|  | Trabajo práctico Nº 1- Parte 4  Archivos  Laboratorio de Computación 2 - 2023 C2 |
| --- | --- |

**Enunciado**

Se ha decidido agregar al sistema que gestiona los jugadores, dos archivos para definir más precisamente los equipos y las categorías de los deportes.

El archivo de equipos debe tener el siguiente formato:

**equipos.dat**

| **Campo** | **Tipo** |
| --- | --- |
| Id de equipo | Número entero. Preferentemente autonumérico. |
| Nombre | Cadena de 30 caracteres |
| Nivel | Entero (entre 1 y 3) |
| Estado | bool |

**Restricciones**

* El ID no debe repetirse en el archivo. Preferentemente se debe asignar de manera automática incrementando de a 1 entre registro y registro.
* El nivel es un valor entero cuyos valores son 1 para principiante, 2 para intermedio y 3 para avanzado.
* El estado debe ser verdadero para que el sistema utilice el registro.

**Cambios a realizar en el proyecto**

* Al ingresar un jugador se debe validar que el equipo coincida con una Id de equipo existente en el archivo de equipos.

**MENÚ DE EQUIPOS**

El menú de equipos debe contener las siguientes opciones:

| MENU EQUIPOS  ------------------  1) AGREGAR EQUIPO  2) LISTAR EQUIPO POR ID  3) LISTAR TODO  4) MODIFICAR NOMBRE  5) ELIMINAR REGISTRO  --------------------------------  0) VOLVER AL MENU PRINCIPAL |
| --- |

Aclaraciones:

* La opción AGREGAR deberá permitir ingresar de a un registro por vez.
* La opción LISTAR POR ID deberá solicitar el ID y listar todos los campos del registro si éste existe, o informar si no existe.
* La opción LISTAR TODOS deberá listar todos los registros.
* La opción MODIFICAR NOMBRE deberá pedir un ID y un nombre y cambiar el nombre existente por el nuevo.
* La opción ELIMINAR deberá solicitar el ID y realizar la baja lógica.

El archivo de categorías debe tener el siguiente formato:

**TipodeDeporte.dat**

| **Campo** | **Tipo** |
| --- | --- |
| Id de tipo | Número entero. Preferentemente autonumérico. Entre 1 y 21 |
| Nombre | Cadena de 30 caracteres |
| Nivel de dificultad | Entero entre 1 y 8 |
| Estado | bool |

**Restricciones**

* El ID no debe repetirse en el archivo. Preferentemente se debe asignar de manera automática incrementando de a 1 entre registro y registro.
* El nivel de dificultad es un valor entero entre 1 y 8.
* El estado debe ser verdadero para que el sistema utilice el registro.

**Cambios a realizar en el proyecto**

* Al ingresar un un deporte se debe validar que el Id de tipo coincida con una Id de tipo existente en el archivo de tipos de deportes
* En el menú de configuración se debe agregar las copias de seguridad y restaurar de los archivos agregados

**MENÚ DE TIPOS DE DEPORTE**

El menú de instrumentos debe contener las siguientes opciones:

| MENU TIPOS DE DEPORTE  ------------------  1) AGREGAR TIPO  2) LISTAR TIPO POR ID  3) LISTAR TODO  4) MODIFICAR NIVEL DE DIFICULTAD  5) ELIMINAR REGISTRO  --------------------------------  0) VOLVER AL MENU PRINCIPAL |
| --- |

Aclaraciones:

* La opción AGREGAR deberá permitir ingresar de a un registro por vez.
* La opción LISTAR POR ID deberá solicitar el ID y listar todos los campos del registro si éste existe, o informar si no existe.
* La opción LISTAR TODOS deberá listar todos los registros.
* La opción MODIFICAR NIVEL DE DIFICULTAD deberá pedir un ID y un valor nuevo y cambiar el nivel existente por el nuevo.
* La opción ELIMINAR deberá solicitar el ID y realizar la baja lógica.

.

El sistema debe contar con un **menú principal** con las siguientes opciones:

| MENU PRINCIPAL  ------------------  1) MENU JUGADORES  2) MENU DEPORTES  3) MENU EQUIPOS  4) MENU TIPOS DE DEPORTE  5) REPORTES  6) CONFIGURACION  --------------------------------  0) FIN DEL PROGRAMA  SELECCIONE UNA DE LAS OPCIONES: |
| --- |

Cada opción del menú debe llamar al submenú correspondiente.

El submenú de reportes debe quedar vacío por el momento, allí se realizarán los reportes para el parcial.

Desarrollar un **proyecto de Codeblocks** con las clases y las funciones globales que se considere necesarias.

Se propone que se desarrolle una clase para representar cada una de las entidades (jugadores, deportes, etc) , y otra para cada archivo (estas últimas clases no son obligatorias para el funcionamiento del programa).

Se sugiere que se utilicen funciones globales o métodos para cada una de las acciones que se realizan sobre los archivos.

**RECORDAR QUE CADA CLASE DEBE DEDICARSE SOLO A LOS OBJETOS DE SU CLASE.**