

El parque **ElValleExtremo** está compuesto de un conjunto de juegos. Del **Parque** se registra: nombre, razón social, domicilio, la colección de juegos y la colección de los pases emitidos. Para la entrada al parque se emite un **pase** diario, el cual contiene una foto y la información del visitante del parque, id pase, fecha de emisión, cantidadJuegos, y si es o no un pase con aptitud. La variable cantidadJuegos se incrementa cada vez que se usa el pase para subir a un juego. Los datos que se guardan de los **visitantes** son: nombre, apellido, fecha de nacimiento, edad, altura, tipo y número de documento. De los **Juegos** se registra: código del juego, cantidad de personas, si está activo o no (esta condición se actualiza a diario posterior a un chequeo que se realiza de los juegos), edad y altura permitida, cantidad máxima permitida y cantidad total de personas que jugaron ese día, y nombre del juego. Los juegos pueden ser especiales o extremos.

Inicialmente y basados en el modelo relacional informado por el responsable de diseño y desarrollo de la aplicación, el Mapeo Objeto Relacional cuenta con las siguientes clases: **Parque, Pase, Persona, Juego y su jerarquía**. En cada una de las clases implementar los métodos básicos para buscar, ingresar, actualizar y buscar los datos en un motor de base de Datos MySQL utilizando la clase BaseDatos.php proporcionada por la cátedra. Además implementar:

- Método constructor que recibe como parámetros los valores iniciales para los atributos definidos en cada clase.
- Los métodos de acceso de cada uno de los atributos de la clase.
- Redefinir el método **_toString** para que retorne la información de los atributos de la clase.

Las clases principales del **Mapeo Objeto Relacional** tienen una clase a nivel de transacciones de la aplicación que implementan las operaciones que permiten las altas, bajas y modificación de los datos además de la implementación de procedimientos complejos.

- Implementar el ABM (**Alta, Baja, Modificación**) de los **parques** que pueden usar el sistema, inicialmente solo será una por lo que no hace falta implementar métodos de búsqueda.
- Implementar el ABM (**Alta, Baja, Modificación**) de los **visitantes**. Además opciones de búsqueda: por nombre, por número de documento.
- Implementar el ABM (**Alta, Baja, Modificación**) de los **pase**. Además opciones de búsqueda: por fecha de emisión, cantidad de juegos a los que se subió, id pase.
- Implementar el ABM (**Alta, Baja, Modificación**) de los **juegos** del parque. Además opciones de búsqueda de juegos: por código, nombre, si está activo o no.
- Implementar el método **jugarJuego** que recibe como parámetro un objeto pase. Para que la persona portadora del pase pueda jugar el juego **extremo** debe ser un pase de aptitud. No olvide controlar que la cantidad de personas en el juego no sea superior a la permitida, y que la edad y estatura de la persona que posea el pase sea la permitida por el juego. Recordar incrementar la cantidad de personas que están jugando el juego, si la cantidad es igual a la cantidad máxima de personas que pueden jugar ese juego ese día, desactive el juego.
- Implementar el método **juegosMasJugados** para conocer cuáles son los N juegos mas jugados, es decir, aquellos en los que se subió la mayor cantidad de personas. Debe retornar un arreglo asociativo con las siguientes claves: especial, extremo y en cada posición del arreglo contiene otro arreglo con los N juegos más jugados de ese tipo y sus cantidades.
- Implementar el método **juegoMasJugadoXTipo** que recibe por parámetro un tipo de juego y retorna una referencia al juego en el que más gente jugó en el parque.

Implementar un script **appParque** el cual invoca a todos los métodos implementados en las clases transaccionales a partir de un menú de usuario.