

Organización y manejo de archivos

www.inf.ucv.cl/ [wpalma/oma](mailto:wpalma@oma)

Dr. Wenceslao Palma
wenceslao.palma@ucv.cl



- Formalmente introducido por Dr. E. F. Codd en el año 1970.
- Representa datos en forma de tablas bidimensionales.
- Una BD Relacional es una colección de tablas relacionales.

campo, atributo, columna

↓

registro, tupla, fila. →

Nombre	Dirección	Edad
Bon Scott	15 Norte 321	35
Cliff Burton	San José 345, Agua Santa	
Chuck Schuldiner	Pedro Montt, 341	30

Propiedades de una tabla

- Sus valores son atómicos.
- Los valores de cada columna son de la misma clase.
- Cada registro es único.
- El orden de las columnas es irrelevante.
- El orden de los registros es irrelevante.
- Cada columna tiene un nombre único.

Relaciones y claves

- Una relación es una asociación entre tablas, la cual se expresa mediante los valores de la clave primaria y la clave foránea.
- Es posible ver una clave foránea como la copia de una clave primaria desde otra tabla.
- Por lo anterior una relación se materializa mediante el calce de los valores de una clave foránea de una tabla con los valores de una clave primaria en otra.

Integridad de los datos

- Significa que los datos almacenado en la BD son correctos y consistentes.
- Existen 2 reglas básicas para asegurar la integridad de los datos: integridad de la entidad y integridad referencial.
- Integridad de la entidad: el valor de la clave primaria nunca puede ser nulo.
- Integridad referencial: indica que si una tabla posee una clave foránea, entonces cada valor debe tener un calce en la tabla en donde dicha clave es primaria.

Postgresql

- El primer ancestro fue Ingres (1977-1985) desarrollado en la Universidad de California, Berkeley (UCB). Luego aparecieron, Postgres (1986-1994) UCB, Postgres95 (1994-1995) y Postgresql (1996-)
- Postgresql es open source.
- Para conectarse a una BD postgresql se puede utilizar psql

```
$psql -U wpalma
```

```
Password:
```

```
Welcome to psql 7.4.1, the PostgreSQL interactive terminal.
```

```
Type: \copyright for distribution terms
```

```
      \h for help with SQL commands
```

```
      \? for help on internal slash commands
```

```
      \g or terminate with semicolon to execute query
```

```
      \q to quit
```

```
wpalma=>
```

Crear una tabla

```
wpalma=> create table persona (  
wpalma(>           nombre char(15),  
wpalma(>           apellido char(20),  
wpalma(>           ciudad  char(15),  
wpalma(>           edad    integer  
wpalma(> );  
CREATE TABLE
```

Eliminar una tabla

```
wpalma=> drop table persona;
```

Mostrar la estructura de una tabla

```
wpalma=> \d persona;
```

```
Table "public.persona"
```

Column	Type	Modifiers
nombre	character varying(15)	
apellido	character(20)	
ciudad	character(15)	
edad	integer	

Ingreso de datos mediante insert.

Es posible escribir un archivo de texto con las sentencias de inserción. Por ejemplo: insert_script.psql

```
insert into persona values('Bon','Scott','Vina del mar',35);
insert into persona values('Cliff','Burton','Vina del Mar',25);
insert into persona values('Chuck','Schuldiner','Valparaiso',30);
```

Luego desde psql:

```
wpalma=> \i insert_script.psql
INSERT 37162 1
INSERT 37163 1
INSERT 37164 1
```

Para verificar el ingreso:

```
wpalma=> select * from persona;
```

nombre	apellido	ciudad	edad
Bon	Scott	Vina del mar	35
Cliff	Burton	Vina del Mar	25
Chuck	Schuldiner	Valparaiso	30

(3 rows)

Eliminación de registros

```
wpalma=> delete from persona where ciudad='Valparaiso';  
DELETE 1
```

Modificación de atributos de un registro

```
wpalma=> select * from persona;
```

nombre	apellido	ciudad	edad
Bon	Scott	Vina del mar	35
Cliff	Burton	Vina del Mar	25
Chuck	Schuldiner	Valparaiso	30

(3 rows)

```
wpalma=> update persona set ciudad='Quillota' where edad<=30;  
UPDATE 2
```

Verificando el resultado

```
wpalma=> select nombre,ciudad from persona;
```

```
  nombre |      ciudad
-----+-----
  Bon    | Vina del mar
  Cliff  | Quillota
  Chuck  | Quillota
(3 rows)
```

Mostrando registros en forma ordenada

```
wpalma=> select * from persona ORDER BY edad;
```

nombre	apellido	ciudad	edad
Cliff	Burton	Quillota	25
Chuck	Schuldiner	Quillota	30
Bon	Scott	Vina del mar	35

(3 rows)

```
wpalma=> select * from persona ORDER BY edad DESC;
```

nombre	apellido	ciudad	edad
Bon	Scott	Vina del mar	35
Chuck	Schuldiner	Quillota	30
Cliff	Burton	Quillota	25

(3 rows)

Tipos de datos

Tipo

char(largo)

varchar(largo)

integer

float

numeric(precisión,decimal)

date

time

timestamp

Descripción

arreglo tamaño fijo

arreglo largo variable

entero, rango +/- 2 billones

número punto flotante, 15 dígitos precisión

precisión definida por el usuario

fecha

hora

marca de fecha y hora

Uso de valores por defecto

```
wpalma=> create table cuenta(  
wpalma(>         nombre varchar(35),  
wpalma(>         balance numeric(16,2) DEFAULT 0,  
wpalma(>         active char(1) DEFAULT 'Y',  
wpalma(>         apertura timestamp DEFAULT CURRENT_TIMESTAMP  
wpalma(> );
```

CREATE TABLE

```
wpalma=> insert into cuenta (nombre) values('juan perez');  
INSERT 37284 1
```

```
wpalma=> select * from cuenta;
```

nombre	balance	active	apertura
juan perez	0.00	Y	2004-05-31 12:00:51.625252

(1 row)

Consultando registros

```
wpalma=> select * from persona where ciudad='Quillota' AND edad <=30;
  nombre |      apellido      |      ciudad      | edad
-----+-----+-----+-----
Cliff   | Burton             | Quillota         | 25
Chuck   | Schuldiner         | Quillota         | 30
(2 rows)
```

```
wpalma=> select * from persona where
(ciudad='Quillota' OR ciudad='Vina del mar') AND edad >=30;
  nombre |      apellido      |      ciudad      | edad
-----+-----+-----+-----
Bon      | Scott              | Vina del mar     | 35
Chuck    | Schuldiner         | Quillota         | 30
(2 rows)
```


Rangos

```
wpalma=> select * from persona where edad BETWEEN 25 AND 30;
```

nombre	apellido	ciudad	edad
Cliff	Burton	Quillota	25
Chuck	Schuldiner	Quillota	30

(2 rows)

```
wpalma=> select * from persona where edad >= 25 AND edad <= 30;
```

Comparaciones con like

```
wpalma=> select * from persona where ciudad like '%V%';
```

nombre	apellido	ciudad	edad
Bon	Scott	Vina del mar	35

(1 row)

Comparación

Comienza con D

Contiene D

D en segundo lugar

No comienza con D

Sintaxis

like 'D%'

like '%D%'

like '_D%'

not like 'D%'

Select distinct

```
wpalma=> select * from persona order by ciudad;
```

nombre	apellido	ciudad	edad
Cliff	Burton	Quillota	25
Chuck	Schuldiner	Quillota	30
Bon	Scott	Vina del mar	35

(3 rows)

```
wpalma=> select distinct ciudad from persona order by ciudad;  
ciudad
```

```
-----  
Quillota  
Vina del mar  
(2 rows)
```

Select distinct

```
wpalma=>select distinct ciudad,nombre,apellido  
        from persona order by ciudad;
```

ciudad	nombre	apellido
Quillota	Chuck	Schuldiner
Quillota	Cliff	Burton
Vina del mar	Bon	Scott

(3 rows)

Constraints

- Muy útiles para evitar datos inválidos en la BD.
- **NOT NULL**: evita que se inserten valores nulos en una columna. Se puede combinar con **DEFAULT**.
- **UNIQUE**: evita valores duplicados en una columna. Se implementa creando un índice. Es posible ingresar valores nulos.
- **PRIMARY KEY**: indica la columna que identifica de manera única a cada registro. Es una combinación de **UNIQUE** y **NOT NULL**. Al igual que con **UNIQUE** se crea automáticamente un índice y se agrega el modificador **NOT NULL** a la columna.
- **REFERENCES**: se utiliza para crear un clave foránea.

Ejemplo

```
wpalma=> create table asignatura(clave char(8) PRIMARY KEY,  
wpalma(> nombre varchar(15),  
wpalma(> credito integer);
```

```
NOTICE: CREATE TABLE / PRIMARY KEY
```

```
#creara el indice implicito "asignatura_pkey" para la tabla "asignatura"  
CREATE TABLE
```

```
wpalma=> create table alumno (rol char(8) PRIMARY KEY,  
wpalma(> nombre varchar(14),  
wpalma(> apellido varchar(12));
```

```
NOTICE: CREATE TABLE / PRIMARY KEY
```

```
#creara el indice implicito "alumno_pkey" para la tabla "alumno"  
CREATE TABLE
```

```
wpalma=> create table inscripcion(clave char(8) REFERENCES asignatura,  
wpalma(> rol char(8) REFERENCES alumno,  
wpalma(> periodo DATE);  
CREATE TABLE
```