

Ejercicio 01

Sin utilizar procedimientos, copia el nombre de las tablas de una base de datos cualquiera (si no dispones de ninguna, crea una con al menos 3 tablas, 3 atributos por tabla e inserta 3 filas por tabla) en un fichero llamado nombre_tablas.txt:

Ejercicio 02

Para la misma base de datos que hayas elegido en el ejercicio anterior, y sin utilizar procedimientos, crea una base de datos de backup y desde esta, copia la estructura de las tablas y su información.

Ejercicio 03

Para una base de datos con la tabla de país (id (PK), nombre_pais, y habitantes), devuelve un archivo con el nombre de cada país y el porcentaje de habitantes que cada uno tiene sobre el total.

Ejercicio 04

Para una base de datos con la tabla de cuentas bancarias (numero_cuenta (PK), saldo) devuelve en un archivo aquellos números de cuenta que estén en números rojos junto con su saldo correspondiente.

Ejercicio 05

Crea una base de datos llamada ex_plsql y crea la tabla de persona (id (PK), nombre, apellido, apellido2, email, i DNI) e inserta datos para tres personas.

Ejercicio 05 Tarea #1

Crea una tabla para insertar todas las combinaciones posibles de nombre, apellido y segundo apellido de la tabla de persona. Utiliza un procedimiento para ello.



Hint:

Hay que utilizar 3 bucles anidados. En el código siguiente se puede ver un ejemplo de bucles anidados:

```
DECLARE done INT DEFAULT 0;
DECLARE var1 VARCHAR(50);
DECLARE var2 VARCHAR(50);
DECLARE cur1 CURSOR FOR SELECT #SELECT 1r CURSOR;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET done=1;
```

```
OPEN cur1;
primer_loop: LOOP
  FETCH cur1 INTO var1;
  IF done=1 THEN
    LEAVE primer_loop;
  END IF;
BEGIN
  DECLARE cur2 CURSOR FOR SELECT #SELECT 2do CURSOR;
  OPEN cur2;
  segundo_loop: LOOP
    FETCH cur2 INTO var2;
    IF done=1 THEN
      LEAVE segundo_loop;
    END IF;

    #HAZ ALGO

  END LOOP segundo_loop;
  CLOSE cur2;
  SET done = 0;
END;
END LOOP primer_loop;
CLOSE cur1;
```


Ejercicio 05 Tarea #2

Dado un DNI como parámetro de entrada, comprueba si existe el usuario con ese DNI. Devuelve por un parámetro de salida el email de esa persona.

Ejercicio 05 Tarea #3

Dado una id concreta como parámetro de entrada asegúrate que:

- El nombre y apellidos empiezan por una letra mayúscula y el resto son minúsculas.
- El email está en letra minúscula.
- El DNI tiene 8 números y una letra al final. Si falta la letra, añadiremos la letra A por defecto. Si faltan números añadiremos 0's por la izquierda hasta que sea necesario.

Actualiza la fila correspondiente a la id de entrada con los parámetros editados.



Hint:

Nunca vamos a tener más de 8 números y, si tenemos letra, siempre estará en la última posición.

Ejercicio 06

Crea un procedimiento llamado lsCloudLocation donde dada una localización a través de un parámetro de entrada, nos guarde en un archivo el servername y Nick del usuario al que pertenece (1 por línea).

El nombre del fichero tiene que ser: location_NOMBRE-LOCALIZACION_FECHA-ACTUAL.txt (por ej. Irlanda_YYYY-MM-DD.txt).

A parte, se tiene que guardar en un parámetro de salida el número de servidores que hay en esa localización.

En el caso que no exista ningún servidor en la localización designada, hay que escribir en el fichero el texto: 'Localización noválida!'



Hint:

La base de datos y tablas a usar para el ejercicio las podéis encontrar en el documento: ICA0010-S5-C1-D9 Código. Tablas LSCloud.

Si es necesario inserta datos para poder comprobar el funcionamiento del procedimiento.