Todos los ficheros se guardarán en un archivo ZIP llamado: **UF1473\_ACT1\_tuNombre\_tuApellido.ZIP**

**Ejercicio 1**

Crea una transacción bloqueada que permita ingresar simultáneamente 100€ de la cuenta de cada miembro de una colectividad de 10 personas en una cuenta general.

La base de datos se debe llamar **colectividad** y debe tener una única tabla llamada **cuenta** donde debe aparecer el número de cuenta, nombre, email y el saldo de cada cuenta.

El número de cuenta debe ser un campo de tipo VARCHAR(4).

Debe crearse un disparador que cada vez que una cuenta tenga una modificación en el saldo quede reflejado el saldo anterior, el saldo nuevo, la fecha y la hora en que se realizó, el número de cuenta y el nombre.

* Graba el fichero SQL con el nombre de **colectividad.sql**

**Ejercicio 2**

Crea una base de datos llamada **curso** con una tabla llamada **alumnos**. Es esta tabla aparecerá el nombre del alumno, la nota final (entre 0 y 10) y la asistencia (entre 0 y 100).

Realiza a partir de la base de datos anterior una función con POSTGRESQL que aumente un punto en la nota si la asistencia es superior al 90 teniendo en cuenta que nunca puede tener una nota superior a 10, y que reste un punto cuando la asistencia es inferior a 50 teniendo en cuenta que la nota no puede ser inferior a 0.

* Graba el fichero SQL con el nombre de **notas.sql**
* Haz un backup de la base de datos **curso** con el nombre **de curso.backup**.

**Ejercicio 3**

En la base de datos **colectividad** del **ejercicio 1** crea otra tabla llamada **usuarios** con el alias del usuario y su contraseña.

Teniendo en cuenta la tabla **usuarios** crea un fichero PHP con un formulario de autenticación que permita acceder a los registros de la tabla **usuarios**. Cuando se acceda debe mostrar dos listados; uno de la tabla **cuenta** con toda la información de cada miembro y otro de la tabla que contiene todas las modificaciones que sean realizado.

* Haz un backup de la base de datos **colectividad** con el nombre **de colectividad.backup**.