CONSULTAS

#>/<>

HACK A BOSS

<CODE YOUR TALENT>

“La tecnología, bien utilizada, es uno de los mayores catalizadores sociales que han existido nunca”

2020 EDITION

ÍNDICE

CONSULTAS

Modelo Relacional

Una vez seleccionado el SGDB

- Convertir el Modelo conceptual en una estructura dependiente del SGDB.
- En BD relacionales: Modelo Relacional
- Definir las tablas, campos y restricciones
- Definir las relaciones
- Resolver posibles problemas

SELECT

La sentencia SELECT permite recuperar los datos

```
SELECT column1, column2, ...
FROM table_name;
```

```
SELECT DISTINCT column1, column2, ...
FROM table_name;
```

```
SELECT CustomerName, City
FROM Customers;
```

```
SELECT DISTINCT Country
FROM Customers;
```

Sintaxis completa

- SELECT ---> ¿Qué queremos?
- FROM -----> ¿De dónde?
- WHERE ----> ¿Bajo qué condiciones?
- GROUP BY -> ¿Hacemos grupos?
- HAVING ---> ¿Condiciones sobre los grupos?
- ORDER BY -> ¿En qué orden?

https://www.w3schools.com/sql/exercise.asp?filename=exercise_select1

WHERE

La cláusula WHERE nos permite filtrar resultados

```
SELECT column1, column2, ...
FROM table name
WHERE condition;
```

```
SELECT * FROM Customers
WHERE Country='Mexico';
```

```
SELECT * FROM Customers
WHERE CustomerID=1;
```

```
SELECT column1, column2, ...
FROM table name
WHERE NOT condition;
```

```
SELECT * FROM Customers
WHERE NOT Country='Germany';
```

```
SELECT * FROM Customers
WHERE NOT Country='Germany' AND NOT Country='USA';
```

WHERE

```
SELECT column1, column2, ...
FROM table name
WHERE condition1 AND condition2 AND
condition3...;
```

```
SELECT * FROM Customers
WHERE Country='Germany' AND
City='Berlin';
```

```
SELECT column1, column2, ...
FROM table name
WHERE condition1 OR condition2 OR
condition3…;
```

```
SELECT * FROM Customers
WHERE Country='Germany' OR
Country='Spain';
```

```
SELECT * FROM Customers
WHERE Country='Germany' AND (City='Berlin' OR City='München');
```

WHERE

Operator	Description
=	Equal
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
<>	Not equal. **Note:** In some versions of SQL this operator may be written as !=
BETWEEN	Between a certain range
LIKE	Search for a pattern
IN	To specify multiple possible values for a column

https://www.w3schools.com/sql/exercise.asp?filename=exercise_where1

LIKE

El operador LIKE se usa con la cláusula WHERE para buscar un patrón en una columna

```sql
SELECT column1, column2, ...
FROM table name
WHERE columnN LIKE pattern;
```

```sql
SELECT * FROM Customers
WHERE CustomerName LIKE 'a%';
```

LIKE

LIKE Operator	Description
WHERE CustomerName LIKE 'a%'	Finds any values that start with "a"
WHERE CustomerName LIKE '%a'	Finds any values that end with "a"
WHERE CustomerName LIKE '%or%'	Finds any values that have "or" in any position
WHERE CustomerName LIKE '_r%'	Finds any values that have "r" in the second position
WHERE CustomerName LIKE 'a__%'	Finds any values that start with "a" and are at least 3 characters in length
WHERE ContactName LIKE 'a%o'	Finds any values that start with "a" and ends with "o"

https://www.w3schools.com/sql/exercise.asp?filename=exercise_like1

ORDER BY

Nos permite ordenar los resultados en orden ascendiente o descendiente

```
SELECT column1, column2, ...
FROM table name
ORDER BY column1, column2, ...
ASC|DESC;
```

```
SELECT * FROM Customers
ORDER BY Country;
```

```
SELECT * FROM Customers
ORDER BY Country DESC;
```

```
SELECT * FROM Customers
ORDER BY Country, CustomerName;
```

```
SELECT * FROM Customers
ORDER BY Country ASC, CustomerName DESC;
```

ALIAS

Nos permite dar otro nombre a un campo en nuestro resultado

```
SELECT column name AS alias_name
FROM table_name;
```

```
SELECT CustomerID AS ID,
CustomerName AS Customer
FROM Customers;
```

```
SELECT CustomerName,
CONCAT(Address,', ',PostalCode,', ',City,',
',Country) AS Address
FROM Customers;
```

```
SELECT column name(s)
FROM table_name AS alias_name;
```

```
SELECT o.OrderID, o.OrderDate, c.CustomerName
FROM Customers AS c, Orders AS o
WHERE c.CustomerName="Around the Horn"
AND c.CustomerID=o.CustomerID;
```

https://www.w3schools.com/sql/exercise.asp?filename=exercise_alias1

LIMIT

Podemos limitar el número de resultados o establecer donde queremos empezar a ver resultados, o ambos.

```
SELECT * FROM Customers LIMIT 50;

SELECT * FROM Customers LIMIT 50 OFFSET 50;

SELECT * FROM Customers LIMIT 50, 50;

SELECT * FROM Customers OFFSET 50;
```

NULL

Podemos comprobar si un valor es NULL o no, pero no podemos operar con valores null.
Cualquier operación con un valor null no funcionará.
Podemos sustituir los valores null por otro valor.

```sql
SELECT * FROM Customers
WHERE Country IS NULL;
```

```sql
SELECT * FROM Customers
WHERE Country IS NOT NULL;
```

```sql
SELECT ProductName, UnitPrice * (UnitsInStock +
COALESCE(UnitsOnOrder, 0))
FROM Products;
```

FUNCIONES

Podemos usar funciones para obtener resultados manipulados de nuestras tablas.

```
SELECT MIN(column_name)
FROM table name
WHERE condition;
```

```
SELECT COUNT(column_name)
FROM table name
WHERE condition;
```

```
SELECT MAX(column_name)
FROM table name
WHERE condition;
```

```
SELECT AVG(column_name)
FROM table name
WHERE condition;
```

```
SELECT SUM(column_name)
FROM table name
WHERE condition;
```

https://www.w3schools.com/sql/exercise.asp?filename=exercise_functions1

FUNCIONES

También hay funciones para texto, fechas, entre otras.

CURRENT_DATE() → fecha actual
CURRENT_TIME() → hora

UPPER --> Mayúsculas
LOWER --> Minúsculas

SUBSTR(string, start, length) -> Substring
CONCAT(a, b) -> Concatenar

https://dev.mysql.com/doc/refman/5.7/en/func-op-summary-ref.html

#>/<>
HACK A BOSS
<CODE YOUR TALENT>

#<THANX!>

#>/<>

HACK A BOSS

<CODE YOUR TALENT>

+34 919 04 23 63

www.hackaboss.com

Av.Linares Rivas 50-51, 15005, A Coruña

2020 EDITION