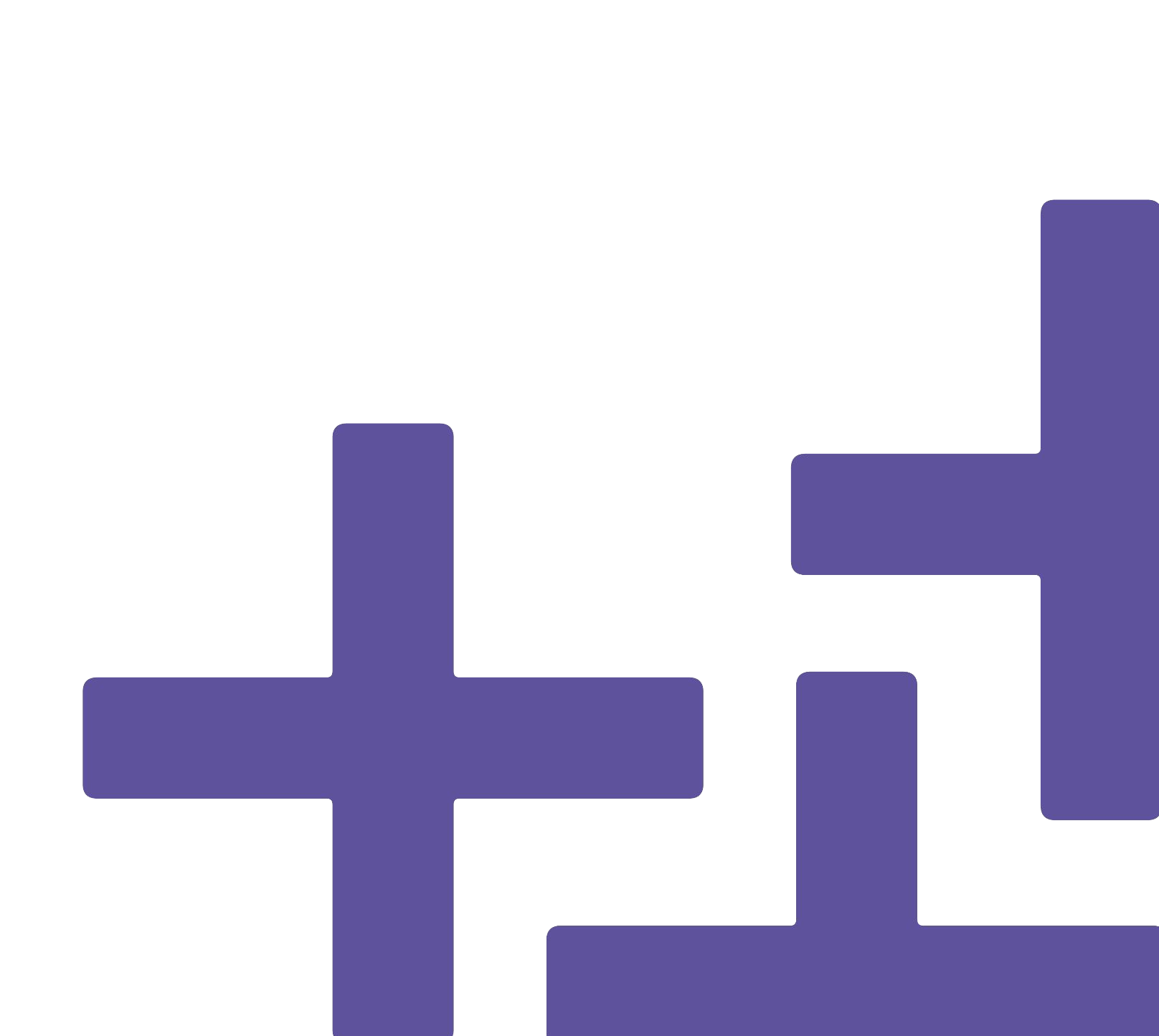
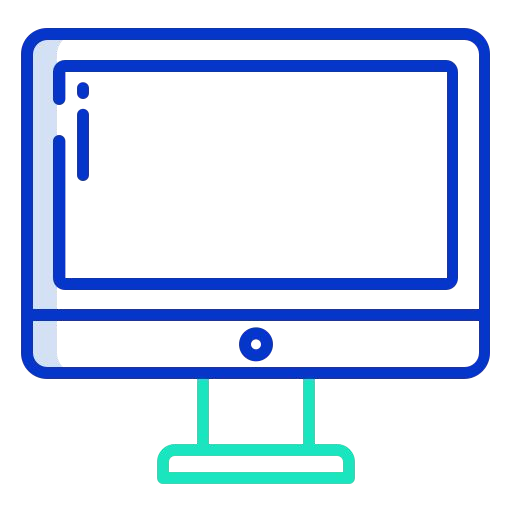


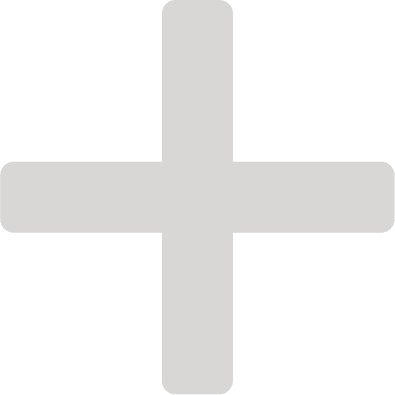
Socio estratégico Impulsan

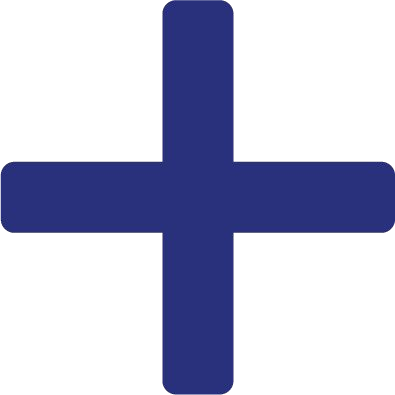
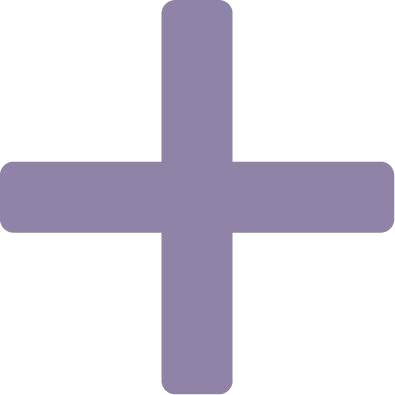
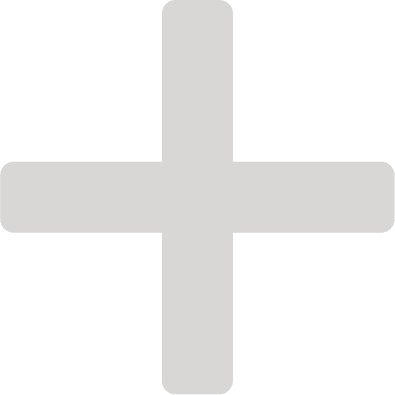


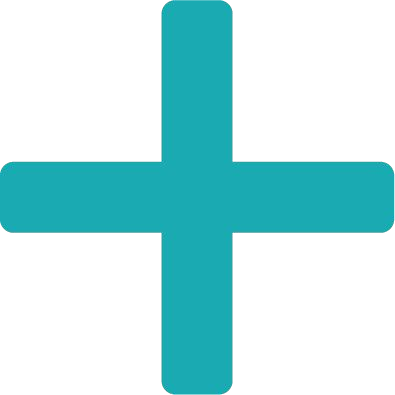
## Cobol – Clase 23

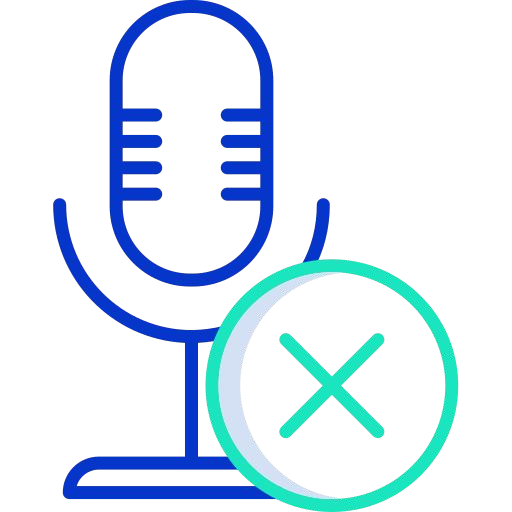
SQL (Parte 2)

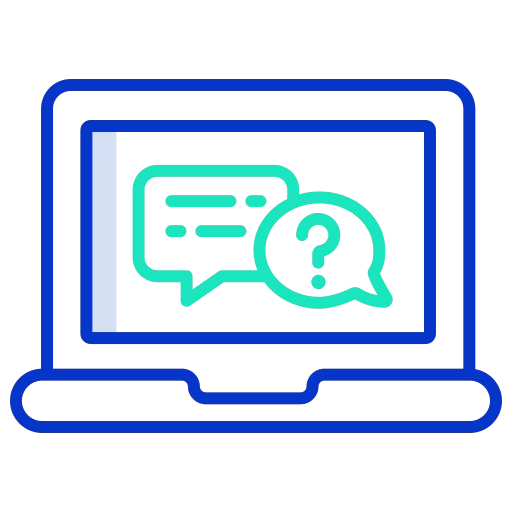




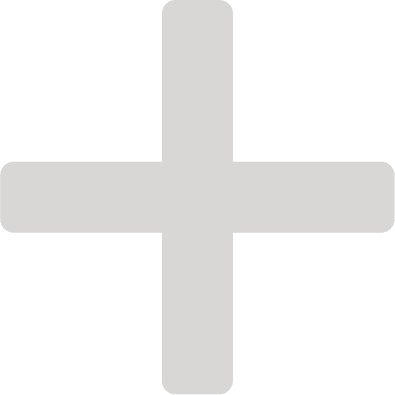
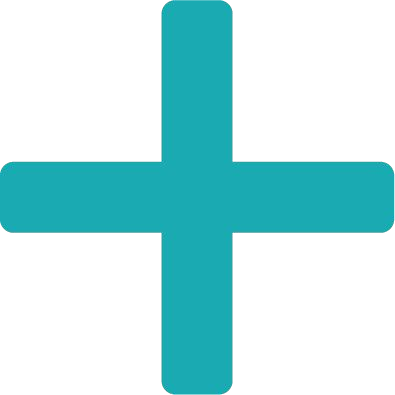
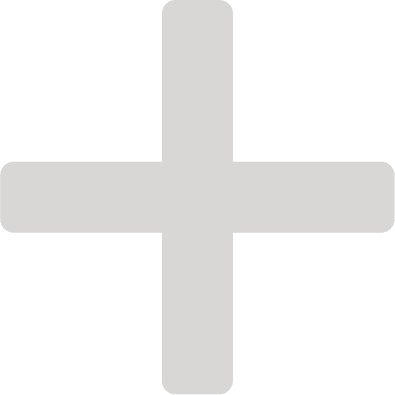


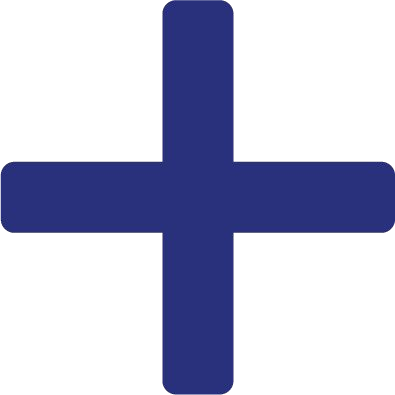
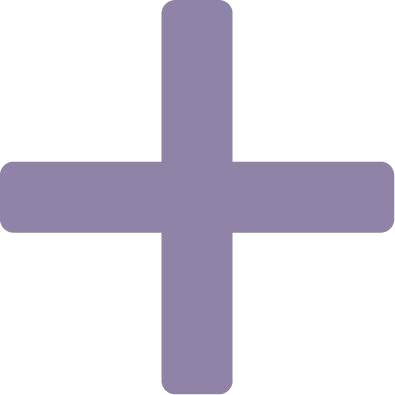
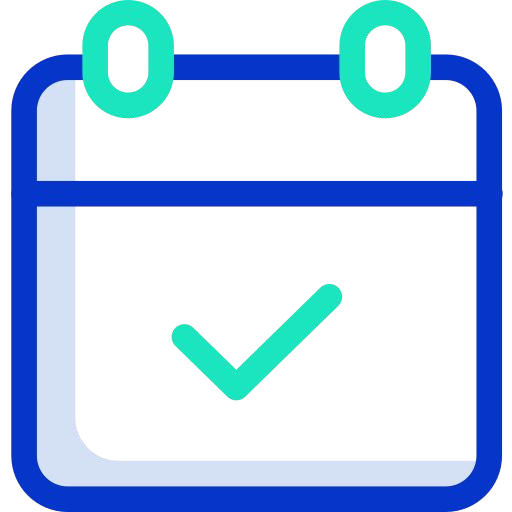
**Reglas de la clase**

 Micrófonos apagados

 Consultas al final de la clase Consultas por chat





**Cronograma**



**Primera Parte**

18:30

a 19:25

**Break**

19:25

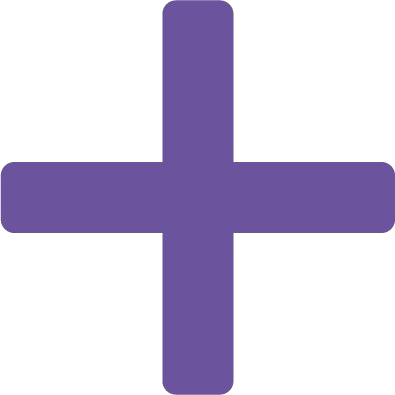
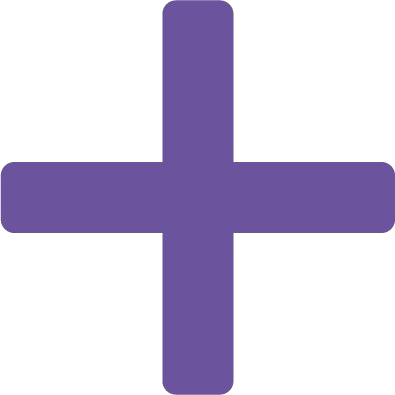
a 19:35

**Segunda Parte**

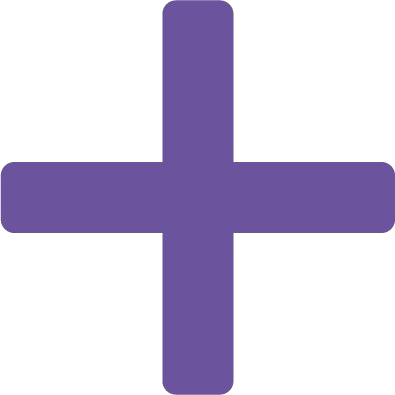
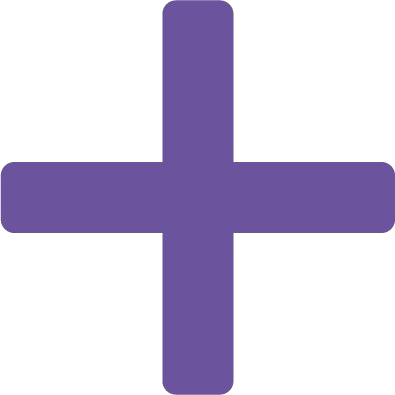
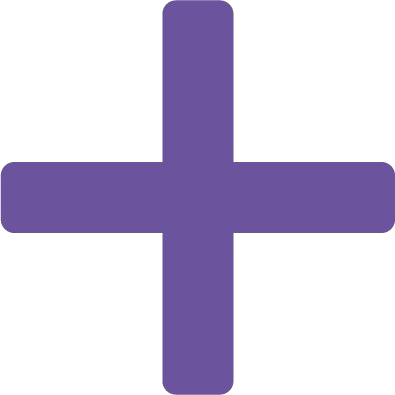
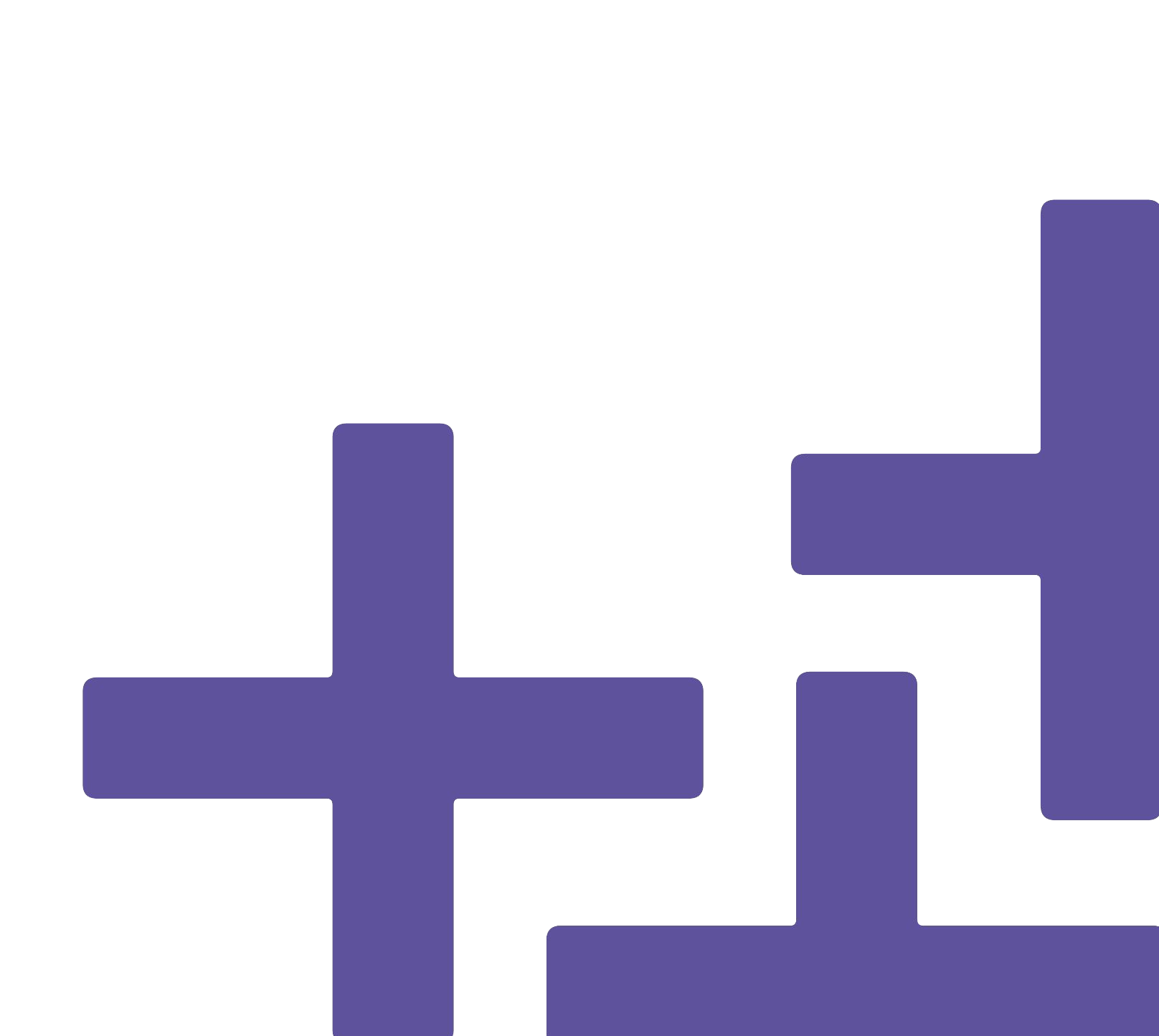
19:35

a 20:30

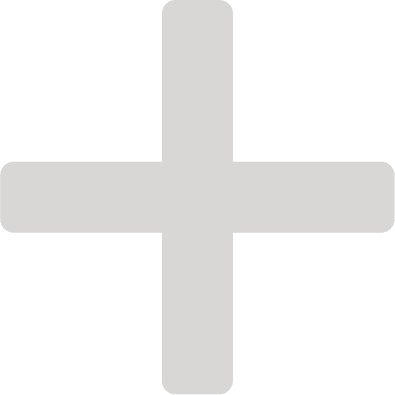
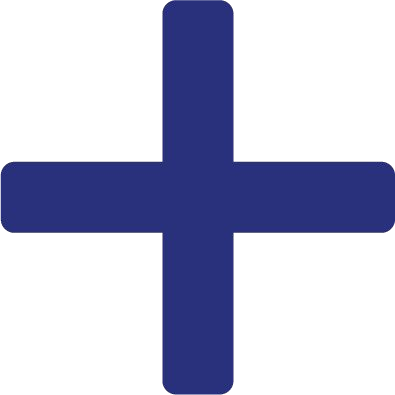
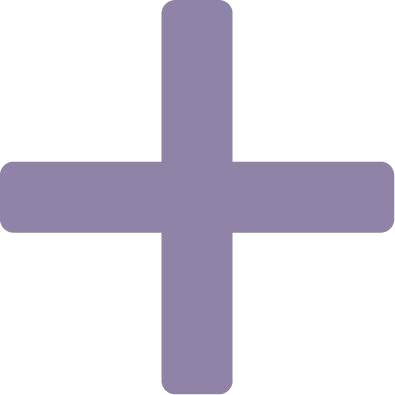
**¿Qué veremos hoy?**

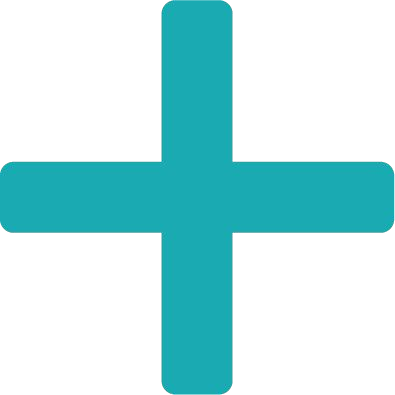
Funciones JOIN

UPDATE

 INSERT DELETE







**SQL**

Structured Query Language

**DDL**

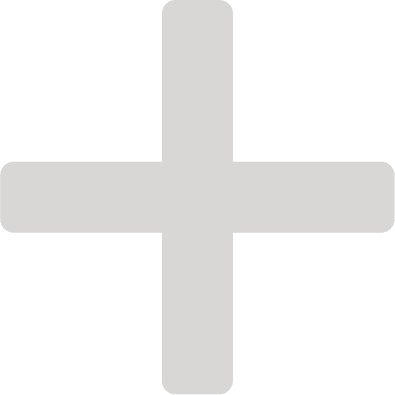
Data Definition Language

**DCL**

Data Control Language

**DML**

Data Manipulation Language

* **SELECT**
* **INSERT**

Roles y permisos, control de acceso

Permite recuperar, almacenar, modificar, eliminar e insertar datos

Crear, eliminar y modificar la estructura de la base de datos y de las tablas

* **UPDATE**
* **DELETE**

## Funciones

#### MIN()

Obtiene el valor mínimo

de una columna

SELECT MIN(ID) FROM ALUMNOS

TABLA ALUMNOS

|  |  |
| --- | --- |
| **ID** | **NOMBRE** |
| 1 | EDUARDO |
| 2 | ANA |
| 3 | MARIA |
| 9 | JOSE |
| 10 | PABLO |

#### MAX()

Obtiene el valor máximo de una columna

SELECT MAX(ID) FROM ALUMNOS

**1**

**10**



## Funciones

#### COUNT() SUM() AVG()

TABLA EMPLEADOS

|  |  |
| --- | --- |
| **ID** | **VALOR** |
| 1 | 10 |
| 2 | 50 |
| 3 | 87 |
| 9 | 99 |
| 10 | 4 |

Obtiene la cantidad de registros de una columna

SELECT COUNT(VALOR) FROM EMPLEADOS

Sumaria los valores de una columna

SELECT SUM(VALOR) FROM EMPLEADOS

Calcula el promedio de los valores de una columna

SELECT AVG(VALOR) FROM EMPLEADOS

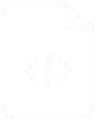
**5**

**250**

**50**



### JOIN

Se utiliza para combinar filas de dos o más tablas basándose en una/s columna/s común entre ellas.



# JOIN



SELECT T1.ID, T2.DETALLE FROM TABLA1 AS T1

INNER JOIN TABLA2 AS T2 ON T2.ID = T1.ID;

**TABLA1**

**TABLA2**

# JOIN

TABLA EMPLEADOS TABLA DIRECCIONES

|  |  |  |
| --- | --- | --- |
| **ID** | **NOMBRE** |  |
| 1 | EDUARDO |
|  |
| 2 | ANA |
|  |
| 3 | MARIA |
|  |
| 9 | JOSE |
| 10 | PABLO |

|  |  |  |
| --- | --- | --- |
| **ID** | **CALLE** | **NÚMERO** |
| 1 | CÓRDOBA | 12 |
| 2 | MARCONI | 4553 |
| 3 | ALCORTA | 565 |
| 6 | LIBERTADOR | 99 |
| 9 | RIVADAVIA | 787 |

SELECT E.ID, E.NOMBRE, D.CALLE, D.NUMERO FROM EMPLEADOS AS E

INNER JOIN DIRECCIOES AS D ON D.ID = E.ID;



|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NOMBRE** | **CALLE** | **NÚMERO** |
| 1 | EDUARDO | CÓRDOBA | 12 |
| 2 | ANA | MARCONI | 4553 |
| 3 | MARIA | ALCORTA | 565 |
| 9 | JOSE | RIVADAVIA | 787 |

# JOIN

TABLA EMPLEADOS TABLA DIRECCIONES

|  |  |  |
| --- | --- | --- |
| **ID** | **NOMBRE** |  |
| 1 | EDUARDO |
| 2 | ANA |
| 3 | MARIA |
|  |
| 9 | JOSE |
| 10 | PABLO |

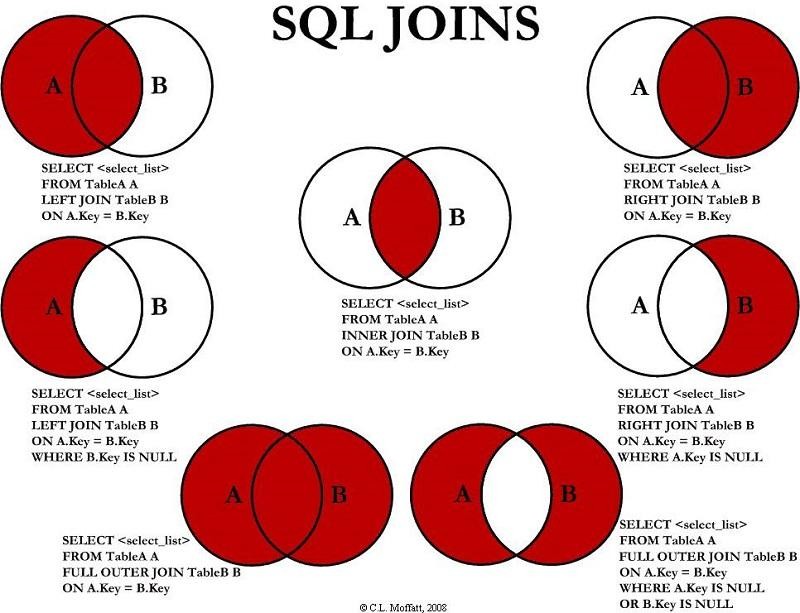
|  |  |  |
| --- | --- | --- |
| **ID** | **CALLE** | **NÚMERO** |
| 1 | CÓRDOBA | 12 |
| 2 | MARCONI | 4553 |
| 3 | ALCORTA | 565 |
| 6 | LIBERTADOR | 99 |
| 9 | RIVADAVIA | 787 |

|  |  |  |
| --- | --- | --- |
| **SELECT** | **E.ID, E.NOMBRE,** | **D.CALLE, D.NUMERO** |
| **FROM** | **EMPLEADOS AS E** |  |
| **INNER** | **JOIN DIRECCIOES** | **AS D ON D.ID = E.ID** |
| **WHERE** | **E.ID = 3;** |  |

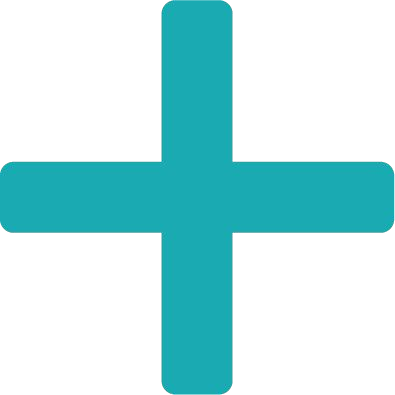
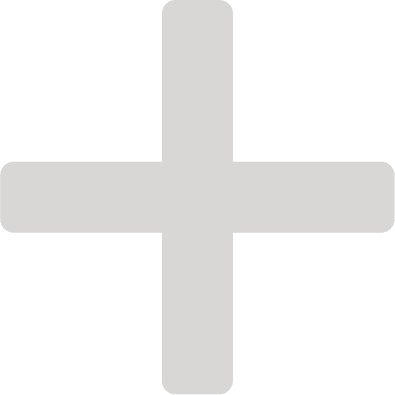


|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NOMBRE** | **CALLE** | **NÚMERO** |
| 3 | MARIA | ALCORTA | 565 |

# JOIN



**INNER JOIN = JOIN**

**JOIN**



Mejor rendimiento

SELECT E.ID, E.NOMBRE, D.CALLE, D.NUMERO FROM EMPLEADOS AS E

INNER JOIN DIRECCIOES AS D ON D.ID = E.ID;

=

Iguales resultado

SELECT E.ID, E.NOMBRE, D.CALLE, D.NUMERO FROM EMPLEADOS AS E

, DIRECCIOES AS D WHERE D.ID = E.ID;



### INSERT

Agrega registros a una tabla

### UPDATE

Actualiza datos

### DELETE

Elimina registros



# INSERT



INSERT INTO tabla (campo1, campo2, campo3, campo4, campo5) VALUES (valor1, valor2, valor3, valor4, valor5);

INSERT INTO tabla VALUES (valor1, valor2, valor3, valor4, valor5);



# INSERT

INSERT INTO tabla VALUES

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **(valor1,** | **valor2,** | **valor3,** | **valor4,** | **valor5),** |
| **(valor1,** | **valor2,** | **valor3,** | **valor4,** | **valor5),** |
| **(valor1,** | **valor2,** | **valor3,** | **valor4,** | **valor5),** |
| **(valor1,** | **valor2,** | **valor3,** | **valor4,** | **valor5);** |

INSERT INTO tabla

SELECT campo1, campo2, campo3, campo4, campo5 FROM tabla2 WHERE condición;



# UPDATE



UPDATE tabla1

SET campo1 = valor1, campo2 = valor2

WHERE campo1 = valorA AND campo5 = valorB;

UPDATE tabla1

SET campo1Tabla1 = valor1Tabla2 FROM tabla2

WHERE campo2Tabla1 = campo2Tabla2;

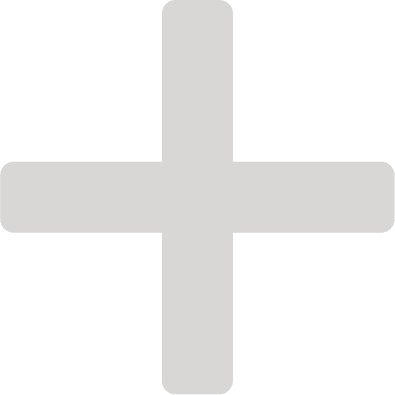
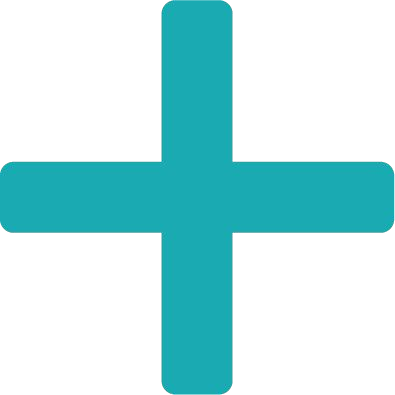
UPDATE tabla1

SET campo1 = valor1, campo2 = valor2;

**¡OJO! Que si no se incluye el WHERE actualizará todos los registros de la tabla.**

# DELETE

DELETE FROM tabla WHERE campo1 = valor1;



DELETE FROM tabla;

**¡OJO! Que si no se incluye el WHERE actualizará todos los registros de la tabla.**





## Comunicación

Foro de consultas TEC:

https://campus.soysilvertech.org

**Mails de consulta TEC**: [consultasCOBOL@soysilvertech.org](mailto:consultasCOBOL@soysilvertech.org)

