

Introducción a Bases de Datos y SQL

Resolución de la etapa 4

Resolución de la etapa 4.1

1. Genera, a partir de la tabla **NACIMIENTOS**, una **nueva tabla** con el nombre **VARONES** con todos los campos de la tabla original y copia en esta nueva tabla todos los registros en los que figure el **sexo MASCULINO**.

Al ejecutar la consulta, debería generarse la tabla con **98.397 filas**. No olvides actualizar los esquemas para verificar la creación de la tabla dentro de la base de datos.

```
CREATE TABLE VARONES
SELECT * FROM NACIMIENTOS WHERE SEXO = 'MASCULINO';
```

2. Genera, a partir de la tabla **NACIMIENTOS**, una **nueva tabla** con el nombre **MUJERES** con todos los campos de la tabla original y copia en esta nueva tabla todos los registros en los que figure el **sexo FEMENINO**. Al ejecutar la consulta, debería generarse la tabla con **94.763 filas**. Actualiza los esquemas para verificar la creación de la tabla dentro de la base de datos.

```
CREATE TABLE MUJERES
SELECT * FROM NACIMIENTOS WHERE SEXO = 'FEMENINO';
```



3. Genera, a partir de la tabla **NACIMIENTOS**, una **nueva tabla** con el nombre **INDETERMINADOS** con todos los campos de la tabla original y copia en esta nueva tabla todos los registros en los que figure **INDETERMINADO** como **sexo**. Al ejecutar la consulta, debería generarse la tabla con **24 filas**. No olvides actualizar los esquemas, para verificar la creación de la tabla dentro de la base de datos.

```
CREATE TABLE INDETERMINADOS
SELECT * FROM NACIMIENTOS WHERE SEXO = 'INDETERMINADO';
```

4. **Elimina** la tabla **NACIMIENTOS**.

```
DROP TABLE NACIMIENTOS;
```



Resolución de la etapa 4.2

1. Actualiza la tabla **CLIENTES_NEPTUNO**.
Cuando figure el valor **ESTADOS UNIDOS** en el campo **PAIS**, **cambiar** dicho valor por **USA**.
Para verificar el cambio realizado, muestra el contenido de la tabla **CLIENTES_NEPTUNO**.

```
SET SQL_SAFE_UPDATES = 0;  
UPDATE CLIENTES_NEPTUNO  
SET PAIS = 'USA' WHERE PAIS = 'ESTADOS UNIDOS';  
SELECT * FROM CLIENTES_NEPTUNO;
```



2. Actualiza la tabla **CLIENTES_NEPTUNO** para mostrar todos los valores del campo **NOMBRECOMPANIA** en **mayúsculas**. Para verificar el cambio realizado, muestra el contenido de la tabla **CLIENTES_NEPTUNO**.

```
UPDATE CLIENTES_NEPTUNO  
SET NOMBRECOMPANIA = UPPER(NOMBRECOMPANIA);  
SELECT * FROM CLIENTES_NEPTUNO;
```



3. Actualiza la tabla **CLIENTES_NEPTUNO** para mostrar todos los valores de los campos **CIUDAD** y **PAIS** en **mayúsculas**. De la tabla **CLIENTES_NEPTUNO** muestra el contenido, para verificar el cambio realizado

```
UPDATE CLIENTES_NEPTUNO
SET CIUDAD = UPPER(CIUDAD), PAIS = UPPER(PAIS);
SELECT * FROM CLIENTES_NEPTUNO;
```


4. En la tabla **EMPLEADOS** genera una **nueva columna** llamada **NOMBRE_EMPLEADO** que admita cadenas de texto con un **máximo de 30 caracteres**. Ubica esta columna a la **derecha del campo IDEMPLEADO**. Completa esta nueva columna, **concatenando** los

valores cargados en los campos **APELLIDOS** y **NOMBRE**, separando dichos valores por una **coma y un espacio**. **Elimina** las columnas **APELLIDOS** y **NOMBRE**. Muestra el contenido de la tabla **EMPLEADOS** para verificar el cambio llevado a cabo.

```
ALTER TABLE EMPLEADOS ADD NOMBRE_EMPLEADO VARCHAR(30) AFTER IDEMPLEADO;  
UPDATE EMPLEADOS  
SET NOMBRE_EMPLEADO = CONCAT(APELLIDOS, ', ', NOMBRE);  
ALTER TABLE EMPLEADOS DROP APELLIDOS, DROP NOMBRE;  
SELECT * FROM EMPLEADOS;
```

5. En la tabla **CLIENTES**, genera una **nueva columna** llamada **TIPO** que admita cadenas de texto con un **máximo de 3 caracteres**. Esta columna debe ubicarse **al final** de la tabla. Completa esta nueva columna estableciendo

el **valor VIP** para todos aquellos clientes que residan en la ciudad de **MADRID**. Para verificar el cambio llevado a cabo, muestra el contenido de la tabla **CLIENTES**.

```
ALTER TABLE CLIENTES ADD TIPO VARCHAR(3);  
UPDATE CLIENTES SET TIPO = 'VIP' WHERE CIUDAD = 'MADRID';  
SELECT * FROM CLIENTES;
```

6. Dado que **todos los clientes** que figuran en la tabla **CLIENTES** residen en **España**, **agrega el prefijo +34-** a cada uno de los **teléfonos** que figuran en el campo **TELEFONO**. En caso de

que un cliente no tuviera el teléfono cargado, a ese teléfono no se le debería agregar el prefijo. Muestra el contenido de la tabla **CLIENTES** para verificar el cambio realizado.

```
ALTER TABLE CLIENTES MODIFY TELEFONO VARCHAR(20);  
UPDATE CLIENTES  
SET TELEFONO = CONCAT('+34-', TELEFONO) WHERE TELEFONO IS NOT NULL;  
SELECT * FROM CLIENTES;
```

7. En la tabla **PRODUCTOS** genera una **nueva columna** con el nombre **FECHA** que admita la **carga de fechas**. Completa la nueva columna con fechas coherentes, concatenando los campos **DIA**, **MES** y **ANO**. Cargados los datos en la nueva columna, **elimina las columnas**

originales (DIA, MES y ANO). Además, actualiza el campo **ORIGEN** de manera que, donde figure el **valor ESPANA**, sea **reemplazado por ESPAÑA**. Muestra el contenido de la tabla **PRODUCTOS** para verificar el cambio llevado a cabo.

```
ALTER TABLE PRODUCTOS ADD FECHA DATE;  
UPDATE PRODUCTOS SET FECHA = CONCAT(ANO, '-', MES, '-', DIA);  
ALTER TABLE PRODUCTOS DROP DIA, DROP MES, DROP ANO;  
UPDATE PRODUCTOS SET ORIGEN = 'ESPAÑA'  
WHERE ORIGEN = 'ESPANA';  
SELECT * FROM PRODUCTOS;
```

8. Actualiza el campo **SUSPENDIDO** de la tabla **PRODUCTOS_NEPTUNO**.

```
ALTER TABLE PRODUCTOS_NEPTUNO MODIFY SUSPENDIDO VARCHAR(2);  
UPDATE PRODUCTOS_NEPTUNO  
SET SUSPENDIDO = IF(SUSPENDIDO = '0', 'NO', 'SI');  
SELECT * FROM PRODUCTOS_NEPTUNO;
```

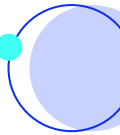


9. Actualiza los **precios de todos los productos** de la tabla ***PRODUCTOS_NEPTUNO***, **recargándolos un 10%** y manteniendo un total de **2 decimales** para cada uno de los precios. Muestra el contenido de la tabla ***PRODUCTOS_NEPTUNO*** para verificar el cambio llevado a cabo.

```
UPDATE PRODUCTOS_NEPTUNO SET PRECIOUNIDAD = ROUND(PRECIOUNIDAD * 1.1, 2);  
SELECT * FROM PRODUCTOS_NEPTUNO;
```

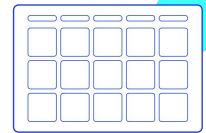
10. Observa la tabla **PROVEEDORES** y su contenido. Para todos aquellos proveedores que **no tengan** cargado un valor en el campo **REGION**, mostrar el **valor NULL** en dicho campo. Muestra el contenido de la tabla **PROVEEDORES** para verificar el cambio llevado a cabo.

```
UPDATE PROVEEDORES SET REGION = NULL WHERE REGION = '';  
SELECT * FROM PROVEEDORES;
```



11. Observa el contenido de la tabla **CLIENTES**. Actualiza el campo **CIUDAD** de manera que **todos los valores** cargados en esta columna muestren la **primera letra en mayúsculas y el resto en minúsculas**. Muestra el contenido de la tabla **PROVEEDORES** para verificar el cambio llevado a cabo.

```
UPDATE CLIENTES
SET CIUDAD = CONCAT(UPPER(LEFT(CIUDAD, 1)),
LOWER(SUBSTRING(CIUDAD, 2, LENGTH(CIUDAD))));
SELECT * FROM CLIENTES;
```



12. Genera una **nueva tabla** con el nombre ***PRODUCTOS_SUSPENDIDOS*** a partir de la **tabla *PRODUCTOS_NEPTUNO***. Vuelca en esta nueva tabla todos los campos de la **tabla *PRODUCTOS_NEPTUNO***, pero **sólo aquellos** registros en los que en el campo ***SUSPENDIDO*** figure la **palabra *SI***. Muestra de la **tabla *PRODUCTOS_SUSPENDIDOS*** el contenido, una vez generada. La nueva tabla debería contener **8 productos** (registros).

```
CREATE TABLE PRODUCTOS_SUSPENDIDOS
SELECT * FROM PRODUCTOS_NEPTUNO
WHERE SUSPENDIDO = 'SI';
SELECT * FROM PRODUCTOS_SUSPENDIDOS;
```



Resolución de la etapa 4.3

1. A partir de este momento, **todos los productos** provistos por el **proveedor 1** pasan a estar **suspendidos** por tiempo indeterminado.

```
UPDATE PRODUCTOS_NEPTUNO SET SUSPENDIDO = 'SI'
WHERE IDPROVEEDOR = 1;

INSERT INTO PRODUCTOS_SUSPENDIDOS (IDPRODUCTO,
NOMBREPRODUCTO, NOMBRECONTACTO, NOMBRECATEGORIA, PRECIOUNIDAD, SUSPENDIDO, IDPROVEEDOR)
SELECT IDPRODUCTO, NOMBREPRODUCTO, NOMBRECONTACTO, NOMBRECATEGORIA, PRECIOUNIDAD,
SUSPENDIDO, IDPROVEEDOR
FROM PRODUCTOS_NEPTUNO
WHERE SUSPENDIDO = 'SI';

DELETE FROM PRODUCTOS_NEPTUNO WHERE SUSPENDIDO = 'SI';
```

Resolución de la etapa 4.4

1. En la etapa anterior, se creó una tabla nueva con el nombre ***PRODUCTOS_SUSPENDIDOS*** y se copió en ella, todos los productos que se encontraban suspendidos de la tabla ***PRODUCTOS_NEPTUNO***. Sin embargo, estos registros se copiaron de una tabla a la otra; es decir, **aún se encuentran cargados en la**

tabla *PRODUCTOS_NEPTUNO*. Por lo tanto, lleva a cabo una consulta de eliminación por la que los ***productos suspendidos se eliminen*** de la tabla ***PRODUCTOS_NEPTUNO***. Al ejecutar esta consulta deberían eliminarse de la tabla ***PRODUCTOS_NEPTUNO*** **8 registros**.

```
SET SQL_SAFE_UPDATES = 0;  
DELETE FROM PRODUCTOS_NEPTUNO  
WHERE SUSPENDIDO = 'SI';
```

**¡Sigamos
trabajando!**