



MAPEO FÍSICO DE UNA DB RELACIONAL



MAPEO FÍSICO

Nos interesa el 'Cómo' trasladar nuestro modelo E/R y nuestro diagrama de tablas a un DBMS en particular. No es una actividad aislada, las decisiones tomadas durante el Mapeo Físico para mejorar el funcionamiento podría afectar a la estructura del modelo lógico de E/R.



STRUCTURED QUERY LANGUAGE (SQL)

Creado en 1974 da acceso a un RDBMS permitiendo ejecutar diferentes tipos de operaciones dentro del sistema. Basado en el álgebra y el cálculo relacional hace posible consultar información y realizar cambios de la misma.



STRUCTURED QUERY LANGUAGE (SQL)

- Lenguaje de Definición de Datos (DDL)
- Lenguaje de Manipulación de Datos (DML)
- Lenguaje de Control de Datos (DCL)



DDL

CREATE

- CREATE TABLE 'EMPLEADO'
- CREATE DATABASE 'VENTAS'

Antes de crear una tabla o cualquier objeto, se debe especificar el esquema en el cual se trabaja.

USE 'VENTAS'



DDL

ALTER

- ALTER TABLE 'ALUMNO' ADD Edad INT UNSIGNED;



DDL

DROP

- DROP TABLE 'ALUMNO'
- DROP DATABASE 'VENTAS'



DDL

TRUNCATE

- TRUNCATE TABLE 'NOMBRE_TABLA';



DDL

TIPOS DE DATOS NUMÉRICOS

- INTEGER
- DECIMAL (FLOAT, DOUBLE)
- BOOLEAN (1/0)



DDL

TIPOS DE DATOS DE TEXTO

- CHAR(N)
- VARCHAR(N)



DDL

TIPOS DE DATOS DE FECHAS

- DATE
- TIME
- DATETIME
- TIMESTAMP



DDL

TIPOS DE DATOS GRANDES

- CLOB (Char Long OBject)
- BLOB (Binary Long OBject)



DDL

TIPOS DE DATOS

- Todos los tipos aceptan valor NULL
- El tipo de datos debe permitir almacenar todos los valores previstos, pero usar un tipo demasiado grande puede desperdiciar espacio y limitar el desempeño.



DDL

RESTRICCIONES (CONSTRAINTS)

Permiten implementar las restricciones del modelo, como llaves primarias, foráneas, NOT NULL, etc.



DDL

CONSTRAINTS DE LLAVE PRIMARIA

- CREATE TABLE ALUMNO (

Carnet INT NOT NULL,

CONSTRAINT PK_ALUMNO PRIMARY KEY (Carnet));
- ALTER TABLE ALUMNO ADD CONSTRAINT
PK_ALUMNO PRIMARY KEY (Carnet) ;



DDL

CONSTRAINTS DE LLAVE FORANEA

- CREATE TABLE ALUMNO (Carnet INT NOT NULL,
CONSTRAINT FK_ALUMNO FOREIGN KEY (Direccion)
REFERENCES DIRECCION(Codigo));
- ALTER TABLE ALUMNO ADD CONSTRAINT
FK_ALUMNO FOREIGN KEY (Direccion) REFERENCES
DIRECCION(Codigo)



DDL

Restricciones de integridad referencial en cascada

- On Delete
- On Update



STRUCTURED QUERY LANGUAGE (SQL)

Lenguaje de Manipulación de datos (DML):

SELECT

```
SELECT [ALL | DISTINCT ]
        <nombre campo> [{,<nombre_campo>}]
FROM <nombre tabla>|<nombre vista>
        [{,<nombre tabla>|<nombre vista>}]
[WHERE <condición> [{ AND|OR <condición>}]]
[GROUP BY <nombre_campo> [{,<nombre_campo >}]]
[HAVING <condición>[{ AND|OR <condición>}]]
[ORDER BY <nombre campo>|<indice campo> [ASC | DESC]
        [{,<nombre_campo>|<indice_campo> [ASC | DESC ]}]]
```



STRUCTURED QUERY LANGUAGE (SQL)

Lenguaje de Manipulación de datos (DML):

SELECT

- `SELECT * FROM ALUMNO`
- `SELECT Carnet, Nombre FROM ALUMNO
WHERE Edad > 20 ORDER BY Edad DESC;`



STRUCTURED QUERY LANGUAGE (SQL)

Lenguaje de Manipulación de datos (DML):

INSERT

- INSERT INTO CARRO VALUES ('valor1','valor2')
- INSERT INTO CARRO ('Modelo', 'Placa') values ('2000', 'P-325BKA')



STRUCTURED QUERY LANGUAGE (SQL)

Lenguaje de Manipulación de datos (DML):

UPDATE

- UPDATE CARRO SET Modelo = '2010'
- UPDATE CARRO SET Modelo = '2010' WHERE Placa = "P-325BKA"



STRUCTURED QUERY LANGUAGE (SQL)

Lenguaje de Manipulación de datos (DML):

DELETE

- `DELETE FROM ALUMNO //`Cuidado con esto
- `DELETE FROM ALUMNO WHERE Edad > 25`