

ACCESO A BASES DE DATOS**EJERCICIO 1**

Se dispone del archivo **corporacion.sql** que crea las tablas necesarias para guardar información sobre empresas, sus empleados, proyectos y demás información para gestionar. Completar los siguientes métodos **HoldingDAO**:

1. El método **agregarEmpleado** que dado los datos de un empleado y el nombre de la empresa a asignar inserta un nuevo empleado en la base de datos.
2. El método **subirSueldo** que sube el sueldo a todos los empleados que pertenecen a una determinada empresa que se pasa como parametro.
3. El método **trasladarEmpleado** que cambia de empresa a un empleado sabiendo su nombre y el de la empresa a la que va a ser trasladado.
4. El método **empleadosEmpresa** que dado el nombre de una empresa devuelve un String con los nombres de todos los empleados de dicha empresa.
5. El método **crearCoche** que dado los datos de un coche y el nombre del empleado a asignárselo lo añade a la tabla de coches.
6. El método **costeProyecto** que dado el nombre de un proyecto nos devuelve un Double con el coste del proyecto, entendido este coste como la suma de los sueldos de los empleados que están asignados a dicho proyecto.
7. El método **resumenProyectos** que devuelve un String con la información de todos los proyectos junto con los nombres de los empleados y el coste asociado de los mismos.
8. El método **empleadosSinCoche** que devuelve un Integer que indica cuántos empleados hay en la empresa que no tienen coche asignado.
9. El método **BorrarProyectosSinEmp** que borra los proyectos de la base de datos que no tienen empleados asignados.
10. El método **BorrarAño** que borra todos los proyectos de un año concreto que se pasa como parametro borrando previamente sus asignaciones de empleados.

EJERCICIO 2

Se dispone del archivo **club.sql** que crea las tablas necesarias con la información sobre los socios y los eventos de un club. Completar los métodos de la clase **ClubDAO**:

1. El método **crearEvento** que dado los datos relativos a un evento lo añade a su tabla.
2. El método **añadirSocio** que dado los datos relativos a un socio lo añade a su tabla.
3. El método **apuntarseEvento** que dado el nombre de un socio y el nombre de un evento

inscribe al socio en dicho evento.

4. El método **eventosSocio** que dado el nombre de un socio devuelve un String con los nombres de los eventos a los que está apuntado.
5. El metodo **resumenEventos** que muestra la informacion de un evento juntos con los nombres de las personas asistentes.
6. El metodo **valoracionesEvento** que dado el nombre de un evento devuelve un String con todos los comentarios de sus valoraciones.
7. El método **eventoMultitudinario** que devuelve el nombre del evento con más asistentes que ha tenido o tendrá el club.
8. El método **sinSocios** que devuelve un String con los nombres de los eventos a los cuales no se ha apuntado ningún socio.
9. El metodo **mejorValorado** que devuelve un String con el nombre del evento mejor valorado tomando como referencia las medias de las puntuaciones de sus reseñas.
10. El metodo **borrarEventos** que borra todos los eventos anteriores a un año concreto.

EJERCICIO 3

Se dispone del archivo **tienda.sql** que crea las tablas con la información de las ventas que se realizan en una tienda online. Completar los siguientes métodos de la clase **TiendaDAO**:

1. El método **añadirVenta** que, a partir del nombre de un producto, un cliente y una cantidad de unidades que se lleva añade una venta a la tabla con la fecha y hora actual.
2. El metodo **comprasCliente** que partiendo del nombre de un cliente devuelve un String con los nombres de cada producto comprado y su cantidad.
3. El metodo **recaudacionTotal** que devuelve un Double con el dinero que ha ingresado la tienda por ventas.
4. El metodo **porCategoria** que devuelve un String con la cantidad de ventas organizadas por categorías de productos.
5. El método **ultimaVenta** que devuelve un String con el nombre del cliente y del producto de la última venta realizada en la tienda.
6. El método **masVendido** devuelve el nombre de producto con mayor cantidad de ventas en total de todos los clientes de la tienda. Incluir el número **de ventas junto al nombre**.
7. El método **sinVentas** que nos devuelve un String con los nombres de los productos de los cuales no se ha realizado ninguna venta todavía.
8. El metodos **borrarProveedor** que dado el nombre de un proveedor lo borra de la base de datos por lo que debiera borrar previamente sus productos.