BASES DE DATOS – Parcial 2

Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ UO\_\_\_\_\_\_ PL: \_\_\_\_\_\_\_

**INSTRUCCIONES**

La entrega de las soluciones será en formato PDF con nombre de fichero:

Ex2\_UOXXXXXX\_Nombre\_Apellido1.pdf

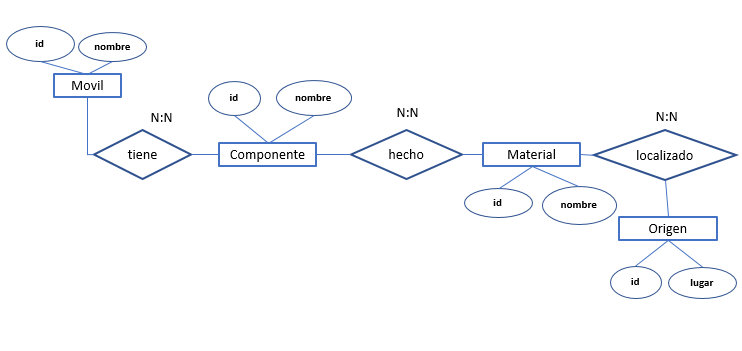
No incluir signos de puntuación en el nombre del fichero tales como acentos o diéresis.

Indicar en el apartado **RESPUESTA** la solución a cada ejercicio. En algunos casos se proporciona la salida esperada como resultado. **Importante:** Obtener la misma salida no es indicativo de obtener la máxima puntuación del ejercicio. Los nombres indicados en negrita de tablas, funciones, procedimientos, triggers, columnas, tuplas, etc. deben respetarse en la solución propuesta o bien son explicativos (en el caso de los parámetros formales de funciones y procedimientos se podrá elegir la denominación que se desee).

Nota: los ejercicios 1 y 2 deberían abordarse en secuencia pues son dependientes.

BD: **movil**

**MODELO E-R de la BD movil**



**Ejercicio 1 (1,5 puntos)**

Crear una tabla aislada del resto de la base de datos **movil** llamada **estandar**, popularizarla y codificar una función que opere sobre ella, todo conforme al siguiente detalle:

* Columnas: **especificacion** (varchar(10) Primary Key); **velocidad** (integer Not Null)
* Popularizar con las tuplas: **(‘GPRS’,50)**; **(‘EDGE’,350)**; **(‘HSPA’,21000)**; **(‘LTE’,60000)**; **(‘LTEA’,100000); (‘NR’,1000000)**
* Crear la función **alo\_sumo\_21000()** que proyecte la tabla **estandar** por la columna **especificacion** incluyendo solo aquellas especificaciones que a lo sumo ofrezcan una velocidad de 21000.

RESPUESTA:

**Ejercicio 2 (2 puntos)**

La tabla **estandar** está aislada en la base de datos **movil** porque la utilizamos como referencia documental acumulada, supongamos que queremos bloquear temporalmente las operaciones de cambio en la citada tabla y procedemos como sigue: codificar el trigger **cambio\_estandar\_trigger** y la función asociada **cambio\_estandar()** que operen antes de un update, delete o insert en la tabla **estandar** impidiendo llevar a término cualquiera de esas operaciones e imprimiendo por consola el mensaje ‘tabla bloqueada: operación no realizada por falta de autorizacion’.

RESPUESTA:

**Ejercicio 3 (2 puntos)**

Codificar el trigger **cambios\_movil\_trigger** y la función asociada **cambios\_movil()** que operen como sigue: tras una operación de inserción, borrado o actualización en la tabla **movil** se imprimirá por consola un mensaje específico que indique el tipo de operación que se ha realizado.

RESPUESTA:

**Ejercicio 4 (2 puntos)**

Codificar un procedimiento con un parámetro **numero\_materiales\_componentel(x)** que opere como sigue: informará del número total de materiales que se asocian con un determinado componente, la invocación del procedimiento se corresponderá, considerando su único parámetro que además será sólo de entrada, con el **id** del componente que se esté consultando. El procedimiento imprimirá por consola el resultado junto con algún texto explicativo.

La siguiente captura se corresponde a la salida de la invocación **numero\_materiales\_componentel(103)**

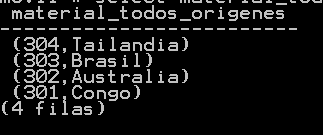


RESPUESTA:

**Ejercicio 5 (2,5 puntos)**

Codificar una función con un parámetro **material\_todos\_origenes(x)** que opere el siguiente resultado: invocándola con el **nombre** de un **material** listará el **id** y **nombre** de todos los orígenes que se asocian con el material en cuestión , lo hará en orden descendente según el **id** de **origen**, finalmente, en el caso de que el material no exista en la base de datos, en relación con la información que se solicita, se imprimirá por consola un mensaje notificándolo.

La siguiente captura se corresponde a la invocación **material\_todos\_origenes(‘coltan’)**



RESPUESTA: