Escribir los programas en JAVA 14 que se indican a continuación, con codificación UTF8:

A partir de los contenidos facilitados, crea dos proyectos cuyo nombre sean tus apellidos **ApellidosEv2Preg1 y ApellidosEv2Preg2**, con tus apellidos obviamente. Por ejemplo ***LopezPerezEv2Preg1***.

Al finalizar el examen, genera un **fichero comprimido con tu nombre .zip .rar .7z** que contenga **ambos proyectos** **y lo subes a la tarea de moodle** y lo envías por correo electrónico a las direcciones [miguel.angel.alvarez@cpilosenlaces.com](mailto:miguel.angel.alvarez@cpilosenlaces.com) **y** [daniel.val@cpilosenlaces.com](mailto:daniel.val@cpilosenlaces.com).

Si tienes tu propia clase de utilidades, cópiala en el proyecto de tu examen y envíala dentro del fichero comprimido de la solución.

**ES RESPONSABILIDAD TUYA ENVIAR CORRECTAMENTE LA SOLUCIÓN. Si no es legible o no cumple lo anteriormente expuesto, el examen no podrá ser corregido y la nota será 1.**

## Primera pregunta: (5 puntos)

A partir del fichero de texto separado por comas EXAMEN.TXT realiza un programa que se llame **Carga.java** para llenar una base de datos en MariaDB o SQLite (la que quieras). Dispones del SQL para cada sistema con el que crear la base de datos.

**Carga.java** **(3 puntos)**.

**Si no consigues** crear la base de datos con el programa anterir, dispones de la base de datos de SQLite **Examen.db** con la base de datos creada y con los datos incluidos en el fichero TXT.

Crea un segundo programa **Gestion.java** en el que te conectes a la base de datos.

La conexión la tienes que leer de un fichero de texto en el que tengas en la primera línea la cadena de conexión a la base de datos, en la segunda el usuario y en la tercera el password. Si necesitas alguna línea más indica en el programa su uso. El programa se tiene que ejecutar independientemente de la base de datos a la que te conectes. **(1 punto)**

El segundo programa (gestión.java) realizará el mantenimiento de la base de datos pasando a objetos y controlando las excepciones,

* Alta de atleta (sin repetidos ver clave primaria) **(1.5 puntos)**
* Baja de atleta (selección por valores, por posición a elección) **(1.5 puntos)**
* Modificación de valores de tabla de referencia (sin repetidos) **(1.5 puntos)**
* Listado con selección: 1)por prueba y marca, 2)por apellidos y nombre **(1.5 puntos)**

## Segunda pregunta: (5 puntos)

Dispones del fichero **JJOOTokio2021.XML** con los datos de deportista y entrenador. También está la clase **Participante.java** (abstracta) y **Principal.java**, que tiene un método static para cargar la lista de participantes si no consigues leer el XML, **SOLO EN ESTE CASO**.

Generar la herencia **(1 punto)**

El programa cargará del XML los datos en la lista **(4 puntos)** o desde el método **(1 punto)**

A partir de la colección cargada hay que generar los 2 siguientes listados (uno a continuación del otro separados por una línea de guiones, y los ficheros tal y como se indica:

* un listado de todos los participantes ordenado por apellidos y nombre **(1 punto)**
* un listado de todos los participantes ordenado por fecha de nacimiento **(1 punto)**
* un fichero de texto por prueba que se llamen **prueba.txt** donde **prueba** será el nombre de la prueba y dentro del fichero estarán primero los entrenadores y luego los atletas ordenados por apellidos y nombre (en un solo bucle todo) **(2 puntos)**
* Sacar un fichero de objetos con toda la colección (**Participantes.dat**) **(1 punto)**