



# Trabajo 12 - SQL

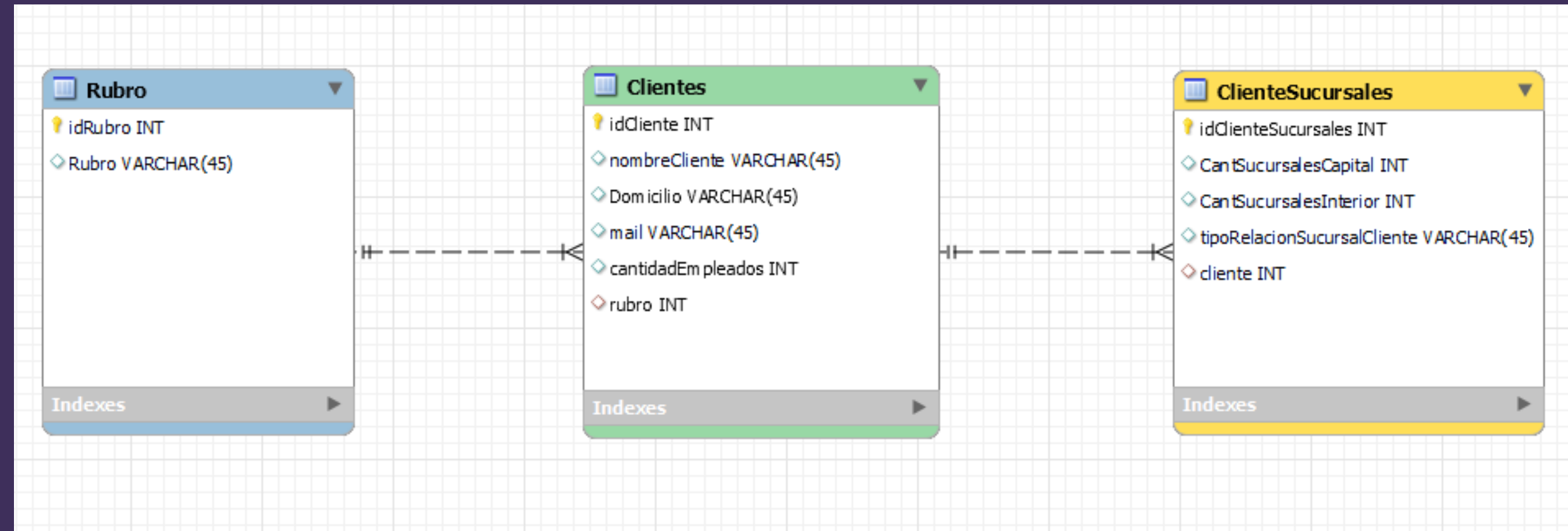
## Bases de Datos

### 2023

GRUPO 9:

- CEJAS, LUCAS
- MARTIN, DENISE
- NAVA, ALEJANDRO
- POLESEL, FRANCO
- TEXEIRA, JUANSE

# Creado de las tablas en el Diagrama



# Creado de las tablas en el código



```
-- Table `mydb`.`Rubro`
```

```
CREATE TABLE IF NOT EXISTS `grupo9`.`Rubro` (  
  `idRubro` INT NOT NULL,  
  `Rubro` VARCHAR(45) NULL,  
  PRIMARY KEY (`idRubro`),  
  UNIQUE INDEX `idRubro_UNIQUE` (`idRubro` ASC) VISIBLE)  
ENGINE = InnoDB;
```

```
-- Table `mydb`.`Clientes`
```

```
CREATE TABLE IF NOT EXISTS `grupo9`.`Clientes` (  
  `idCliente` INT NOT NULL,  
  `nombreCliente` VARCHAR(45) NULL,  
  `Domicilio` VARCHAR(45) NULL,  
  `mail` VARCHAR(45) NULL,  
  `cantidadEmpleados` INT NULL,  
  `rubro` INT NULL,  
  PRIMARY KEY (`idCliente`),  
  INDEX `rubro_idx` (`rubro` ASC) VISIBLE,  
  CONSTRAINT `rubro`  
    FOREIGN KEY (`rubro`)  
    REFERENCES `grupo9`.`Rubro` (`idRubro`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
-- Table `mydb`.`ClienteSucursales`
```

```
CREATE TABLE IF NOT EXISTS `grupo9`.`ClienteSucursales` (  
  `idClienteSucursales` INT NOT NULL,  
  `CantSucursalesCapital` INT NULL,  
  `CantSucursalesInterior` INT NULL,  
  `tipoRelacionSucursalCliente` VARCHAR(45) NULL,  
  `cliente` INT NULL,  
  PRIMARY KEY (`idClienteSucursales`),  
  UNIQUE INDEX `idClienteSucursales_UNIQUE` (`idClienteSucursales` ASC) VISIBLE,  
  INDEX `cliente_idx` (`cliente` ASC) VISIBLE,  
  CONSTRAINT `cliente`  
    FOREIGN KEY (`cliente`)  
    REFERENCES `grupo9`.`Clientes` (`idCliente`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

# Inserción de datos



```
• INSERT INTO Rubro (idRubro, Rubro)
VALUES
(1, 'Electricidad'),
(2, 'Telecomunicaciones'),
(3, 'Redes'),
(4, 'Telefonía IP'),
(5, 'Miscelaneas');
```

```
• INSERT INTO Clientes (idCliente, nombreCliente, Domicilio, mail, cantidadEmpleados, Rubro)
VALUES
(2, 'ElectroTodo', 'Virrey del Pino 1434', 'et@yahoo.com', 5, 1),
(3, 'Conectar S.A.', 'Agüero 37', 'conectar@fibertel.com', 10, 2),
(4, 'Metro SRL', 'Cerrito esq San Juan', '-', 12, 3),
(5, 'MyIP', 'Salguero 3421', 'MIP@hotmail.com', 3, 2),
(6, 'Router & Switch', 'Las Heras esq Austria', 'RS@google.com', 4, NULL),
(7, 'Sat', 'Cabildo esq Juramento', '-', 1, NULL),
(8, 'Cableado SRL', 'Guemes 234', NULL, 7, 4);
```

```
INSERT INTO ClienteSucursales(idClienteSucursales, CantSucursalesCapital, CantSucursalesInterior, tipoRelacionSucursalCliente, cliente)
VALUES
(2, 30, 4, 'Principal', 2),
(3, 0, 12, 'Principal', 3),
(5, 9, 0, 'Secundaria', 4),
(6, 25, 0, 'Secundaria', 5),
(7, 0, 2, 'Terciaria', 6);
```

Un listado con todos los datos de todos los clientes



	idCliente	nombreCliente	Domicilio	mail	cantidadEmpleados	rubro
▶	2	ElectroTodo	Virrey del Pino 1434	et@yahoo.com	5	1
	3	Conectar S.A.	Agüero 37	conectar@fibertel.com	10	2
	4	Metro SRL	Cerrito esq San Juan	-	12	3
	5	MyIP	Salguero 3421	MIP@hotmail.com	3	2
	6	Router & Switch	Las Heras esq Austria	RS@google.com	4	NULL
	7	Sat	Cabildo esq Juramento	-	1	NULL
	8	Cableado SRL	Guemes 234	NULL	7	4
	NULL	NULL	NULL	NULL	NULL	NULL

```
-- A)
SELECT * FROM Clientes;
```

Un listado de Clientes Sucursales, que muestre el idCliente y la CantidadSucursalesCapital, ordenado por CantidadSucursalesCapital de forma descendiente.



cliente	CantSucursalesCapital
2	30
5	25
4	9
3	0
6	0

```
-- B)
SELECT cliente, CantSucursalesCapital FROM ClienteSucursales
ORDER BY CantSucursalesCapital DESC;
```

Un listado con todos los datos de clientes y su rubro asociado, donde solo figuren clientes que tienen rubros.



nombreCliente	rubro
ElectroTodo	1
Conectar S.A.	2
MyIP	2
Metro SRL	3
Cableado SRL	4

```
-- C)
SELECT nombreCliente, rubro FROM Clientes
WHERE rubro IS NOT NULL;
```

Un listado de clientes indicando su nombre, rubro y cantidad de empleados, solo para aquellos clientes cuyo nombre comience con la letra M.



nombreCliente	rubro	cantidadEmpleados
Metro SRL	3	12
MyIP	2	3

```
-- D)
SELECT nombreCliente, rubro, cantidadEmpleados FROM Clientes
WHERE nombreCliente LIKE 'M%';
```



Un listado con todos los datos de clientes y su rubro asociado, donde figuren clientes que tienen rubros y también aquellos clientes que no tienen rubro asociado.



idCliente	nombreCliente	Domicilio	mail	cantidadEmpleados	rubro
6	Router & Switch	Las Heras esq Austria	RS@google.com	4	NULL
7	Sat	Cabildo esq Juramento	-	1	NULL
NULL	NULL	NULL	NULL	NULL	NULL

```
-- E)
```

```
SELECT * FROM Clientes  
WHERE rubro IS NULL;
```

```
SELECT * FROM Clientes  
WHERE rubro IS NOT NULL;
```

Un listado con todos los datos de clientes y su rubro asociado, donde figuren clientes que tienen rubros y también aquellos rubros que no tienen están asociado a ningún cliente.



idCliente	nombreCliente	Domicilio	mail	cantidadEmpleados	rubro
2	ElectroTodo	Virrey del Pino 1434	et@yahoo.com	5	1
3	Conectar S.A.	Agüero 37	conectar@fibertel.com	10	2
5	MyIP	Salguero 3421	MIP@hotmail.com	3	2
4	Metro SRL	Cerrito esq San Juan	-	12	3
8	Cableado SRL	Guemes 234	NULL	7	4
NULL	NULL	NULL	NULL	NULL	NULL

```
-- F)
SELECT c.idCliente, c.nombreCliente, c.Domicilio, c.mail, c.cantidadEmpleados, r.Rubro, r.idRubro FROM Clientes c
LEFT JOIN Rubro r ON c.rubro = r.idRubro
UNION
SELECT null, null, null, null, null, r.Rubro, r.idRubro FROM Rubro r
LEFT JOIN Clientes c ON c.rubro = r.idRubro
WHERE c.idCliente IS NULL;
```

Un listado con todos los datos de clientes y su rubro asociado, clientes sin rubros y rubros sin clientes.



idCliente	nombreCliente	Domicilio	mail	cantidadEmpleados	Rubro	idRubro
2	ElectroTodo	Virrey del Pino 1434	et@yahoo.com	5	Electricidad	1
3	Conectar S.A.	Agüero 37	conectar@fibertel.com	10	Telecomunicaciones	2
4	Metro SRL	Cerrito esq San Juan	-	12	Redes	3
5	MyIP	Salguero 3421	MIP@hotmail.com	3	Telecomunicaciones	2
6	Router & Switch	Las Heras esq Austria	RS@google.com	4	NULL	NULL
7	Sat	Cabildo esq Juramento	-	1	NULL	NULL
8	Cableado SRL	Guemes 234	NULL	7	Telefonia IP	4
NULL	NULL	NULL	NULL	NULL	Miscelaneas	5

```
-- G)
SELECT c.idCliente, c.nombreCliente, c.Domicilio, c.mail, c.cantidadEmpleados, r.Rubro, r.idRubro FROM Clientes c
INNER JOIN Rubro r ON c.rubro = r.idRubro
UNION
SELECT c.idCliente, c.nombreCliente, c.Domicilio, c.mail, c.cantidadEmpleados, null, null FROM Clientes c
LEFT JOIN Rubro r ON c.rubro = r.idRubro
WHERE r.idRubro IS NULL
UNION
SELECT null, null, null, null, null, r.Rubro, r.idRubro FROM Clientes c
RIGHT JOIN Rubro r ON c.rubro = r.idRubro
WHERE c.idCliente IS NULL;
```

Un listado con aquellos clientes que tengan rubro y cuya cantidad total de sucursales mayor a 15.



idCLiente	nombreCliente	rubro	Rubro	TotalSucursales
2	ElectroTodo	1	Electricidad	34
5	MyIP	2	Telecomunicaciones	25

```
-- H)
SELECT c.idCLiente, c.nombreCliente, c.rubro, r.Rubro,
       SUM(cs.CantSucursalesCapital + cs.CantSucursalesInterior) AS TotalSucursales
FROM Clientes c
INNER JOIN Rubro r ON c.rubro = r.idRubro
INNER JOIN ClienteSucursales cs ON c.idCLiente = cs.cliente
GROUP BY c.idCLiente
HAVING TotalSucursales > 15;
```

Un listado con aquellos clientes que tengan rubro o cuya cantidad total de sucursales mayor a 15.



idCliente	nombreCliente	rubro	Rubro	TotalSucursales
2	ElectroTodo	1	Electricidad	34
3	Conectar S.A.	2	Telecomunicaciones	12
4	Metro SRL	3	Redes	9
5	MyIP	2	Telecomunicaciones	25
8	Cableado SRL	4	Telefonía IP	NULL

```
-- I)
SELECT c.idCliente, c.nombreCliente, c.rubro, r.Rubro,
       SUM(cs.CantSucursalesCapital + cs.CantSucursalesInterior) AS TotalSucursales
FROM Clientes c
LEFT OUTER JOIN Rubro r ON c.rubro = r.idRubro
LEFT OUTER JOIN ClienteSucursales cs ON c.idCliente = cs.cliente
GROUP BY c.idCliente
HAVING TotalSucursales > 15 OR c.rubro IS NOT NULL;
```

Un listado de la cantidad de empleados (en los clientes), agrupados por rubro



Rubro	totalEmpleados
Electricidad	5
Telecomunicaciones	13
Redes	12
Telefonía IP	7
Miscelaneas	NULL

```
-- 3)
SELECT r.Rubro, SUM(c.cantidadEmpleados) AS totalEmpleados FROM Rubro r
LEFT JOIN Clientes c
ON r.idRubro = c.rubro
GROUP BY r.Rubro;
```

Un listado de clientes indicando su nombre, id de rubro y descripción de rubro donde solo figuren aquellos clientes cuyo rubro este en el conjunto de rubros conformados por los ids 1, 2 o 4.



idCliente	nombreCliente	rubro
2	ElectroTodo	1
3	Conectar S.A.	2
5	MyIP	2
8	Cableado SRL	4

```
-- K)
SELECT c.idCliente, c.nombreCliente, c.rubro FROM Clientes c
INNER JOIN Rubro r
ON r.idRubro = c.rubro
HAVING c.rubro IN (1,2,4);
```

Un listado indicando para cada tipo de relación de ClientesSucursales, la suma de sucursales de interior y el promedio de sucursales de capital, siempre y cuando el promedio de sucursales de capital se inferior a 16.



tipoRelacionSucursalCliente	CantSucursales	promCap
Principal	16	15.0000
Terciaria	2	0.0000

```
-- L)
SELECT cs.tipoRelacionSucursalCliente,
SUM(cs.CantSucursalesInterior) AS CantSucursales,
AVG(cs.CantSucursalesCapital) AS promCap FROM ClienteSucursales cs
GROUP BY cs.tipoRelacionSucursalCliente
HAVING promCap < 16;
```