

Tarea UD3

0. Escenario

Al usar Odoo, la parte relacionada con la base de datos queda ofuscada. Esto se debe a la capa de abstracción que añade Odoo.

Crear un módulo y programarlo a bajo nivel es un punto de este curso por lo que se hace necesario dedicar al menos esta unidad didáctica a la base de datos en exclusiva.

En esta guía resolvemos la tarea propuesta para esta UD3.

1.-(20%) Mediante la "Herramienta de consulta" de pgAdmin o mediante CLI, realiza la captura de la sentencia SQL que cree un tabla en la base de datos. Esta tabla debe llamarse como las iniciales de tu nombre y apellidos. Los campos de esta nueva tabla serán:

- **socio:** autonumérico. Este campo debe ser clave primaria.
- **nombre:** texto con un tamaño máximo de 20 caracteres
- **observaciones:** texto largo, muy largo.
- **cantidad:** número entero
- **fechaAlta:** tipo fecha

2.-(20%) Mediante la "Herramienta de consulta" de pgAdmin o mediante CLI, realiza la inserción de 5 registros inventados mediante una sentencia SQL. Realiza la captura de pantalla.

3.-(20%) Realiza una consulta SQL en el que se muestren todos los datos de la tabla aaa ordenados ascendentemente por la fecha de alta. Realiza captura de pantalla de la sentencia SQL y el resultado.

4.-(20%) Realiza una consulta SQL en el que se muestre el nombre del registro con fecha más antigua. Realiza captura de la sentencia y el resultado.

5.-(20%) Usando las tablas de Odoo, crea una consulta que muestre los siguientes datos y captura pantalla de la consulta y resultados:

- Código o id de cada tabla en la que se base la consulta.
- Nombre: nombre de la empresa.
- Título: Tipo de empresa (SL, SA,).
- Idioma.
- Crédito concedido.
- Calle.
- Código postal.
- Ciudad.
- Teléfono de la empresa.

1. Crear tabla

El lenguaje SQL proporciona unas palabras clave (CREATE TABLE) para la creación y definición de una tabla en la base de datos.

En la definición hay que especificar obligatoriamente el nombre de la tabla y, entre paréntesis, la lista de definiciones de las columnas por las que está compuesta. Para cada columna hay que especificar el

nombre y el tipo de dato. De manera opcional se pueden definir valores por defecto y otro tipo de restricciones.

También es posible añadir ciertos parámetros como "IF NOT EXISTS" que permite no generar un error si intentamos crear una tabla con el mismo nombre que otra que ya exista.

```
DROP TABLE IF EXISTS aaa;  
  
CREATE TABLE IF NOT EXISTS aaa  
(  
    socio serial,  
    nombre varchar(20),  
    observaciones text,  
    cantidad int,  
    fechaAlta date,  
  
    primary key (socio)  
);
```

El primero de los comandos anteriores permiten eliminar la tabla si es que previamente existe. Esto viene bien en este momento de pruebas en el que lanzaremos varias veces las consultas. Fíjate que el comando acaba con un punto y coma (;)

El segundo comando es el propio de creación de la tabla, en este caso la tabla llamada "aaa". Después, entre paréntesis y separados por comas la lista de nombres de columna y tipos de datos.

El primer campo se llama "socio" y es de tipo serial. Este tipo de datos es una particularidad de postgres para valores numéricos que se autoincrementan. En estándar SQL sería "... GENERATED ALWAYS AS IDENTITY", y en otros sistemas gestores de bases de datos como MySQL sería "...AUTO_INCREMENT", en MS SQL sería "...INT IDENTITY(1,1)",...

El segundo campo es de tipo varchar. Este tipo almacena caracteres de longitud variable hasta un tope de 20 caracteres (en este caso).

El tercer campo es de tipo text que indica almacenamiento de texto sin límite.

El cuarto campo, cantidad, es de tipo int que guarda número enteros con 4 bytes de precisión.

El quinto y último campo es fechaNacimiento y usa un tipo date que guarda fechas.

La última línea indica que la clave primaria es el campo socio y, por tanto, que este campo es el que identifica de manera unívoca cada uno de los registros de la tabla. Es decir, cada socio debe ser único y no nulo.

The screenshot shows the pgAdmin 4 interface. In the left sidebar, the 'Tablas (344)' folder is expanded, and the 'aaa' table is selected. The 'Columnas (5)' section shows the table's structure: 'socio' (serial), 'nombre' (varchar(20)), 'observacione' (text), 'cantidad' (int), and 'fechanacimie' (date). The 'Consulta' tab is active, displaying the following SQL script:

```
1 DROP TABLE IF EXISTS aaa;
2 CREATE TABLE IF NOT EXISTS aaa
3 (
4     socio serial,
5     nombre varchar(20),
6     observaciones text,
7     cantidad int,
8     fechaNacimiento date,
9
10    primary key (socio)
11 );
```

The 'Data Output' tab shows the execution result: 'NOTICE: table "aaa" does not exist, skipping CREATE TABLE' and 'Consulta retornó exitosamente en 84 msec.'

2. Insertar registros.

Ahora que ya sabemos crear una tabla tiene sentido aprender a añadir datos a ella. Para ello el lenguaje SQL proporciona las siguientes palabras clave: INSERT INTO tabla (campo) VALUES (valor). Lo que hace esta sentencia es añadir en la última posición de la tabla indicada una nueva fila y guardará en ella los valores de los campos indicados.

```
INSERT INTO aaa (nombre, observaciones, cantidad, fechaAlta)
VALUES ('Ford', ' ', 55, '2023/02/14');
```

Este comando añade a la tabla "aaa" un nuevo registro en la última posición con los valores 'Ford' en la columna nombre, ' ' en la columna observaciones, 55 en la cantidad y '2023/02/14' en la columna fechaAlta.

Fíjate en el nombre y en el orden de los campos. Se considera de buen uso y fácil lectura especificar los campos que se tratan pero si se añaden datos exactamente en el orden en el que se especificaron los campos no sería necesario ponerlos.

En esta parte hay que prestar atención a como se pasan los datos. En general textos y fechas van encerrados entre comillas, las fechas pueden ir en muchos formatos y postgres las debería reconocer pero el predeterminado y preferido es el especificado en la norma ISO 8601 (año-mes-día). Los números van sin comillas y el separador decimal es el punto (.)

The screenshot shows the pgAdmin 4 interface. On the left, the 'Navegador' (Navigator) pane shows a tree structure with 'aaa' selected under 'Tablas (344)'. The 'Columnas (5)' for 'aaa' are listed: socio, nombre, observaciones, cantidad, and fechaAlta. The main pane shows the 'Consulta' (Query) tab with the following SQL code:

```

1 DROP TABLE IF EXISTS aaa;
2 CREATE TABLE IF NOT EXISTS aaa
3 (
4     socio serial,
5     nombre varchar(20),
6     observaciones text,
7     cantidad int,
8     fechaAlta date,
9     primary key (socio)
10 );
11 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
12 VALUES ('Ford',' ',55,'2023/02/14');
13 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
14 VALUES ('Renault','Una pequeña descripción.',16,'2023/01/14');
15 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
16 VALUES ('Ferrari',' ',76,'2023/01/30');
17 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
18 VALUES ('Mercedes',' ',2,'2022/12/31');
19 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
20 VALUES ('Honda',' ',49,'2023/02/28');
21

```

En esta tarea no se pide pero junto con el comando INSERT sería interesante probar los comandos UPDATE y DELETE para actualizar y eliminar registros.

3. Consulta sencilla

Con la tabla creada y algunos registros en ella llega el momento de hacer alguna consulta para aprovechar la verdadera potencia de las bases de datos, obtener información a partir de datos.

La estructura más sencilla para esto es mediante la sentencia SQL SELECT campos FROM tabla

Solo este apartado ya tiene entidad suficiente para tener una asignatura propia (y la tiene) así que te remito a ella o al tutorial enlazado en los apuntes para que investigues lo que necesites sobre las consultas SQL.

```

SELECT *
FROM aaa
ORDER BY fechaAlta ASC;

```

Esta sentencia SQL devuelve todas las columnas de la tabla aaa (sin ningún criterio que cumplir ya que no tiene la cláusula WHERE) y los ordena ascendentemente por el contenido de la columna fechaAlta.

The screenshot shows the pgAdmin 4 interface. On the left, the 'Navegador' (Navigator) pane shows a tree structure with 'aaa' selected under 'Tablas (344)'. The 'Columnas (5)' (Columns) pane shows the following columns: socio, nombre, observaciones, cantidad, and fechaAlta. The 'Consulta' (Query) pane shows the following SQL code:

```

1 DROP TABLE IF EXISTS aaa;
2 CREATE TABLE IF NOT EXISTS aaa
3 (
4     socio serial,
5     nombre varchar(20),
6     observaciones text,
7     cantidad int,
8     fechaAlta date,
9     primary key (socio)
10 );
11 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
12 VALUES ('Ford','',55,'2023/02/14');
13 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
14 VALUES ('Renault','Una pequeña descripción.',16,'2023/01/14');
15 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
16 VALUES ('Ferrari','',76,'2023/01/30');
17 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
18 VALUES ('Mercedes','',2,'2022/12/31');
19 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
20 VALUES ('Honda','',49,'2023/02/28');
21
22 SELECT *
23 FROM aaa
24 ORDER BY fechaAlta ASC;

```

The 'Data Output' pane shows the results of the query:

	socio [PK] integer	nombre character varying (20)	observaciones text	cantidad integer	fechaAlta date
1	4	Mercedes		2	2022-12-31
2	2	Renault	Una pequeña descripción.	16	2023-01-14
3	3	Ferrari		76	2023-01-30
4	1	Ford		55	2023-02-14
5	5	Honda		49	2023-02-28

Total rows: 5 of 5 Query complete 00:00:00.128

4. Subconsultas

En ocasiones las consultas son algo más complicadas porque requieren de varias tablas o de la misma varias veces. Este último caso es el que te propongo en este apartado.

Mi recomendación detenerse a pensar como haría esa consulta sin usar ordenadores, solo usando las tarjetas de los viejos archivadores. En este ejercicio me piden el nombre del registro con fecha más antigua.

Si lo tuviera que hacer a mano, en la primera pasada me dedicaría a detectar la fecha más antigua. En principio sería la de la primera tarjeta. Al pasar a la segunda (y siguientes) me preguntaría si esa fecha es más antigua que mi candidata. En función de la respuesta actualizaría o no el valor de fecha más antigua. Al acabar esta pasada ya tendría el valor de la fecha más antigua.

En la segunda iteración iría pasando tarjetas fijándome en si su fecha coincide con la que tengo definida como más antigua. Cuando por fin la encuentre devolveré su nombre.

```
SELECT nombre
FROM aaa
WHERE fechaAlta = (SELECT min(fechaalta) FROM aaa);
```

Esta sentencia SQL contiene las dos iteraciones que te contaba antes. Para leerla y entenderla debemos analizarla desde el final hasta el principio. Puede parecer un poco raro que la primera parte en ejecutarse (buscar la menor fecha) sea la última en aparecer pero tiene sentido si piensas en las prioridades matemáticas y como lo primero en calcularse es lo de los paréntesis más internos.

The screenshot shows the pgAdmin 4 interface. On the left, the 'Navegador' (Navigator) pane shows the database structure, with 'aaa' table selected under 'Tablas (344)'. The table has 5 columns: 'socio', 'nombre', 'observaciones', 'cantidad', and 'fechaAlta'. The main pane shows the SQL query being executed:

```
1 DROP TABLE IF EXISTS aaa;
2 CREATE TABLE IF NOT EXISTS aaa
3 (
4     socio serial,
5     nombre varchar(20),
6     observaciones text,
7     cantidad int,
8     fechaAlta date,
9     primary key (socio)
10 );
11 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
12 VALUES ('Ford',' ',55,'2023/02/14');
13 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
14 VALUES ('Renault','Una pequeña descripción.',16,'2023/01/14');
15 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
16 VALUES ('Ferrari',' ',76,'2023/01/30');
17 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
18 VALUES ('Mercedes',' ',2,'2022/12/31');
19 INSERT INTO aaa (nombre, observaciones,cantidad,fechaAlta)
20 VALUES ('Honda',' ',49,'2023/02/28');
21
22 SELECT nombre
23 FROM aaa
24 WHERE fechaAlta = (SELECT min(fechaalta) FROM aaa);
```

The 'Data Output' pane shows the result of the query:

nombre
Mercedes

5. Las tablas de Odoo



La solución para esta consulta de las tablas de Odoo ya aparece en otra guía anterior así que te remito a ella para completarla. A pesar de ya estar resuelta la dejo aquí para ver una consulta en la que intervienen más de una tabla.

