

# LENGUAJE DE CONSULTAS A UNA BASE DE DATOS

# UNIDAD 5

**Agrupaciones y consultas**

# COMPETENCIAS

- Realizar agrupaciones en consultas
- Realizar consultas a una base de datos.

# CONSULTAS DE SELECCIÓN CON FUNCIONES DE AGRUPACIÓN

Función	Significado
<b>COUNT</b> (expresion)	Cuenta los elementos de un grupo. Se suele indicar un asterisco ( <b>COUNT(*)</b> ) en lugar de una expresión, ya que la cuenta no varía por indicar una expresión concreta; el resultado siempre es el número de elementos del grupo. Hay que tener en cuenta que esta función ignora los valores nulos a la hora de contar, por lo que la expresión <i>COUNT(telefono)</i> cuenta la cantidad de teléfonos que hay (ignorando los nulos).
<b>SUM</b> (expresión)	Suma los valores de la expresión
<b>AVG</b> (expresión)	Calcula la media aritmética sobre la expresión indicada
<b>MIN</b> (expresión)	Mínimo valor que toma la expresión indicada
<b>MAX</b> (expresión)	Máximo valor que toma la expresión indicada
<b>STDDEV</b> (expresión)	Calcula la desviación estándar
<b>VARIANCE</b> (expresión)	Calcula la varianza

# USO DE DISTINCT EN LAS FUNCIONES DE TOTALES

```
SELECT COUNT(DISTINCT salario);
```

# AGRUPACIONES

## GROUP BY

```
SELECT tipo,modelo  
FROM existencias  
GROUP BY tipo,modelo;
```

```
SELECT tipo,modelo, cantidad,  
SUM(Cantidad)  
FROM existencias  
GROUP BY tipo,modelo;
```

# HAVING

```
SELECT tipo, modelo, cantidad,  
SUM(Cantidad)  
FROM existencias  
GROUP BY tipo, modelo  
HAVING SUM(Cantidad) > 500;
```

# UNIDAD 6

**Modelos de datos y relaciones**



# COMPETENCIAS

- Reconocer distintos modelos de datos.
- Conocer el modelo entidad relación.
- Describir y utilizar relaciones entre tablas.
- Asociaciones y querys anidadas.

# MODELO ENTIDAD RELACIÓN



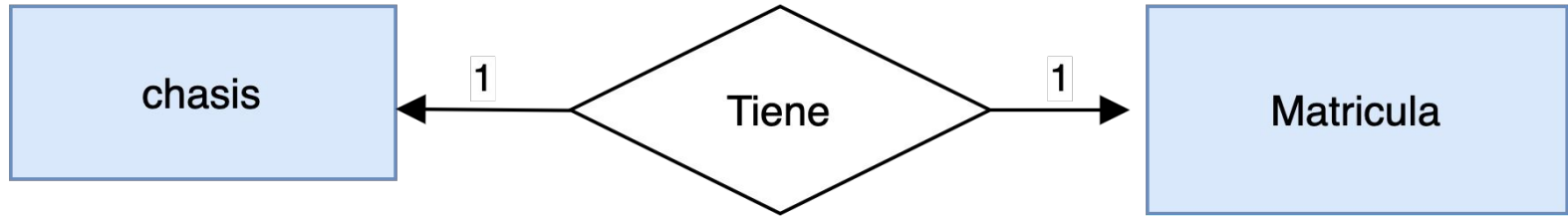
**Tabla empleado**

Nombre	Identificación	Cargo
Luis	115544223	001
Ana	223344556	002
Ignacio	998877665	001

**Tabla Cargo de empleado**

Cargo	Descripción
001	Mecánico
002	Gerente

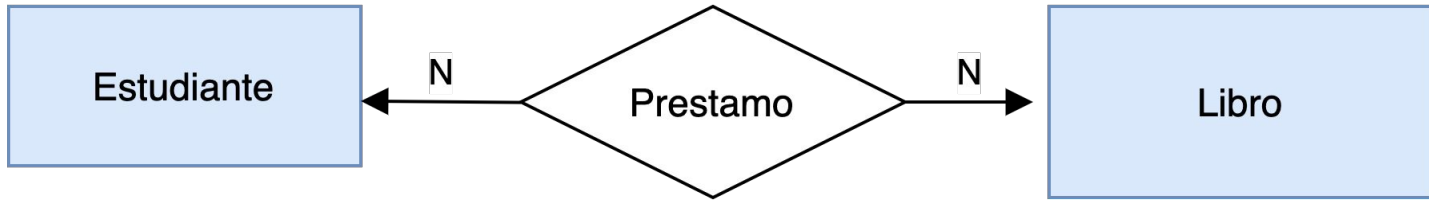
# RELACIÓN UNO A UNO:



# RELACIÓN UNO A VARIOS:



# RELACIÓN VARIOS A VARIOS



# ASOCIACIONES SIMPLES

Para crear una tabla dentro de nuestro motor, debemos utilizar el comando SQL  
CREATE TABLE

```
SELECT nombre, apellido,  
departamento  
FROM departamentos,empleados  
WHERE  
departamentos.cod_dep=empleados.cod_dep  
ORDER BY  
nombre,apellido,departamento;
```

# ASOCIAR MÁS DE UNA TABLA

**Select**

**Insert**

**Update**

**Delete**

```
SELECT nombre,apellido1,apellido2,  
       c.n_curso,  
       fecha_inicio,fecha_fin  
FROM alumnos a, asistir s, cursos  
c  
WHERE a.dni=s.dni AND  
c.n_curso=s.n_curso;
```

# QUERY ANIDADAS

```
select nombre
  from editoriales
 where codigo in
    (select codigoeditorial
      from libros
      where autor='Richard Bach');
```

```
select nombre
  from editoriales
 where codigo not in
    (select codigoeditorial
      from libros
      where autor='Richard Bach');
```



# QUERYS CON DISTINTOS TIPOS DE JOIN

INNER JOIN de Oracle (o, a veces, llamado unión simple)

- Oracle LEFT OUTER JOIN (o algunas veces llamado LEFT JOIN)
- Oracle RIGHT OUTER JOIN (o algunas veces llamado RIGHT JOIN)
- Oracle FULL OUTER JOIN (o algunas veces llamado FULL JOIN)

# INNER JOIN

```
SELECT columns  
FROM table1  
INNER JOIN table2  
ON table1.column = table2.column;
```

# LEFT OUTER JOIN

```
SELECT columns  
FROM table1  
LEFT [OUTER] JOIN table2  
ON table1.column = table2.column;
```

```
SELECT suppliers.supplier_id,  
suppliers.supplier_name,  
orders.order_date  
FROM suppliers  
LEFT OUTER JOIN orders  
ON suppliers.supplier_id =  
orders.supplier_id;
```

# RIGHT OUTER JOIN

```
SELECT columns  
FROM table1  
RIGHT [OUTER] JOIN table2  
ON table1.column = table2.column;
```

```
SELECT orders.order_id,  
orders.order_date,  
suppliers.supplier_name  
FROM suppliers  
RIGHT OUTER JOIN orders  
ON suppliers.supplier_id =  
orders.supplier_id;
```

# ACTIVIDADES PROPUESTAS

Mucho  
éxito y no  
olvides  
consultar  
las dudas.



Foto de [Tom Swinnen](#)  
en [Pexels](#)