

Fundamentos Tecnológicos para el Tratamiento de Datos

Profesor: Serhiy Lyalkov

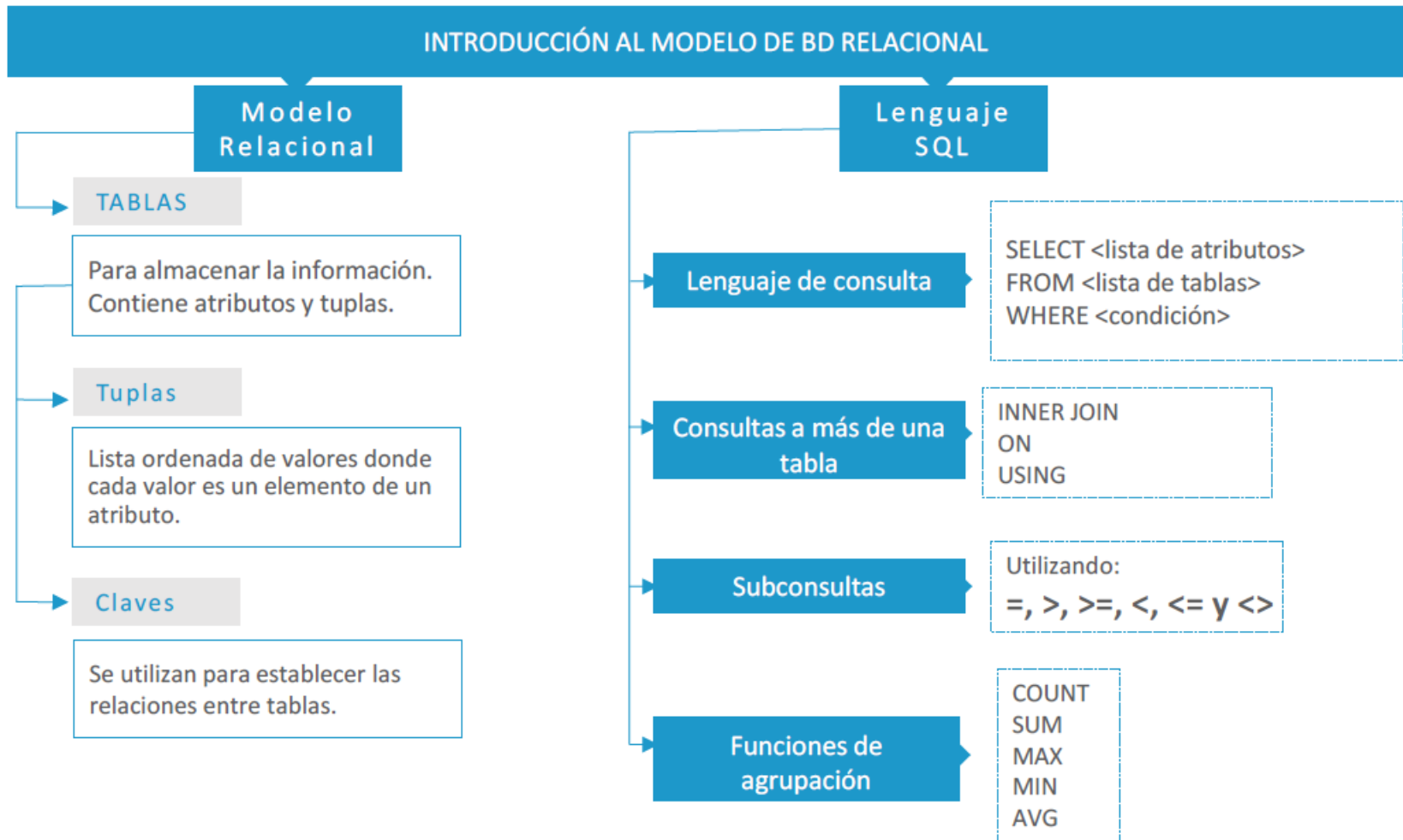
Tema 3 Sesión 5 y 6

Bases de datos relacionales

Índice de la sesión

- Esquema
- 3.1 Introducción y objetivos
- 3.2 Introducción al modelo relacional
- 3.3 Estructura del modelo relacional
- 3.4 Lenguaje SQL
- 3.5 Consultas SQL
- Presentación actividad: Consultas SQL

Esquema



3.1 Introducción y objetivos

3.1 Introducción y objetivos

Objetivos

- Modelo Relacional, características y uso
- Estructura del modelo relacional
- Almacenamiento y gestión de datos en modelo relacional
- Terminología del lenguaje SQL
- Consultar bases de datos mediante consultas SQL

3.2 Introducción al modelo relacional

3.2 Introducción al modelo relacional

Modelo relacional



- Más utilizado por SGBD y bases de datos
- Objetivo: independencia de la estructura lógica de las bases de datos
- Teoría de relacionales (tablas)
- Interconexión para la representación de la realidad

ELEMENTOS DE LA TABLA

- Nombres ÚNICOS de tablas
- Columnas/Atributos
- Dominio/tipo de valores posibles
- Filas/Tuplas
- Cardinalidad = número de filas/tuplas
- Grado = número de columnas/atributos

```
SELECT * FROM dbo.Alumno
```

	nAl	DNI	NombreAlumno	fechaNac	lugar	nH	computador
1	88	29214856	Teresa Diaz Camacho	1971-10-25	Madrid	4	Ord010
2	112	44125325	Francisco Gallego Macias	1975-01-01	Cadiz	2	Ord009
3	145	29254123	Manuel Lopez Gonzalez	1975-03-15	Huelva	0	Ord009
4	210	29555247	Beatriz Rico Vazquez	1974-11-12	Cadiz	2	Ord025
5	214	29147158	Mercedes Gomez Martin	1976-06-20	Huelva	1	Ord010

3.3 Estructura del modelo relacional

3.3 Estructura del modelo relacional

Elementos

```
SELECT * FROM dbo.Alumno
```

← Nombre de la tabla

	nAI	DNI	NombreAlumno	fechaNac	lugar	nH	computador
1	88	29214856	Teresa Diaz Camacho	1971-10-25	Madrid	4	Ord010
2	112	44125325	Francisco Gallego Macias	1975-01-01	Cadiz	2	Ord009
3	145	29254123	Manuel Lopez Gonzalez	1975-03-15	Huelva	0	Ord009
4	210	29555247	Beatriz Rico Vazquez	1974-11-12	Cadiz	2	Ord025
5	214	29147158	Mercedes Gomez Martin	1976-06-20	Huelva	1	Ord010

Tuplas:
Lista ordenada de valores siendo un elemento de un atributo y formando parte de un dominio, o valor nulo "NULL"

Claves:
Cjto no vacío de atributos que identifica unívocamente cada tabla

Dominio: Nombre + Tipo de Datos

- Intensión: especificando el tipo
- Extensión: valores posibles

- 1. Candidatas
- 2. Clave Primaria o Alternativa

- Restricciones:**
- No tuplas repetidas
 - Sin orden establecido
 - No 2 atributos con mismo nombre
 - Orden de atributos no relevante
 - Valores atómicos en atributos
 - Tuplas con valores desconocidos "nulos"

3.3 Estructura del modelo relacional

Elementos

```
SELECT * FROM dbo.Alumno
```

	nAl	DNI	NombreAlumno	fechaNac	lugar	nH	computador
1	88	29214856	Teresa Diaz Camacho	1971-10-25	Madrid	4	Ord010
2	112	44125325	Francisco Gallego Macias	1975-01-01	Cadiz	2	Ord009
3	145	29254123	Manuel Lopez Gonzalez	1975-03-15	Huelva	0	Ord009
4	210	29555247	Beatriz Rico Vazquez	1974-11-12	Cadiz	2	Ord025
5	214	29147158	Mercedes Gomez Martin	1976-06-20	Huelva	1	Ord010

↑ CP

↑ CA

```
SELECT * FROM dbo.Computador
```

	Ordenador	Nombre	lugar
1	Ord025	Servidor NT	Control
2	Ord008	PC Practicas	Aula 7
3	Ord009	PC Practicas	Aula 7
4	Ord010	PC Practicas	Aula 8
5	Ord040	Beatriz Rico Vazquez	Aula 10

↑ CP

```
SELECT * FROM dbo.Alumno AS al
INNER JOIN dbo.Computador AS cm
ON cm.Ordenador = al.computador
```

	nAl	DNI	NombreAlumno	fechaNac	lugar	nH	computador	nAlID	nAl	Nombre	lugar
1	88	29214856	Teresa Diaz Camacho	1971-10-25	Madrid	4	Ord010	10	Ord010	PC Practicas	Aula 8
2	112	44125325	Francisco Gallego Macias	1975-01-01	Cadiz	2	Ord009	9	Ord009	PC Practicas	Aula 7
3	145	29254123	Manuel Lopez Gonzalez	1975-03-15	Huelva	0	Ord009	9	Ord009	PC Practicas	Aula 7
4	210	29555247	Beatriz Rico Vazquez	1974-11-12	Cadiz	2	Ord025	25	Ord025	Servidor NT	Control
5	214	29147158	Mercedes Gomez Martin	1976-06-20	Huelva	1	Ord010	10	Ord010	PC Practicas	Aula 8

3.3 Estructura del modelo relacional

Restricciones adicionales

No permite
valores **nulos**
en ciertos
atributos

Obligatoriedad

Unicidad

No permite
valores
repetidos en
ciertos
atributos

Integridad
Semántica

Introducción
solo de valores
deseados

3.4 Lenguaje SQL

3.4 Lenguaje SQL

SQL (Structured Query Language)



- Definir
 - Manipular
 - Consultar
- Los datos en las bases de datos relacionales

Lenguaje de definición de datos (DDL)

- Crear estructura y vistas externas de la base de datos
- Se describen y nombran las entidades, atributos y relaciones
- Se definen las restricciones de integridad y seguridad

```
CREATE TABLE Alumno(  
  nAI INT PRIMARY KEY,  
  DNI INT,  
  NombreAlumno VARCHAR(100),  
  fechaNac DATE,  
  lugar VARCHAR(100),  
  nH INT,  
  computador VARCHAR(100))
```

```
DELETE FROM Alumno  
WHERE DNI = 29555247
```

```
UPDATE Alumno  
SET fechaNac = '20020127'  
WHERE nAI = 112
```

```
INSERT INTO Computador (nAIID,nAI, Nombre, lugar)  
VALUES (25 , 'Ord025', 'Servidor NT', 'Control'),  
(8 , 'Ord008' , 'PC Practicas', 'Aula 7'),
```

3.4 Lenguaje SQL

SQL (Structured Query Language)

Lenguaje de Consulta

Nombre de atributo
o atributos
(Todos los atributos
de una tabla: *)

```
SELECT <Atributos>  
FROM <Tabla> AS tab  
WHERE <Condición>
```

Nombre de la tabla que
contiene los atributos
- 2 o más tablas: **JOIN**
- *Alias* de la tabla

Condiciones específicas que deben
cumplir los atributos de la tabla

Conocer código, nombre y especialidad de asignaturas de 3 curso con mas de 4.5 créditos

```
Atributos → SELECT A.idAsig AS CodigoAsignatura, A.Nombre, A.Especialidad  
FROM Asignatura as A ← Nombre tabla y su alias  
WHERE A.Curso = 3 and A.Creditos > 4.5  
          ↑           ↑  
          Condiciones
```

3.5 Consultas SQL

3.5 Consultas SQL

SQL (Structured Query Language)

Peculiaridades:

- Alias para las tablas eliminando ambigüedad
- Renombrar los nombres de los atributos al extraerlos
- **WHERE** es opcional
- Todos los valores: caracter *
- Tuplas únicas: clausula **DISTINCT**

```
FROM Asignatura as A
```

```
SELECT A.idAsig AS CodigoAsignatura
```

```
SELECT *  
FROM Asignatura AS asi
```

```
SELECT DISTINCT Nombre  
FROM Computador
```


3.5 Consultas SQL

SQL (Structured Query Language)

Operaciones aritméticas:

- ORDER BY ... DESC ASC
- BETWEEN
- IN

```
SELECT nH * 300 as Descuento
FROM Alumno
ORDER BY Descuento DESC, NombreAlumno ASC
```

```
SELECT DNI
FROM Alumno
WHERE fechaNac BETWEEN '19900101' and '20001231'
```

```
SELECT *
FROM Alumno
WHERE Lugar IN ('Huelva', 'Cadiz')
```

```
SELECT DNI
FROM Alumno
WHERE NombreAlumno LIKE 'A%' AND lugar
LIKE '_____%'
AND lugar NOT LIKE 'M%'
```

Operaciones aritméticas:

- LIKE
- NOT LIKE

3.5 Consultas SQL

SQL (Structured Query Language)

Consultas a 2 o más tablas:

- INNER JOIN <tabla> ON <CP> = <CA>
- USING

OJO: debemos utilizar *alias*

```
SELECT asi.Nombre, pr.Nombre  
FROM Asignatura AS asi  
INNER JOIN Profesor AS pr ON pr.ProfID = asi.ProfID
```

```
SELECT ma.nAI  
FROM Matricula AS ma  
INNER JOIN Asignatura AS asi ON asi.idAsig = ma.idAsig  
WHERE asi.Nombre = 'Fundamentos Tecnológicos para el Tratamiento de  
Datos' AND ma.An=2019
```

3.5 Consultas SQL

SQL (Structured Query Language)

Subconsultas:

```
SELECT al.NombreAlumno
FROM Alumno AS al
WHERE lugar = (SELECT Lugar FROM Alumno WHERE NombreAlumno =
'Francisco Gallego Macias')
AND nH = (SELECT nH FROM Alumno WHERE NombreAlumno = 'Beatriz Rico
Vazquez')
```

3.5 Consultas SQL

SQL (Structured Query Language)

Funciones de agrupación:

- COUNT
- SUM
- MAX
- MIN
- AVG

```
SELECT al.NombreAlumno, avg(ma.Calificacion)
FROM Alumno AS al INNER JOIN Matricula AS ma ON ma.nAI = al.nAI
WHERE ma.An = 2019 AND ma.Convocatoria='Junio'
GROUP BY al.NombreAlumno
HAVING avg(ma.Calificacion) > 5
```

```
SELECT COUNT(*) AS NumMatri, MIN(Calificacion) AS MINIMA, AVG(Calificacion) AS Media
FROM Asignatura AS asi INNER JOIN Matricula AS ma ON ma.idAsig = asi.idAsig
WHERE asi.Nombre = 'Fundamentos Tecnológicos para el Tratamiento de Datos' and
ma.An=2019
```

```
SELECT ma.Convocatoria, asi.Nombre, count(*) as Presentados, AVG(Calificacion) AS
Media
FROM Asignatura AS asi INNER JOIN Matricula AS ma ON asi.idAsig = ma.idAsig
GROUP BY ma.Convocatoria, asi.Nombre
```

OJO: en el caso de que queramos agrupar por atributo, debemos usar
GROUP BY



www.unir.net