Desarrollar un programa Java que permita llevar la gestión de centros de ocio, parques de atracciones y zoológicos, de una empresa de España.

Tanto los **parques de atracciones** como los **zoológicos** poseen un *identificador*, *nombre*, *precioEntrada*, *provincia* , *anyo* de construcción, *valoraciones* de los usuarios que los han visitado (entre 1 y 5). El parque de atracciones tiene un campo adicional que es la edad mínima para entrar. Los zoológicos, guardan información acerca de si cuentan con delfinario o no, y si tienen pingüinos o no. El precio de la entrada al parque de atracciones depende del tipo de centro de ocio y de la edad de la persona que la vaya a comprarla. En el caso del parque de atracciones la entrada será gratis para los mayores de 65 años. Además si el día es festivo valdrá un 40% más la entrada. En el caso del zoológico la entrada será gratis para los niños menores de 7 años y valdrá un 30% más si es festivo. Todos los centros de ocio deben proporcionar un método que permita calcular el precio real de la entrada, teniendo en cuenta la edad del interesado y si es o no festivo. Ambos datos los recibirá el método como argumento de entrada.

Crear una clase **Empresa** que contenga a los centros de ocio y una clase **GestionFicheros** que contenga los métodos para manejar ficheros.

Controlar mediante excepción que el precio de la entrada del parque es entre 15 y 25 y el del zoo entre 10 y 20€. Así mismo controlar mediante excepción que las valoraciones introducidas es un número entre 1 y 5.

Se valorará la utilización de las ventajas de la POO, principalmente el uso de la encapsulación de datos, herencia y polimorfismo, así como la escalabilidad de la aplicación y la interoperabilidad.

Crear una clase Tester que presente un menú al usuario con las siguientes acciones:

1. Listar Centros
2. Añadir centro de Ocio. Permitir añadir tanto parque de atracciones como zoológico. Los datos se cargarán mediante E/S. dentro del método
3. Consulta de centros de ocio. Listar, a partir de la provincia, los centros de ocio cuyo año de construcción sea anterior al año introducido por el usuario. La provincia, así como el año serán introducidos como argumentos de entrada al método
4. Añadir valoración a un centro de Ocio identificado por su identificador. Tanto el identificador como la valoración entrarán como argumentos de entrada al método
5. Actualizar el atributo delfinario que tiene un zoológico identificado por su identificador recibido como argumento de entrada. El método devolverá un booleano informando de si se ha realizado la actualización o no
6. Eliminar centros de ocio cuyo año de construcción sea anterior al introducido por el usuario. El método, dentro de su implementación, preguntará el año de construcción
7. Ordenar los centros de ocio por valoración media.
8. Devolver una colección de los parques de atracciones ordenados por provincia y a igualdad de provincia por precio de entrada.
9. Volcar información a un fichero texto con la información de todos los centros
10. Volcar información a un fichero binario
11. Cargar información a partir del fichero binario sustituyendo los centros que hubiera en RAM
12. Cargar información a partir del fichero de texto, sustituyendo los centros que hubiera en RAM