

CONSULTAS BÁSICAS

1. Realiza una consulta que nos muestre los campos Título, FECHAADQUISICION de todas las películas, ordenando descendientemente por el Título.

```
SELECT titulo, fechadquisicion  
FROM peliculas  
ORDER BY titulo DESC;
```

2. Realiza una consulta que nos muestre los campos Título, FECHAADQUISICION y Género de todas las películas, ordenando ascendientemente por FECHAADQUISICION y descendientemente por Género.

```
SELECT titulo, fechadquisicion, genero  
FROM peliculas  
ORDER BY fechadquisicion, genero DESC;
```

3. Realiza una consulta que nos muestre los campos Título, FECHAADQUISICION, Género y Tipo de todas las películas, ordenando ascendientemente por Tipo y Título.

```
SELECT titulo, fechadquisicion, genero, tipopelicula  
FROM peliculas  
ORDER BY tipopelicula, titulo;
```

4. Realiza una consulta que nos muestre en qué diferentes fechas concretas se han adquirido películas.

```
SELECT DISTINCT fechadquisicion  
FROM peliculas  
ORDER BY fechadquisicion;
```

5. Realiza una consulta que nos muestre el Título y Género de las 7 últimas películas (en orden alfabético) del género 5.

```
SELECT TOP 7 titulo, genero  
FROM peliculas  
WHERE (genero = 5)  
ORDER BY titulo DESC;
```

6. Realiza una consulta que nos muestre todos los campos de las películas cuyo género esté entre 5 y 12, ordenadas por género.

```
SELECT *  
FROM peliculas, generos  
WHERE GENERO BETWEEN 5 AND 12  
ORDER BY GENERO;
```

7. Realiza una consulta que nos muestre todos los campos de las películas cuyo precio esté entre 15 y 18, ordenadas por título.

```
SELECT *  
FROM Peliculas  
WHERE precio BETWEEN 15 AND 18  
ORDER BY titulo;
```

8. Realiza una consulta que nos muestre todos los campos de las películas adquiridas en el año 98.

```
SELECT *  
FROM Peliculas  
WHERE Year(fechadquisicion)=1998;
```

9. Realiza una consulta que nos muestre todos los campos de las películas adquiridas en el mes de agosto del año 99.

```
SELECT *  
FROM Peliculas  
WHERE Month(fechadquisicion)=8 and Year(fechadquisicion)=1999;
```

10. Realiza una consulta que nos muestre el Título de la película y al lado una columna donde aparezca 'Para niños' si el género es 2 ó 9, o que aparezca 'Para adultos' en caso contrario. (El título de la nueva columna se llamará RECOMENDADA).

```
SELECT titulo, iif(genero=2 or genero=9,'Para niños','Para adultos') AS Recomendada  
FROM peliculas;
```

11. Realiza una consulta que nos muestre el Título de la película y al lado una columna donde aparezca 'Estreno' si el tipo es 1, 'Normal' si el tipo es 2 o que aparezca 'Oferta' en caso contrario. (El título de la nueva columna se llamará MODALIDAD).

```
SELECT titulo, iif(tipopelicula=1,'Estreno',iif(tipopelicula=2,'Normal','Oferta')) AS  
Modalidad  
FROM peliculas;
```

12. Realiza una consulta que nos muestre los Títulos de películas que empiecen por M o P.

```
SELECT titulo  
FROM peliculas  
WHERE titulo LIKE 'M*' or titulo LIKE 'P*';
```

13. Realiza una consulta que nos muestre los Títulos de películas que acaben en la letra S.

```
SELECT titulo  
FROM peliculas  
WHERE titulo LIKE '*S';
```

14. Realiza una consulta que nos muestre los Títulos de películas que contengan la palabra REY.

```
SELECT titulo
FROM peliculas
WHERE titulo LIKE 'LEY *' OR titulo LIKE '* LEY *' OR titulo LIKE '* LEY' OR titulo LIKE 'LEY';
```

15. Realiza una consulta que nos muestre los Títulos y Géneros de películas que tengan 4 caracteres en su título.

```
SELECT titulo, genero
FROM peliculas
WHERE titulo LIKE '????';
```

16. Realiza una consulta que nos muestre los Títulos y Géneros de películas que tengan 4 caracteres en su título y sean de un género superior al 3.

```
SELECT titulo, genero
FROM peliculas
WHERE titulo LIKE '????' AND genero>3;
```

17. Realiza una consulta que nos muestre los Títulos de películas que tengan por lo menos un carácter numérico.

```
SELECT titulo, genero
FROM peliculas
WHERE titulo LIKE '*#*';
```

18. Realiza una consulta que nos muestre los Títulos y la fecha de adquisición de las películas que empiecen por alguno de los siguientes caracteres: C,D,E,F,G,H

```
SELECT titulo, fechadquisicion
FROM peliculas
WHERE titulo LIKE '[C-H]*';
```

19. Realiza una consulta que nos muestre los Títulos y la fecha de adquisición de las películas que empiecen por alguno de los siguientes caracteres:
C,D,E,F,G,H,P,Q,R,S,T,U,V

```
SELECT titulo, fechadquisicion
FROM peliculas
WHERE titulo LIKE '[C-H,P-V]*';
```

20. Realiza una consulta que nos muestre los Títulos y la fecha de adquisición de las películas que no terminen por alguno de los siguientes caracteres: I,J,K,L,M,N,O,P

```
SELECT titulo, fechadquisicion
FROM peliculas
WHERE titulo NOT LIKE '*[I-P]';
```

21. Realiza una consulta que muestre los Títulos de películas que no contengan la letra a.

```
SELECT titulo, fechadquisicion  
FROM peliculas  
WHERE titulo NOT LIKE '*A*';
```

22. Realiza una consulta que nos muestre los Títulos y el género de las películas cuyo género sea 2,4,6,8,10,12, ordenadas ascendentemente por el título.

```
SELECT titulo, genero  
FROM peliculas  
WHERE genero IN(2,4,6,8,10,12);
```

CONSULTAS DE AGRUPACION DE REGISTROS

1. Realiza una consulta que nos agrupe las películas por genero.

```
SELECT genero
FROM peliculas
GROUP BY genero;
```

2. Realiza una consulta que nos muestre cuantas películas existen de cada género. (Mostrar como título de columna TOTAL PELICULAS)

```
SELECT genero, COUNT(codigopelicula) AS [TOTAL PELICULAS]
FROM peliculas
GROUP BY genero;
```

3. Realiza una consulta que nos muestre cuantas películas existen de los géneros entre 5 y 10. (Mostrar como título de columna TOTAL PELICULAS)

```
SELECT genero, COUNT(codigopelicula) AS [TOTAL PELICULAS]
FROM peliculas
WHERE genero IN(5,6,7,8,9,10)
GROUP BY genero;
```

4. Realiza una consulta que nos agrupe las películas por fecha de adquisición.

```
SELECT fechadquisicion
FROM peliculas
GROUP BY fechadquisicion;
```

5. Realiza una consulta que nos muestre cuantas películas existen de cada fecha de adquisición. (Mostrar como título de columna TOTAL PELICULAS)

```
SELECT fechadquisicion, COUNT(codigopelicula) AS [TOTAL PELICULAS]
FROM peliculas
GROUP BY fechadquisicion;
```

6. Realiza una consulta que nos agrupe las películas por genero y fecha de adquisición.

```
SELECT genero, fechadquisicion
FROM peliculas
GROUP BY genero, fechadquisicion;
```

7. Realiza una consulta que nos muestre cuantas películas existen de cada género y fecha de adquisición. (Mostrar como título de columna TOTAL PELICULAS)

```
SELECT genero, fechadquisicion, COUNT(codigopelicula) AS [TOTAL PELICULAS]
FROM peliculas
GROUP BY genero, fechadquisicion;
```

8. Realiza una consulta que nos muestre la cantidad gastada en la adquisición de películas por cada fecha. (Mostrar como título de columna TOTAL COMPRAS)

```
SELECT fechadquisicion, SUM(precio) AS [TOTAL COMPRAS]
FROM peliculas
GROUP BY fechadquisicion;
```

9. Realiza una consulta que nos muestre la cantidad gastada en la adquisición de películas del año 99. (Mostrar como título de columna TOTAL COMPRAS AÑO 99)

```
SELECT SUM(precio) AS [TOTAL COMPRAS AÑO 99], COUNT(codigopelicula) AS [TOTAL
PELICULAS]
FROM peliculas
WHERE YEAR(fechadquisicion)=1999;
```

10. Realiza una consulta que nos muestre la cantidad gastada (añadiendo el IVA del 16% incluido) en la adquisición de películas del año 99. (Mostrar como título de columna TOTAL COMPRAS AÑO 99)

```
SELECT SUM(precio) AS [TOTAL COMPRAS AÑO 99], SUM(precio)*1.16 AS [TOTAL
COMPRAS CON INVA AÑO 99]
FROM peliculas
WHERE YEAR(fechadquisicion)=1999;
```

11. Realiza una consulta que nos muestre el promedio cantidad gastada en la adquisición de películas por cada fecha. (Mostrar como título de columna PROMEDIO)

```
SELECT fechadquisicion, AVG(precio) AS PROMEDIO
FROM peliculas
GROUP BY fechadquisicion;
```

12. Realiza una consulta que nos muestre de cada tipo de película la última fecha de adquisición.

```
SELECT tipopelicula, MAX(fechadquisicion) AS [ULTIMA ADQUISICION]
FROM peliculas
GROUP BY tipopelicula;
```

13. Realiza una consulta que nos muestre de cada tipo de película la primera fecha de adquisición.

```
SELECT tipopelicula, MIN(fechadquisicion) AS [PRIMERA ADQUISICION]
FROM peliculas
GROUP BY tipopelicula;
```

14. Realiza una consulta que nos muestre el precio más barato de las películas

```
SELECT MIN(precio) AS [PRECIO MAS BARATO], MAX(precio) AS [PRECIO MAS CARO]  
FROM peliculas;
```

15. Realiza una consulta que nos muestre el precio más caro de las películas.

```
SELECT MIN(precio) AS [PRECIO MAS BARATO], MAX(precio) AS [PRECIO MAS CARO]  
FROM peliculas;
```

16. Realiza una consulta que nos muestre el precio más barato de las películas cada género.

```
SELECT genero, MIN(precio) AS [PRECIO MAS BARATO], COUNT(codigopelicula) AS  
[CANTIDAD PELICULAS]  
FROM peliculas  
GROUP BY genero;
```

SUBCONSULTAS

```
SELECT*FROM ALUMNOS WHERE EDAD>(SELECT AVG(EDAD)FROM ALUMNOS)
SELECT*FROM ALUMNOS WHERE EDAD>ANY(SELECT EDAD FROM ALUMNOS WHERE
PROCEDENCIA='GM')
SELECT*FROM ALUMNOS WHERE EDAD>ALL(SELECT EDAD FROM ALUMNOS WHERE
PROCEDENCIA='GM')
SELECT*FROM ALUMNOS WHERE EDAD NOT IN(SELECT EDAD FROM ALUMNOS WHERE
PROCEDENCIA='GM')
```

1. Selecciona el Título y Precio de las películas del genero 2 cuyo precio sea superior a cualquier película del genero 6

```
SELECT titulo, precio
FROM peliculas
WHERE genero=2 AND precio>ANY(SELECT precio FROM peliculas WHERE genero=6);
```

2. Selecciona el Título y Precio de las películas del genero 2 cuyo precio sea superior a todas películas del genero 6

```
SELECT titulo, precio
FROM peliculas
WHERE genero=2 AND precio>ALL(SELECT precio FROM peliculas WHERE genero=6);
```

3. Selecciona el Título y Precio de las películas del genero Aventuras cuyo precio sea superior a cualquier película del genero Drama

```
SELECT titulo, precio
FROM peliculas
WHERE genero=2 AND precio>ANY(SELECT precio FROM peliculas WHERE genero=6);
```

4. Selecciona el Título y Precio de las películas del genero 2 cuyo precio sea igual a los precios de las películas del genero 6

```
SELECT titulo, precio
FROM peliculas
WHERE genero=2 AND precio=ANY(SELECT precio FROM peliculas WHERE genero=6);
```

5. Selecciona el Título y Precio de las películas del genero 2 cuyo precio sea diferente a los precios de las películas del genero 6

```
SELECT titulo, precio
FROM peliculas
WHERE genero=2 AND precio NOT IN(SELECT precio FROM peliculas WHERE genero=6);
```

6. Selecciona el Título de las películas que nunca han sido alquiladas.

```
SELECT titulo
FROM peliculas
WHERE codigopelicula NOT IN(SELECT codigopelicula FROM alquileres);
```


7. Selecciona el Título y el Precio de las películas cuyo precio supere la media de todas las películas.

```
SELECT titulo, precio  
FROM peliculas  
WHERE precio > (SELECT AVG(precio) FROM peliculas);
```

CONSULTAS DE UNION

```
SELECT titulo, precio FROM peliculas AS p, generos AS g WHERE  
p.genero=g.codigogenero AND nombregenero="Bélica" AND precio=ANY(SELECT  
precio FROM peliculas AS p, generos AS g WHERE p.genero=g.codigogenero AND  
nombregenero="Drama");
```

1. Realiza una consulta que nos muestre todos los campos de la unión de los registros de películas con los de tipos.

```
SELECT p.*, t.*  
FROM peliculas AS p, tipospelicula AS t  
WHERE p.tipopelicula=t.codigoentrega;
```

2. Realiza una consulta que nos muestre el título de la película y la modalidad (que se encuentra en la tabla de tipos)

```
SELECT p.titulo, t.modalidad  
FROM peliculas AS p, tipospelicula AS t  
WHERE p.tipopelicula=t.codigoentrega;
```

3. Realiza una consulta que nos muestre el título de la película y la modalidad (que se encuentra en la tabla de tipos) de todas las modalidades de Estreno

```
SELECT p.titulo, t.modalidad  
FROM peliculas AS p, tipospelicula AS t  
WHERE p.tipopelicula=t.codigoentrega and t.modalidad="Estreno";
```

4. Realiza una consulta que nos muestre la cantidad de películas que hay de cada modalidad.

```
SELECT t.modalidad, COUNT(p.codigopelicula) AS [Cantidad Peliculas]  
FROM peliculas AS p, tipospelicula AS t  
WHERE p.tipopelicula=t.codigoentrega  
GROUP BY t.modalidad;
```

5. Realiza una consulta que nos muestre el título de la película y el nombre del genero (que se encuentra en la tabla de generos)

```
SELECT p.titulo, g.nombregenero  
FROM peliculas AS p, generos AS g  
WHERE p.genero=g.codigogenero;
```

6. Realiza una consulta que nos muestre el título de la película y el nombre del genero (que se encuentra en la tabla de generos) de los generos de Terror y Suspense

```
SELECT p.titulo, g.nombregenero
FROM peliculas AS p, generos AS g
WHERE p.genero=g.codigogenero AND (nombregenero="Acción" OR
nombregenero="Suspense");
```

7. Realiza una consulta que nos muestre la cantidad de películas que hay de cada genero.

```
SELECT g.nombregenero, COUNT (p.codigopelicula) AS [Cantidad Peliculas]
FROM peliculas AS p, generos AS g
WHERE p.genero=g.codigogenero
GROUP BY g.nombregenero;
```

8. Realiza una consulta que nos muestre el título de la película, el nombre del genero (que se encuentra en la tabla de generos) y la modalidad (que se encuentra en la tabla de tipos).

```
SELECT p.titulo, g.nombregenero, t.modalidad
FROM peliculas AS p, generos AS g, tipospelicula AS t
WHERE p.genero=g.codigogenero AND p.tipopelicula=t.codigoentrega;
```

9. Realiza una consulta que nos muestre el título de la película, el nombre del genero (que se encuentra en la tabla de géneros) y la modalidad (que se encuentra en la tabla de tipos) y el código de cliente que tiene esa película (que se encuentra en la tabla de alquileres).

10. Mostrar los nombres de clientes y empleados residentes en la ciudad Londres.

```
SELECT p.titulo, g.nombregenero, t.modalidad, a.codigocliente
FROM peliculas AS p, generos AS g, tipospelicula AS t, alquileres AS a
WHERE p.genero=g.codigogenero AND p.tipopelicula=t.codigoentrega AND
p.codigopelicula=a.codigopelicula;
```

11. Mostrar los nombres de clientes que residan en Madrid y empleados residentes en Londres.

12. Mostrar los nombres y la ciudad de clientes que residan en el país Alemania y empleados residentes en Reino Unido.

CONSULTAS DE REFERENCIAS CRUZADAS

```
TRANSFORM COUNT(c.dni)
SELECT cl.ciudad, COUNT(c.dni) AS TOTAL
FROM ciudades AS cl, clientes AS c
WHERE cl.idciudad=c.idciudad
GROUP BY cl.ciudad
PIVOT YEAR(fechanacimiento);
```

1. Realiza una consulta que nos muestre los generos de películas por filas y por columnas los meses de la fecha de adquisición, con la finalidad de saber la cantidad de películas que tenemos.

```
TRANSFORM COUNT(p.codigopelicula)
SELECT g.nombregenero
FROM generos AS g, peliculas AS p
WHERE g.codigogenero=p.genero
GROUP BY g.nombregenero
PIVOT MONTH(fechadquisicion);
```

2. Realiza una consulta que nos muestre los generos de películas por filas y por columnas los tipos de películas, con la finalidad de saber el promedio de los precios de las películas que tenemos.

```
TRANSFORM AVG(p.precio)
SELECT g.nombregenero
FROM generos AS g, peliculas AS p, tipospelicula AS t
WHERE g.codigogenero=p.genero AND p.tipopelicula=t.codigoentrega
GROUP BY g.nombregenero
PIVOT (t.modalidad);
```

CONSULTAS DE ACCION

- 1 Borrar todos los registros de clientes que residan fuera de Zaragoza.

```
DELETE *  
FROM clientes  
WHERE ciudad NOT LIKE 'zaragoza';
```

- 2 Borrar todos los registros de Películas cuyo precio sea inferior a 15 €.

```
DELETE *  
FROM peliculas  
WHERE precio <15;
```

- 3 Borrar todos los registros de películas que empiecen por H.

```
DELETE *  
FROM peliculas  
WHERE título LIKE 'H*';
```

- 4 Borrar todos los registros de películas cuya modalidad sea ESTRENO

```
DELETE *  
FROM tipospelicula  
WHERE modalidad="Estreno";
```

- 5 Borrar todos los registros de películas del género de TERROR

```
DELETE *  
FROM generos  
WHERE nombregenero="Terror";
```

- 6 Borrar todos los registros de películas cuyo género sea AVENTURAS y hayan sido adquiridas en el año 98

```
DELETE P.*  
FROM peliculas AS p INNER JOIN generos AS g ON p.genero=g.codigogenero  
WHERE g.nombregenero="Acción" AND YEAR(P.fechadquisicion)=1998;
```

- 7 Añadir un registro nuevo en Generos cuyo numero sea 14 y se denomine DOCUMENTAL

```
INSERT INTO generos ( codigogenero, nombregenero )  
VALUES (17, "Documental");
```

- 8 Añadir un registro nuevo en la tabla de clientes cuya información corresponda a vuestros datos personales.

```
INSERT INTO clientes ( nombrecliente )  
VALUES ("Merche");
```

- 9 Crear una tabla vacía (llamada CopiaGeneros) con los mismos campos de la tabla de Generos. Traspasar toda la información de Generos a CopiaGeneros.

```
INSERT INTO copiageneros
SELECT *
FROM Generos;
```

- 10 Eliminar los registros de la tabla de CopiaGeneros cuyo nombre comience por C

```
DELETE *
FROM copiageneros
WHERE nombregenero LIKE "C*";
```

- 11 Añadir a la tabla CopiaGeneros los registros de Generos cuyo nombre comience por C

```
INSERT INTO copiageneros
SELECT *
FROM generos
WHERE nombregenero LIKE "C*";
```

- 12 Crear una tabla vacía (llamada Infantiles) con los mismos campos de la tabla de Peliculas. Traspasar todos los registros de la tabla Peliculas a la tabla Infantiles, que tengan como genero Infantil, Aventuras, Ciencia-ficción

```
SELECT p.* INTO infantiles
FROM peliculas AS p, generos AS g
WHERE p.genero=g.codigogenero AND (g.nombregenero="Aventuras" OR
g.nombregenero="Infantil" OR g.nombregenero="Ciencia-ficción");
```

- 13 Dividir la tabla de clientes en dos tablas llamadas Capital y Provincias con la misma estructura, en la primera guardaremos todos los registros de clientes que sean de Zaragoza y en Provincias el resto.

```
SELECT * INTO capital
FROM clientes
WHERE ciudad="Zaragoza";
```

- 14 Modificar el campo deCodigo Postal de la tabla de clientes para que a todos les aparezca 50900

```
UPDATE clientes SET codigopostal = "50900";
```

- 15 Modificar el campo de Observaciones de la tabla de clientes para que a todos les ponga un CODIGO formado por 3 caracteres de la izda del nombre + los 2 ultimos del 2 apellido+ 3 digitos centrales del telefono

```
UPDATE clientes SET observaciones =  
LEFT(nombrecliente,3)+RIGHT(apellido2cliente,2)+MID(telefono,4,3);
```

- 16 Modificar el campo de Observaciones de la tabla de clientes para que a todos los que se dieron de alta en el mes de Abril del 99 les aparezca el mensaje de BONIFICADO

```
UPDATE clientes SET observaciones = "Bonificado"  
WHERE MONTH(fechadquisicion)=4 AND YEAR(fechadquisicion)=1999;
```

- 17 Modificar el campo de Ciudad de la tabla de clientes para que todos los que residan en Zaragoza les aparezca la ciudad en mayúsculas.

```
UPDATE clientes SET ciudad = UCASE(ciudad)  
WHERE ciudad="Zaragoza";
```

- 18 Modificar el título de películas para que en todas que empiecen por R les aparezca -----

- 19 Incrementar el precio de cada película un cinco por ciento.

```
UPDATE peliculas SET precio = precio * 1.05;
```

- 20 Acentuar el apellido de López en la tabla de Clientes.

```
UPDATE clientes SET apellido1cliente = "López"  
WHERE apellido1cliente="Lopez";
```