Módulo 4. Consultas básicas SQL

- Consultas básicas de selección
- Consultas calculadas
- Consultas básicas sobre una base de datos en Coding Rooms
- Un ejemplo de consulta multitabla

Copyright 2026

Consultas básicas de selección

La sentencia SELECT

El comando SELECT permite:

- Obtener datos de ciertas columnas de una tabla.
- Obtener filas de una tabla de acuerdo con ciertos criterios.
- Realizar cálculos sobre los datos.
- Mezclar datos de tablas diferentes.
- Agrupar datos.

En este curso, vamos a quedarnos en lo más sencillo, que son las opciones 1, 2 y 3.

En su forma más simple, la sintaxis para la expresión SELECT es:

SELECT campos FROM tablas [WHERE condiciones]

- **campos:** son las columnas (separadas por comas) o cálculos que se desea recuperar de la base de datos. Se puede utilizar * si se desean recuperar todas las columnas.
- **tablas:** indica las tablas de las que se desean recuperar los registros. Mínimo debe haber una tabla para poder recuperar registros.
- **condiciones:** las condiciones que deben cumplir las filas que se desean recuperar.

La sentencia más sencilla sería la de recuperar todos los campos de una tabla:

```
SELECT *
FROM alumnos;
```

Si se desea recuperar solo DNI, nombre y apellidos de los alumnos, entonces se pondría:

```
SELECT DNI, nombre, apellidos
FROM alumnos;
```

Copyright 2026

Condiciones

La cláusula **WHERE** sirve para restringir los datos de salida. Permite poner una o varias condiciones que han de cumplir las filas que se desean recuperar, las que no cumplan esas condiciones no aparecerán en el resultado de la consulta.

Por ejemplo, si solo se desea mostrar los datos de las asignaturas que tengan una duración mayor de 2 horas a la semana. Sería como sigue:

```
SELECT *
FROM Asignatura
WHERE Horas_semanales > 2;
```

Se van a explicar a continuación los operadores básicos que se pueden utilizar en la cláusula WHERE.

Operadores de comparación

Operador	Significado
>	Mayor que
<	Menor que
>=	Mayor o igual que
<=	Menor o igual que
=	Igual
<>	Distinto
!=	Distinto

Valores lógicos

Operador	Significado
AND	Devuelve verdadero si las expresiones de su izquierda y derecha son ambas verdaderas.
OR	Devuelve verdadero si cualquiera de las dos expresiones a izquierda y derecha del OR, son verdaderas.

Copyright 2026

NOT	Invierte la lógica de la expresión que está a su derecha. Si es verdadera, mediante NOT pasa a ser falsa.

Por ejemplo, si se desea mostrar los datos de las asignaturas que tengan una duración mayor de 2 horas a la semana pero menor de 6. Sería:

```
SELECT *
FROM Asignatura
WHERE Horas_semanales>2 AND Horas_semanales<6;
```

LIKE

Se usa sobre todo con textos, permite obtener registros cuyo valor en un campo cumpla una condición textual. LIKE utiliza una cadena que puede contener estos símbolos:

Símbolo	Significado
%	Una serie cualquiera de caracteres
_	Un carácter cualquiera

Por ejemplo, mostrar los nombres de los alumnos que empiecen por S:

```
SELECT nombre
FROM alumnos
WHERE nombre LIKE 'S%';
```

IS NULL

Devuelve verdadero si el valor que examina es nulo. Los campos tienen valor nulo cuando al insertar una fila, ese campo se deja vacío.

La siguiente sentencia devuelve el nombre y apellidos de los alumnos que no tienen email.

```
SELECT nombre, apellidos
FROM alumno
WHERE email IS NULL
```

> Hay otros operadores como **BETWEEN** o **IN**, que no se van a explicar en este curso.

Copyright 2026

ORDENACIÓN

El orden inicial de los registros obtenidos por una SELECT es el orden en el que fueron introducidos. Para ordenar basándose en otros criterios, se utiliza la cláusula **ORDER BY**.

En esa cláusula se coloca una lista de campos que indica la forma de ordenar. Se ordena primero por el primer campo de la lista, si hay coincidencias por el segundo, si ahí también las hay por el tercero, y así sucesivamente.

Se puede colocar la palabra **ASC** O **DESC** (por defecto, si se indica nada, utiliza ASC).

Por ejemplo, la siguiente sentencia muestra todos los datos de los alumnos que no tienen email, ordenados alfabéticamente por el nombre:

```
SELECT nombre, apellidos FROM alumno
WHERE email IS NULL
ORDER BY nombre ASC;
```

Contar filas

Existe una función que sirve para contar filas, es la función **count**. Sirve para contar filas. Vamos a verlo con algún ejemplo.

La siguiente consulta cuenta cuántos alumnos hay en la tabla alumno:

```
SELECT count(*)
from alumno;
```

Si queremos contar cuántos alumnos hay cuyo apellido empiece por A:

```
SELECT count(*)
from alumno
where apellidos like 'A%';
```

Financiado por el Ministerio de Educación y Formación Profesional y por la Unión Europea - NextGenerationEU

Copyright 2026



Copyright 2026

Consultas calculadas

CONSULTAS CALCULADAS

Cálculos aritméticos

En las consultas se pueden utilizar los operadores + (suma), - (resta), * (multiplicación) y / (división) para hacer cálculos. Cuando se utilizan como expresión en una consulta SELECT, no modifican los datos originales sino que como resultado de la vista generada por la consulta, aparece una nueva columna.

Ejemplo:

```
SELECT dni, nota, nota*2
FROM alumnos;
```

Esa consulta obtiene tres columnas: el dni, la nota y la nota multiplicada por 2 . La tercera tendrá como nombre la expresión utilizada, para poner un alias basta utilizar dicho alias tras la expresión **as**:

```
SELECT dni, nota, nota*2 as nota_doble
FROM alumnos;
```

La prioridad de esos operadores es la normal: tienen más prioridad la multiplicación y división, después la suma y la resta. En caso de igualdad de prioridad, se realiza primero la operación que esté más a la izquierda. Se puede evitar cumplir esa prioridad usando paréntesis; el interior de los paréntesis es lo que se ejecuta primero.

Cuando una expresión aritmética se calcula sobre valores NULL, el resultado es el mismo valor NULL.

Concatenación de textos

Todas las bases de datos incluyen algún operador para encadenar textos. En SQLSERVER es el signo + en Oracle son los signos || y en MySQL se hace con la función **CONCAT** .

Ejemplo:

```
SELECT dni, concat(apellidos,', ',nombre), email
FROM alumnos;
```

El resultado de esa consulta es una tabla con tres columnas: la primera el DNI, la segunda columna es la concatenación de los apellidos con una coma y un espacio y luego el nombre, y la tercera columna el email.

Financiado por el Ministerio de Educación y Formación Profesional y por la Unión Europea - NextGenerationEU



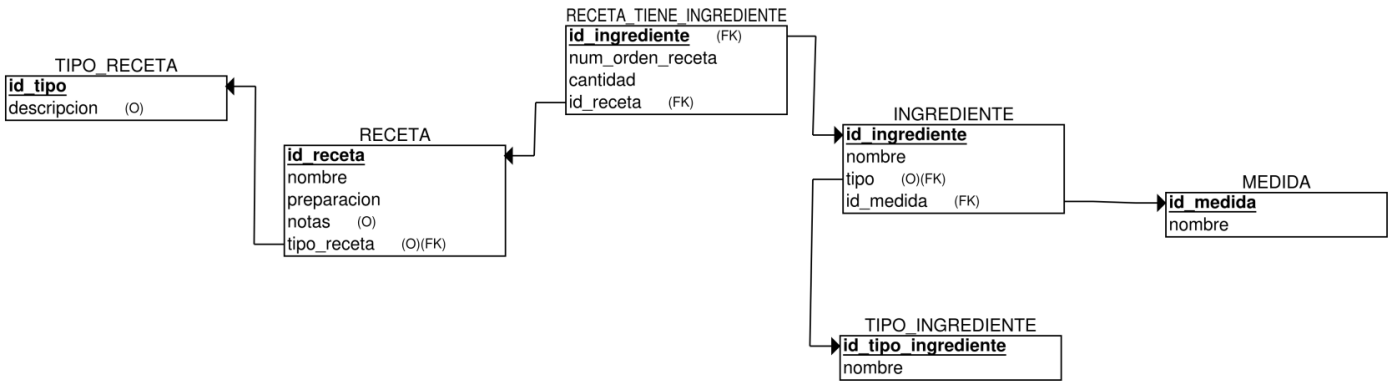
Copyright 2026

Consultas básicas sobre una base de datos en Coding Rooms

Aquí van unos pocos ejemplos de consultas básicas sobre una base de datos, para practicar sobre la herramienta Coding rooms.

Antes de nada, aquí puedes descarar la base de datos para ejecutarla en la herramienta que más abajo se explica paso a paso cómo hacerlo.

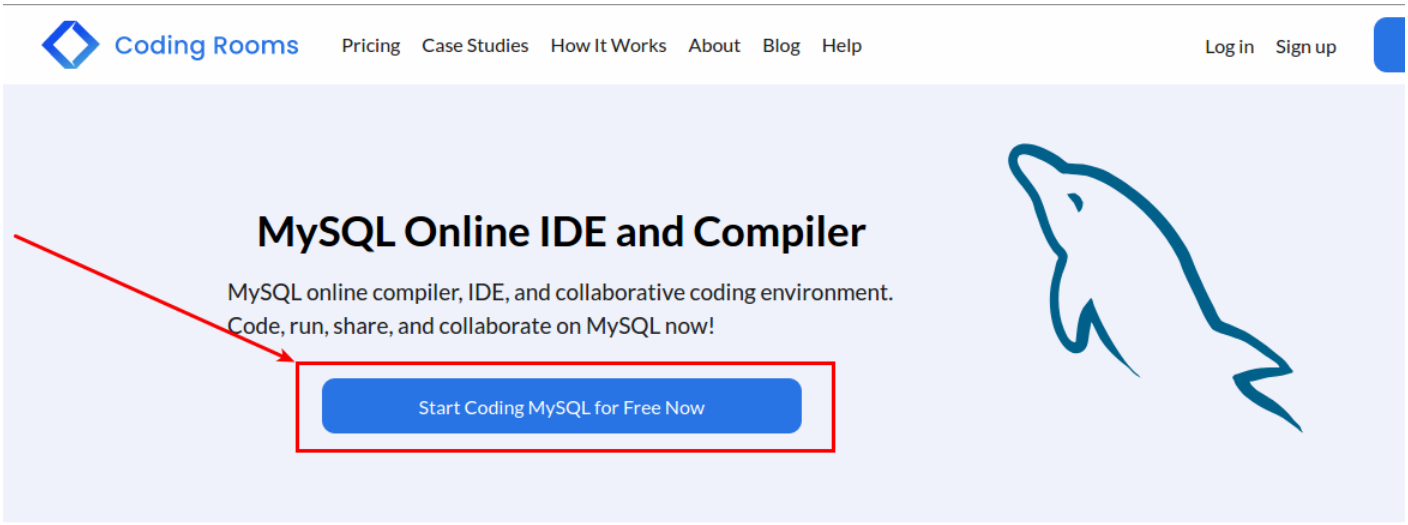
La base de datos es sobre recetas, tiene el siguiente diagrama relacional:



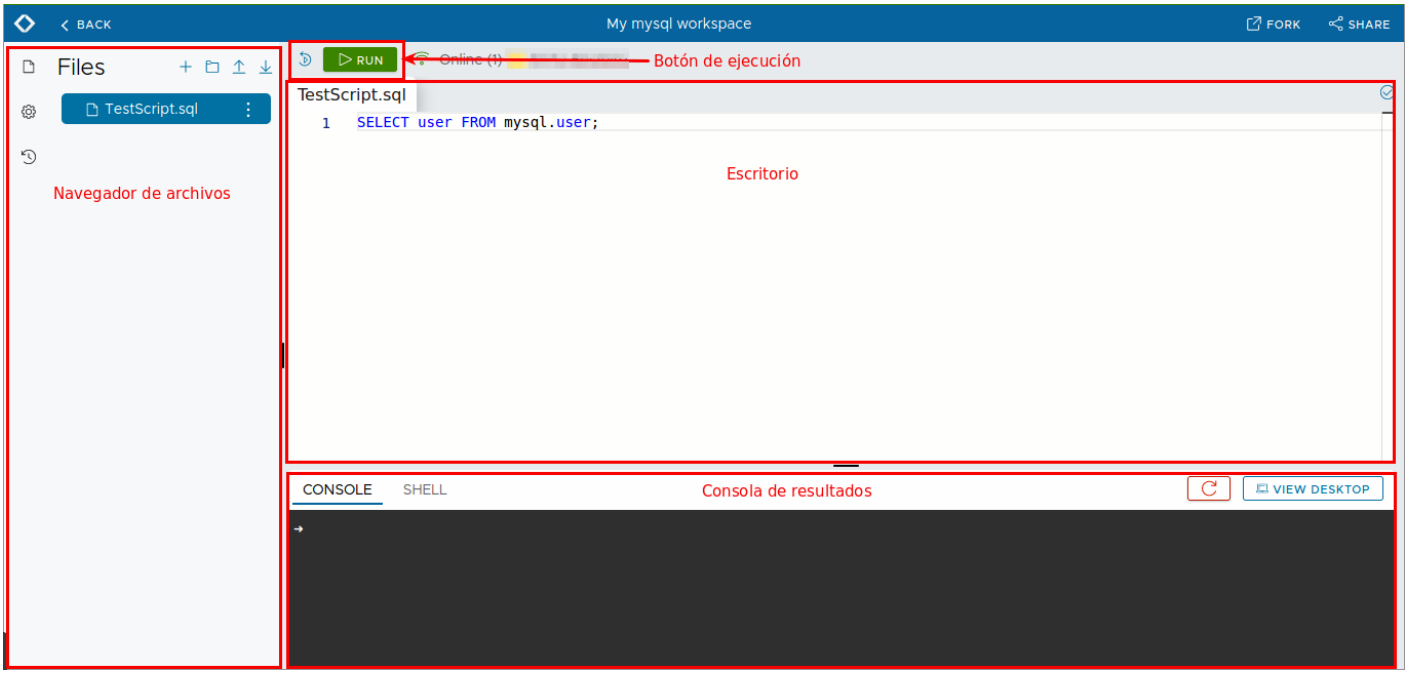
Coding Rooms

Para utilizar Coding Rooms no se necesario registrarse si se accede directamente al workspace desde este enlace. Pero si te registras se guardan los workspace que vayas utilizando, por lo que yo lo recomiendo.

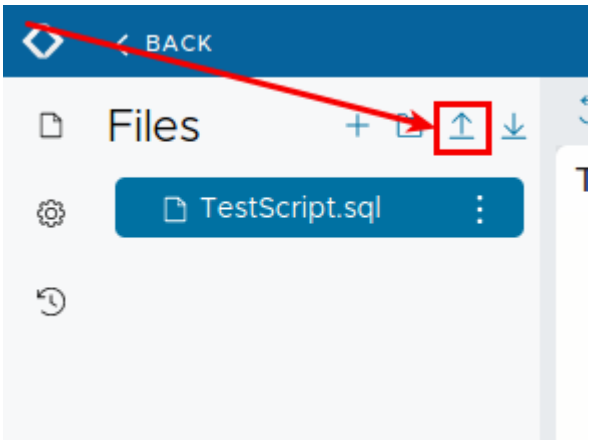
Para iniciar el compilador solo hay que pulsar sobre el botón "Start coding MySQL for free now".



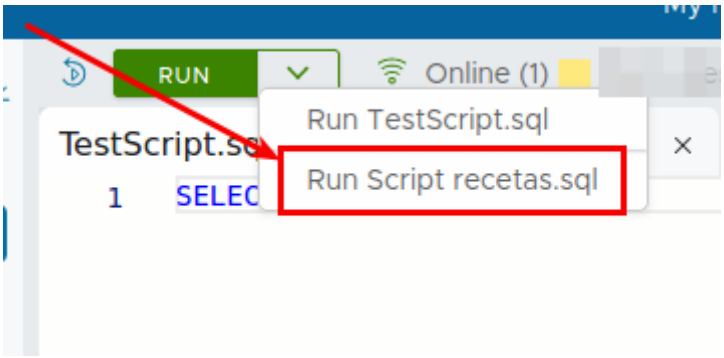
Esto abre el Workspace de MySQL siguiente.



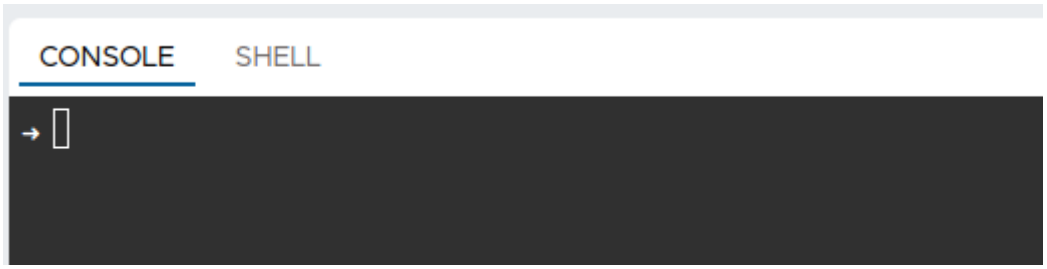
Lo primero que tenemos que hacer es ejecutar el script descargado para crear la base de datos de recetas. Para eso, en el área de navegación de archivos, clicar sobre la flecha hacia arriba para subir el archivo.



El archivo se carga y haciendo clic sobre él se abre en el escritorio. Ahora solo hay que desplegar los archivos del botón de ejecución y seleccionar el archivo que se desea ejecutar.



El resultado de la ejecución se carga en la consola de debajo. En el caso del script de creación de la base de datos de recetas, como no son consultas, no muestra nada por pantalla, así que si no muestra ningún error, es que todo ha ido bien.



Con esto ya deberíamos tener la base de datos creada en el servidor de coding rooms para empezar a hacer consultas.

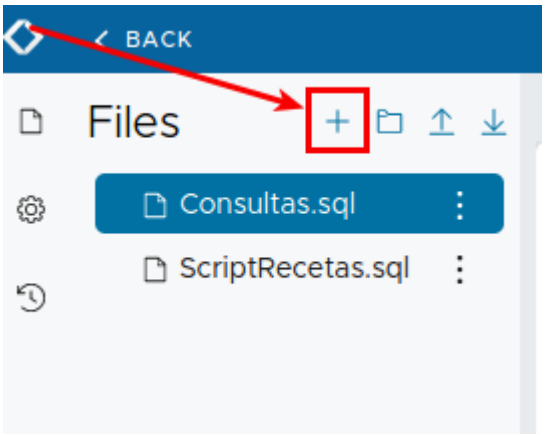
Consultas SQL sencillas

En el archivo que podéis descargar aquí, hay ejemplos de consultas que a continuación se explican.

Ejecutar consultas en Coding Rooms

Para ejecutar cada consulta en Coding Rooms, hay que poner la consulta en un archivo y ejecutar ese archivo.

Para crear un archivo nuevo hay que pulsar en el + del navegador de ficheros:



En ese nuevo archivo podéis ir escribiendo cada consulta, y el resultado se mostrará en la consola.

Ejemplos del archivo

1. Mostrar todos los tipos de recetas que hay en la base de datos:

```
select * from tipo_receta;
```

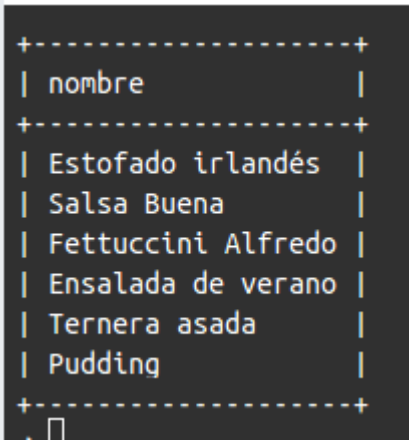
La consola muestra el resultado:



2. Mostrar el nombre de todas las recetas que no tienen anotaciones (en el campo notas)

```
select nombre from receta where notas is null;
```

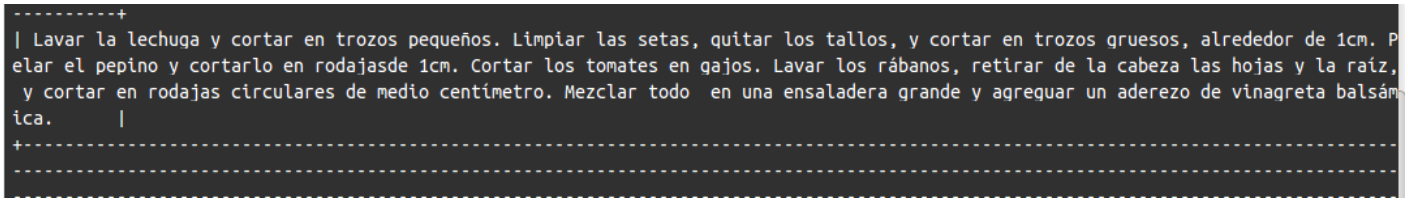
Resultado:



3. Queremos saber cómo se prepara la "Ensalada de verano"

```
select preparacion from receta where nombre like 'Ensalada de verano';
```

Resultado:



Ahora vamos a ver cómo sacaríamos información que hay en varias tablas sin haber visto las consultas multitabla.

4. Queremos saber qué recetas llevan 'Ternera'.

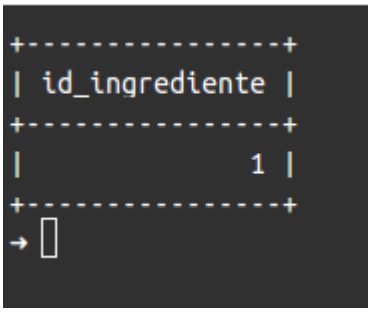
Los ingredientes están en la tabla **ingrediente**, la unión de recetas con ingredientes está en la tabla **receta_tiene_ingrediente**, pero el nombre de las recetas está en la tabla **receta**.

Primero hay que buscar cuál es el identificador del ingrediente 'Ternera' para buscarlo en la tabla **receta_tiene_ingrediente**. Así esta tabla nos dará los identificadores de las recetas que llevan ternera y con ese identificador podremos ir a buscar las recetas en la tabla **receta**.

Identificador del ingrediente 'Ternera':

```
select id_ingrediente from ingrediente where nombre like 'Ternera';
```

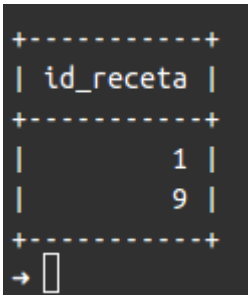
Resultado:



Identificador de las recetas que llevan 'Ternera':

```
select id_receta from receta_tiene_ingrediente where id_ingrediente=1;
```

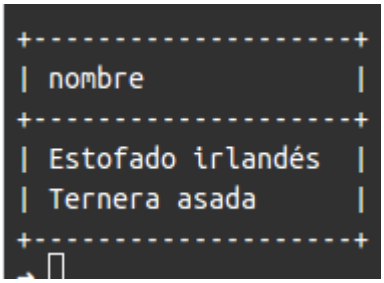
Resultado:



Recetas que llevan 'Ternera':

```
select nombre from receta where id_receta=9 or id_receta=1;
```

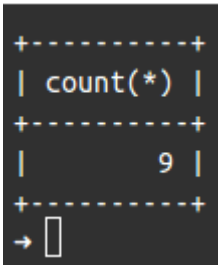
Resultado:



5. Contar cuántas recetas tienen anotaciones (en el campo notas).

```
select count(*) from receta where notas is not null;
```

Resultado:



Financiado por el Ministerio de Educación y Formación Profesional y por la Unión Europea - NextGenerationEU

Un ejemplo de consulta multitabla

Vamos a ver cómo sacaríamos los nombres de las recetas que llevan ternera usando solo una consulta sobre arias tablas a la vez.

La manera de hacerlo consultando las tablas por separado

Los ingredientes están en la tabla **ingrediente**, la unión de recetas con ingredientes está en la tabla **receta_tiene_ingrediente**, pero el nombre de las recetas está en la tabla **receta**.

Primero hay que buscar cuál es el identificador del ingrediente 'Ternera' para buscarlo en la tabla **receta_tiene_ingrediente**. Así esta tabla nos dará los identificadores de las recetas que llevan ternera y con ese identificador podremos ir a buscar las recetas en la tabla **receta**.

Identificador del ingrediente 'Ternera':

```
select id_ingrediente from ingrediente where nombre like 'Ternera';
```

Identificador de las recetas que llevan 'Ternera':

```
select id_receta from receta_tiene_ingrediente where id_ingrediente=1;
```

Recetas que llevan 'Ternera':

```
select nombre from receta where id_receta=9 or id_receta=1;
```

Consulta multitabla

El modo básico de hacerlo es poniendo en la cláusula **from** las tablas separadas por comas, y en la cláusula **where** indicar cómo se relacionan esas tablas, es decir, indicando qué columnas son las que deben ser iguales en cada fila.

En el ejemplo anterior sería:

Copyright 2026

```
SELECT receta.nombre
FROM ingrediente, receta_tiene_ingrediente, receta
WHERE ingrediente.id_ingrediente = receta_tiene_ingrediente.id_ingrediente
and receta_tiene_ingrediente.id_receta = receta.id_receta
    and ingrediente.nombre like 'Ternera';
```

> Fíjate que se están indicando las columnas con el nombre de la tabla delante seguidos por un punto. Esto es necesario cuando en una consulta multitabla hay columnas en diferentes tablas con el mismo nombre, como es este caso, para saber a la columna de qué tabla se está haciendo referencia.

Financiado por el Ministerio de Educación y Formación Profesional y por la Unión Europea - NextGenerationEU



Copyright 2026