMIENTO, NO SE PERMITE OBTENER SUBLISTAS DE LOS GESTORES PARA SOLUCIONAR LAS FUNCIONALIDADES SOLICITADAS.**