

# Bases de Datos

## Carrera Programador full-stack

*Consultas Multi-Tabla*

# Consultas Multi-Tablas

- En SQL existe la posibilidad de realizar consultas de datos de más de una tabla al mismo tiempo. La palabra utilizada para unir tablas es **JOIN**.
- Esta operación permite combinar filas de datos de dos o más tablas basándose en columnas comunes entre ellas.

# INNER JOIN

- Por medio del operador **JOIN** podemos combinar dos tablas según una condición para obtener tuplas compuestas por atributos de las dos relaciones combinadas.
- Además de **INNER JOIN** que devuelve todas las filas que cumplen con la condición de igualdad de atributos expresada en la cláusula **ON**.

# 4 Tipos de Join

- **INNER JOIN** (también conocido como "simple" JOIN). Este es el tipo más común de JOIN.
- **LEFT JOIN** (o LEFT OUTER JOIN)
- **RIGHT JOIN** (o RIGHT OUTER JOIN)
- **FULL JOIN** (o FULL OUTER JOIN)

# Consultas Multi-Tablas

## *Ejemplo*

Seleccionar los datos de los clientes junto con sus teléfonos.

```
SELECT c.nombre, c.apellido, t.nro_telefono  
FROM e01_cliente c  
INNER JOIN e01_telefono t  
ON c.nro_cliente = t.nro_cliente;
```

nombre	apellido	nro_telefono
--------	----------	--------------

Xerxes	Hale	4998612
--------	------	---------

Xerxes	Hale	4241515
--------	------	---------

...

...

...

# Ejemplo Inner Join

```
SELECT account.*,  
       customer.name,  
       customer.lastname,  
       customer.gender,  
       customer.marital_status  
FROM account  
JOIN customer  
ON account.customer_id=customer.customer_id;
```

# Ejemplo Inner Join

account_id	overdraft_amount	type_id	segment	customer_id		customer_id	name	lastname	gender	gender	marital_status
2556889	12000	2	RET	4		1	MARC	TESCO	M	M	Y
1323598795	1550	1	RET	1		2	ANNA	MARTIN	F	F	N
2225546	5000	2	RET	5		3	EMMA	JOHNSON	F	F	Y
5516229	6000	5	RET	4		4	DARIO	PENTAL	M	M	N
5356222	7500	5	RET	5		5	ELENA	SIMSON	F	F	N
2221889	5400	2	RET	1		6	TIM	ROBITH	M	M	N
2455688	12500	2	CORP	50		7	MILA	MORRIS	F	F	N
1322488656	2500	1	CORP	51		8	JENNY	DWARTH	F	F	Y
1323598795	3100	1	CORP	52							
1323111595	1220	1	CORP	53							

- Los registros que comparten el mismo valor de ID de cliente se emparejan. (Los registros que no coinciden en ninguna de las dos tablas (mostrados en gris) no se incluyen en el conjunto de resultados.
- Para los registros que tienen una coincidencia, todos los atributos de la tabla **account** se muestran en el conjunto de resultados. Los atributos de nombre, apellido, sexo y estado civil de la tabla **customer** de la tabla.

# Ejemplo Inner Join

Después de ejecutar este código, SQL devuelve lo siguiente:

account_id	overdraft_amount	customer_id	type_id	segment	name
2556889	12000	4	2	RET	DARIO
1323598795	1550	1	1	RET	MARC
2225546	5000	5	2	RET	ELENA
5516229	6000	4	5	RET	DARIO
5356222	7500	5	5	RET	ELENA
2221889	5400	1	2	RET	MARC

lastname	gender	marital_status
PENTAL	M	N
TESCO	M	Y
SIMSON	F	N
PENTAL	M	N
SIMSON	F	N
TESCO	M	Y



# Consultas Multi-Tablas

## *Alias*

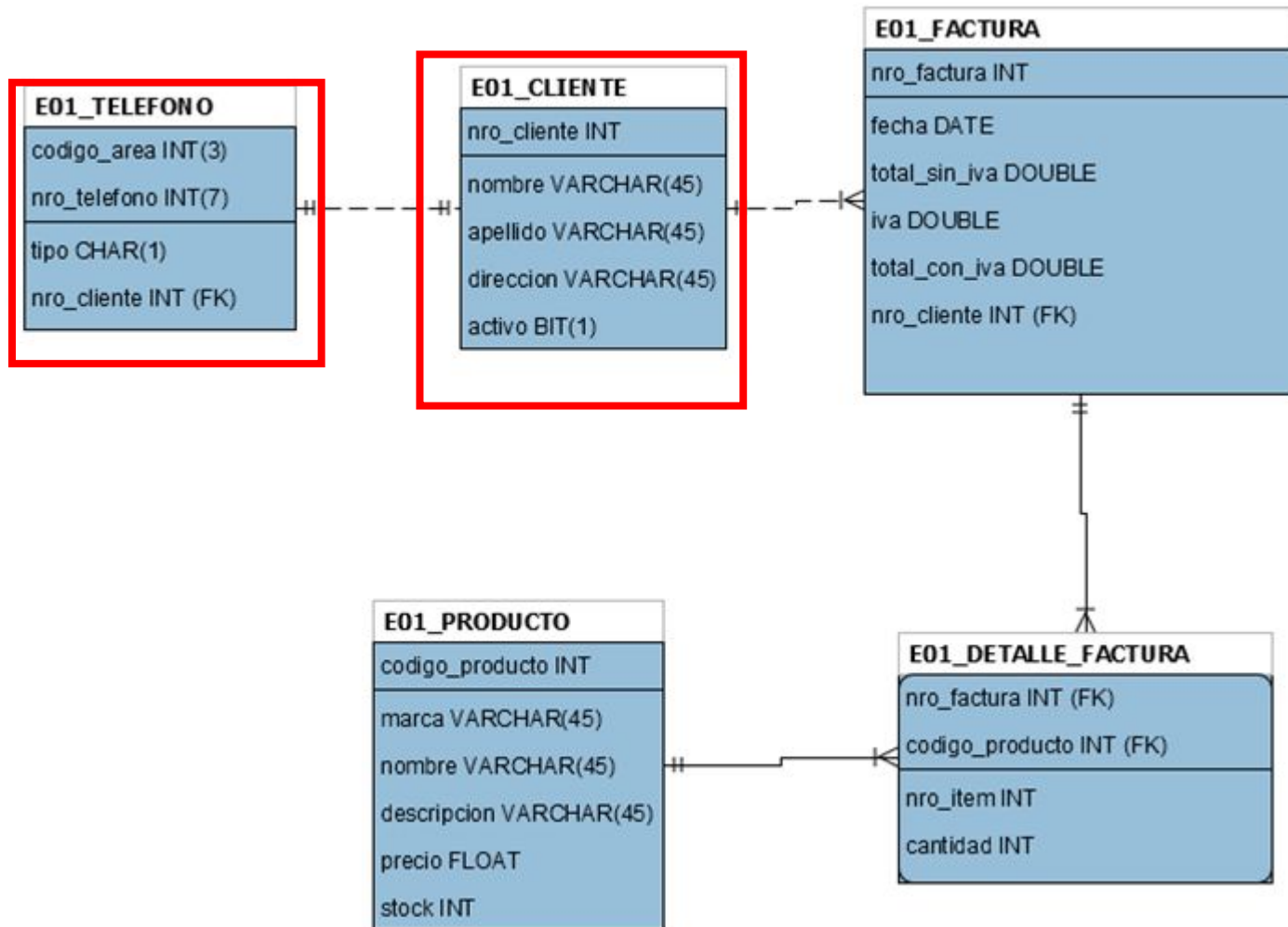
En el caso de las consultas multi-tablas es útil utilizar abreviaciones de nombres de tablas para llamarlas de forma más simple y breve. También puede darse apodos a los nombres de columnas.

```
SELECT nombre, apellido,  
       nro_telefono AS telefono  
FROM e01_cliente c  
     INNER JOIN e01_telefono t  
ON c.nro_cliente = t.nro_cliente;
```

Renombrado de columna:  
en vez de *nro\_telefono*,  
en los **resultados** dicha  
columna se llamará  
*telefono*.

Abreviación  
del nombre de  
la tabla

# Esquema



# LEFT JOIN

- Devuelve todas las filas de la tabla izquierda y las filas coincidentes de la tabla derecha.
  - Si existe alguna fila de la tabla izquierda que no tiene coincidencia con ninguna fila de la tabla derecha, dicha fila es mostrada en el resultado, pero aparecerá el valor NULL en las columnas de la tabla derecha ya que no existe coincidencia.

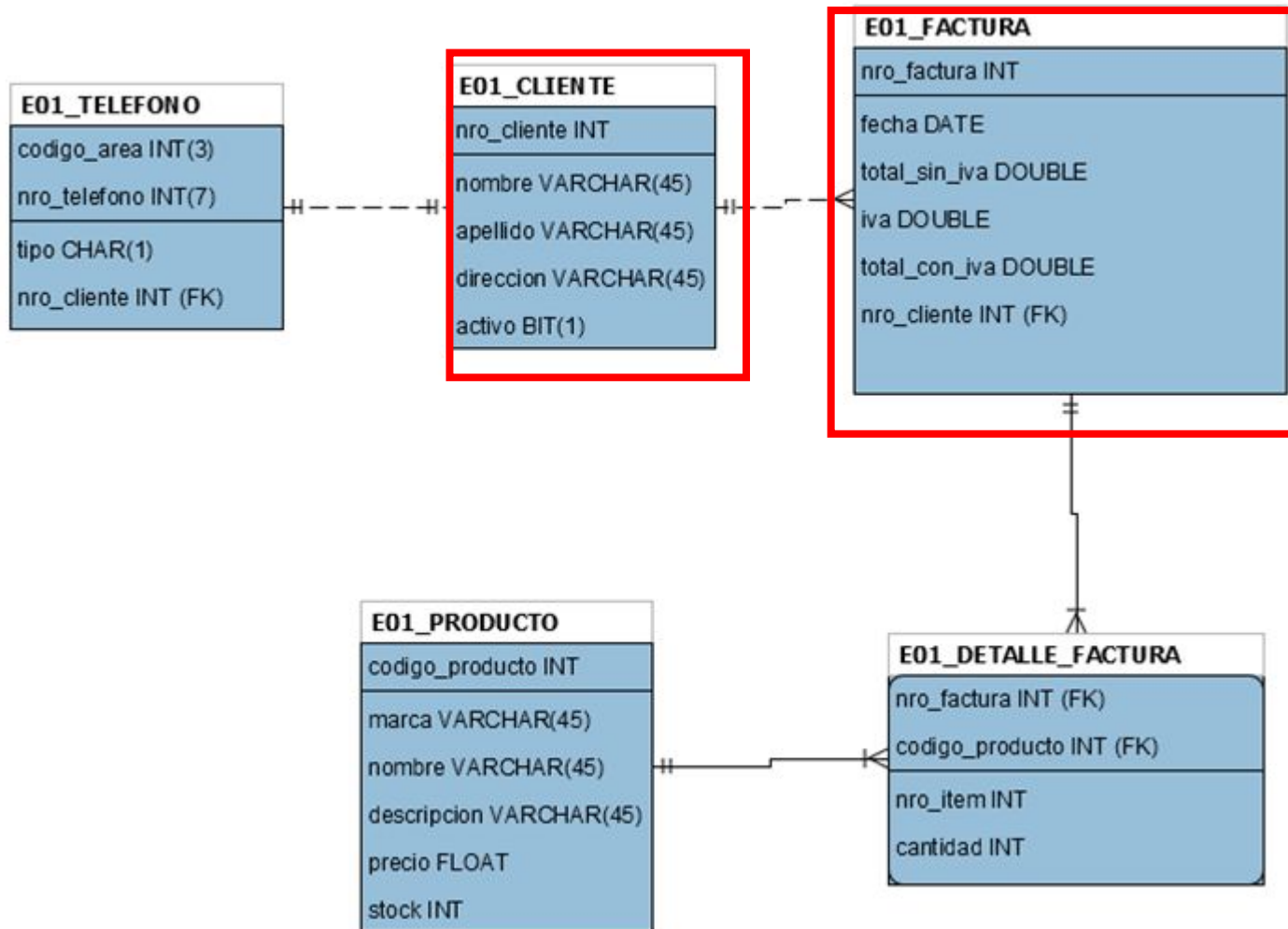
# LEFT JOIN

## *Ejemplo*

- Devolver todos los clientes, con la cantidad de facturas que tienen registradas.

```
SELECT c.*, COUNT(f.nro_factura)
FROM e01_cliente c LEFT JOIN
     e01_factura f
  ON (c.nro_cliente = f.nro_cliente)
GROUP BY c.nro_cliente;
```

# Esquema



# LEFT JOIN

## *Ejemplo*

```
SELECT c.*, COUNT(f.nro_factura)
FROM e01_cliente c LEFT JOIN
     e01_factura f
ON (c.nro_cliente = f.nro_cliente)
GROUP BY c.nro_cliente;
```

Nro_cliente	nombre	apellido	...	count( )
1	Xerxes	Hale	...	4
2	Brent	Leblanc	...	2
...		...	...	...
58	Kaye	Stokes	...	0

# Ejemplo Left Join

```
SELECT account.*,  
       customer.name,  
       customer.lastname,  
       customer.gender,  
       customer.marital_status  
FROM account  
LEFT JOIN customer  
ON account.customer_id=customer.customer_id;
```

# Ejemplo Left Join

Después de ejecutar este código, SQL devuelve lo siguiente:

account_id	overdraft_amount	customer_id	type_id	segment	name
2556889	12000	4	2	RET	DARIO
1323598795	1550	1	1	RET	MARC
2225546	5000	5	2	RET	ELENA
5516229	6000	4	5	RET	DARIO
5356222	7500	5	5	RET	ELENA
2221889	5400	1	2	RET	MARC

lastname	gender	marital_status
PENTAL	M	N
TESCO	M	Y
SIMSON	F	N
PENTAL	M	N
SIMSON	F	N
TESCO	M	Y



# RIGHT JOIN

- Devuelve todas las filas de la tabla derecha y las filas coincidentes de la tabla izquierda.
  - Si existe alguna fila de la tabla derecha que no tiene coincidencia con ninguna fila de la tabla izquierda, dicha fila es mostrada en el resultado, pero aparecerá el valor NULL en las columnas de la tabla izquierda ya que no existe coincidencia.

# Ejemplo Right Join

```
SELECT account.account_id,  
       account.overdraft_amount,  
       account.type_id,  
       account.segment,  
       account.customer_id,  
       customer.customer_id  
       customer.name,  
       customer.lastname,  
       customer.gender,  
       customer.marital_status  
FROM account  
RIGHT JOIN customer  
ON account.customer_id=customer.customer_id;
```

# Ejemplo Right Join

account_id	overdraft_amount	type_id	segment	customer_id	customer_id
1323598795	1550	1	RET	1	1
2221889	5400	2	RET	1	1
NULL	NULL	NULL	NULL	NULL	2
NULL	NULL	NULL	NULL	NULL	3
2556889	12000	2	RET	4	4
5516229	6000	5	RET	4	4
2225546	5000	2	RET	5	5
5356222	7500	5	RET	5	5
NULL	NULL	NULL	NULL	NULL	6
NULL	NULL	NULL	NULL	NULL	7
NULL	NULL	NULL	NULL	NULL	8

name	lastname	gender	marital_status
MARC	TESCO	M	Y
MARC	TESCO	M	Y
ANNA	MARTIN	F	N
EMMA	JOHNSON	F	Y
DARIO	PENTAL	M	N
DARIO	PENTAL	M	N
ELENA	SIMSON	F	N
ELENA	SIMSON	F	N
TIM	ROBITH	M	N
MILA	MORRIS	F	N
JENNY	DWARTH	F	Y

# Bases de Datos

## Carrera Programador full-stack

*Consultas Multi-Tabla*

# Consultas Multi-Tabla

1. Mostrar cada teléfono junto con los datos del cliente.
2. Mostrar todos los teléfonos del cliente número 30 junto con todos sus datos personales.
3. Mostrar nombre y apellido de cada cliente junto con lo que gastó en total (con iva incluido).