

# Backend de Aplicaciones 2025

## Enunciado Simulacro 3K2

Lea atentamente hasta el final del enunciado, a no ser que no le preocupe programar demás, cosas que no se solicitan.

### Datos

Usted dispone del archivo `games_data.csv`, (el archivo tiene encoding UTF-8), con valores separados por ;. El archivo tiene una fila de cabecera con los nombres de cada una de las columnas. El archivo tiene los datos de 60 mil juegos que deseamos procesar para agregarlos a la base de datos.

La estructura del archivo se describe a continuación, **preste atención a las particularidades de cada columna puesto que esto le servirá para evitar idas y vueltas al procesar:**

- **Title**: El nombre del videojuego.
- **Release Date**: La fecha de lanzamiento del videojuego, especificada en formato de año-mes-día.
  - Está expresada como una cadena de caracteres con el formato `MMM dd, yyyy`.

Se espera que usted almacene en la columna de la base de datos el número que se obtiene al invocar el método `getTime()` de las instancias de `java.util.Date`

```
SimpleDateFormat formatter = new SimpleDateFormat("MMM dd, yyyy", Locale.US);

try {
    // Convertir la cadena a un objeto Date
    Date date = formatter.parse(dateString);

    // Obtener el valor en milisegundos usando getTime()
    long timeInMillis = date.getTime();

    // Imprimir el resultado
    System.out.println("Fecha original: " + dateString);
    System.out.println("Fecha en milisegundos: " +
timeInMillis);
} catch (ParseException e) {
    System.out.println("Error al parsear la fecha: " +
e.getMessage());
}
```

- También puede contener la cadena vacía o el valor "TBD", en estos casos simplemente no se cargará este dato.
- **Developers**: El nombre de los desarrolladores del videojuego.

- La lista puede estar vacía y en tal caso contendrá la cadena vacía o la cadena "[]".
- **Summary:** Una descripción general o resumen del videojuego, que puede incluir detalles sobre la trama, características clave y otros aspectos importantes.
  - Es un texto más largo de que una cadena normal.
- **Platforms:** Las plataformas en las que está disponible el videojuego, como PC, PlayStation, Xbox, etc.
  - La lista puede estar vacía y en tal caso contendrá la cadena vacía o la cadena "[]".
- **Genres:** Los géneros del videojuego, como aventura, acción, rol, entre otros.
  - La lista puede estar vacía y en tal caso contendrá la cadena vacía o la cadena "[]".
- **Rating:** La calificación del videojuego, basada en la opinión de los usuarios o críticos.
  - Es un número decimal que está entre 0 y 5 y que expresa el nivel de aceptación del juego.
  - También puede contener la cadena "N/A" y en tal caso simplemente lo vamos a manejar como un valor nulo.
- **Plays:** El número de veces que el videojuego ha sido jugado.
  - Es un número entero pero cargado como cadena de caracteres con la letra K indicando miles.
  - Deberá ser procesado inicialmente como un número decimal para luego en caso de existir la K multiplicarlo por 1000 y volverlo un número entero
- **esrb\_rating:** La clasificación ESRB, que indica la edad recomendada para jugar el videojuego, según el sistema de clasificación de la ESRB.
  - Finalmente la Clasificación que es una cadena de caracteres que requerirá un tratamiento especial de acuerdo con lo que sigue:

## Clasificación ESRB

La clasificación ESRB de los juegos es una clasificación internacional que indica la edad esperada de los jugadores o la edad para la que el juego fue pensado sin tener en cuenta los menores de dicha edad.

En la siguiente tabla se muestran todos los valores posibles con sus códigos:

Código	Clasificación ESRB	Descripción
E	Everyone	Apto para todas las edades. Sin contenido objetable.
E10+	Everyone 10+	Apto para mayores de 10 años. Contiene algo más de fantasía o violencia leve.
T	Teen	Apto para mayores de 13 años. Puede incluir violencia moderada, lenguaje fuerte o temas sugerentes.
M	Mature	Apto para mayores de 17 años. Contiene violencia intensa, sangre, temas sexuales o lenguaje fuerte.
AO	Adults Only	Apto solo para adultos mayores de 18 años. Contiene temas sexuales explícitos o violencia extrema.
RP	Rating Pending	Clasificación aún pendiente de revisión.
UR	Unrated	Clasificación Indeterminada.

Se solicita que esta información se maneje como un enum en la aplicación asociada a la cadena del código que se almacena en la base de datos.

## Base de Datos

La base de datos es la que se suministró previamente con el pre-enunciado

Algunas consideraciones al respecto de la tabla **Juegos** puesto que las acciones a tomar con las demás tablas se definen en el punto 5 de carga de la base de datos:

1. La **fecha\_lanzamiento** está identificada como un entero y es porque se espera que se almacene el valor que se obtiene del método `getTime` de la clase `Date` en ese atributo de forma que si hiciera falta luego se puede volver a fecha de una manera simple.
2. Tanto **rating** como **juegos\_finalizados** se definen como números, ya se mencionó la transformación que se espera para ellos.
3. En el caso de las tablas relacionadas note que siempre vamos a trabajar la relación a una fila de la tabla relacionada, y en el punto 5 de carga de base de datos se explicará como actuar en cada caso.
4. Tenga en cuenta que la columna de género no va a ser tenida en cuenta por lo que usted puede cargarla o no de acuerdo a lo que prefiera.
5. Finalmente, solo para notar la particularidad, las tablas Desarrolladores y Generos son similares por lo que aproveche esta particularidad para ganar tiempo.

## Requerimientos

1. **Cargar el archivo CSV a una estructura de datos en memoria:** Una vez definido el modelo de clases con el mapeo a la base de datos, se espera que usted cree instancias de esas clases a partir de los datos del archivo CSV. Para esto dejamos algunas consideraciones a tener en cuenta:
  - Cargar la línea del archivo CSV si y solo sí tiene al menos un valor para Desarrollador y al menos un valor para Género. [Total de filas a cargar **41464**]
  - Cada linea del archivo CSV se va a transformar en una instancia de la clase **Juego** en memoria que deberá ser almacenada en alguna estructura de datos y en cada instancia de la clase Juego habrá que agregar las instancias de las clases Desarrollador, Género y Plataforma
  - En el caso en que el juego tenga la clasificación vacía se asociará con el valor Unrated que corresponde con Clasificación Indeterminada o No definida.
2. **Determinar el ranking de Géneros más jugados** Determinar los 5 Géneros más jugados teniendo en cuenta la suma de juegos en curso **playing** de todos los juegos del genero.
3. **Determinar la Cantidad de Juegos cargados por cada Desarrollador** Una vez que logró cargar todos los juegos en memoria se requiere que usted muestre un listado de las Desarrolladores diferentes y para cada una de ellos la cantidad de juegos asociados pero solo para los desarrolladores con más de 30 juegos asociados. En este conteo solo cuentan los juegos desarrollados por un único desarrollador. A modo de control, indique también al final la cantidad de todos los juegos que tienen más de un desarrollador.
4. **Determinar cuál es el mejor Desarrollador** Indique cuál es el desarrollador con mayor rating promedio entre todos sus juegos.
5. **Popular la base de datos** con todos los objetos cargados en memoria teniendo en cuenta las siguiente consideraciones:
  - Cada juego en memoria corresponderá con una fila de la tabla **Juegos**

- Cada desarrollador corresponderá con una fila de la tabla **Desarrolladores** y deberán quedar relacionados mediante el id.
- Cada género corresponderá con una fila de la tabla **Generos** y deberán quedar relacionadas mediante el id.
- Cada plataforma corresponderá con una fila de la tabla **Plataformas** y deberán quedar relacionadas mediante el id.
- Finalmente la cadena **Clasificaciones\_ESRB** deberá ser administrada a través de una enumeración en el código Java.

## Recomendaciones y aclaraciones

- Se brindó el material con Proyecto y Base de datos para que el alumno tuviera una visión previa al día del parcial y que no haya sorpresas, la idea de la cátedra es que durante el parcial codifiquen lo que se pide y entreguen cada uno a su leal saber y entender.
- SOLO tienen que escribir código java y configurar la clase que implemente el método main en el plugin del jar para que el programa se pueda ejecutar con **java -jar** ya que hemos quitado todos los elementos de configuración del parcial para eliminar posibles problemas que consumen tiempo valioso en esta instancia
- Por esto:
  - Se espera un trabajo individual y de producción propia, SIN la utilización de bloques previos NI herramientas externas, por esto nos enfocamos en evaluar programación
  - No debe acceder a repositorios ni tampoco a internet y ni usar cualquier tipo de acceso fuera del equipo en el que trabaja puesto que hacerlo habilita al aplazo y análisis de posibles medidas disciplinarias mayores
  - No está permitido el uso IA para la escritura de código más allá de los plugins vistos en clase para trabajo con JPA o base de datos.

La cátedra de Backend les desea todo el Éxito en esta semana de Parcial!!!