Curso de



Bagde 

del curso

**Fundamentos de Bases de Datos**

Israel Vázquez

Introducción

**Introducción y bienvenida**

**Israel Vázquez**

Senior Web Developer en SF startup YouNoodle. Organizer del GDG Cloud Mexico. Seminarista de Bases de Datos relacionales. Data engineering enthusiast.

**Historia de la persistencia de la información**

**¿Por qué**

**necesitamos persistencia?**

Espacio para imagen

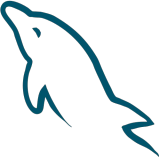
**Entonces,**

**¿Que \*&%)$@ son las Bases de Datos?**

Espacio para imagen

**Tipos de bases de datos** Relacionales No relacionales

**Bases de datos relacionales**

**Ejemplos de bases de datos relacionales**

**Bases de datos no relacionales**

**Bases de datos no relacionales**

****

****

**Bases de datos no relacionales**

****

**Servicios**

Auto

administrados

Administrados

**Introducción a la bases de datos relacionales**

**Historia**

Espacio para imagen

**No me hables de álgebra**Espacio para imagen

**Entidades laptops**

Espacio para imagen

**Atributos**

color

pantalla

año

modelo

no de serie **laptops**

Espacio para imagen

método de

antigüedad disco duro entrada

trackpad teclado

**Atributos**

**no de serie**

LKJ789JKAS

KCO3100KJH

NSDJOIH128

09KSIHBD71

color gris

negro negro gris

año

2017 2019 2018 2017

pantalla AX4829i AX4930i AX4930i AX4829i

**Entidades débiles**

Libros Ejemplares

**Entidades débiles: identidad**

**id**

**Libros**título

...

**libro\_id**

**Ejemplares** localización

edición

LKJ789JKAS KCO3100KJH NSDJOIH128 09KSIHBD71

Viaje al cent... El señor de ... De la tierra... Amor en tie...

... ... ... ...

LKJ789JKAS KCO3100KJH NSDJOIH128 09KSIHBD71

pasillo 1 pasillo 1 pasillo 1 pasillo 1

1 1 3 1

**Entidades débiles: existencia**

**id**

**Libros** título

...

**id**

**Ejemplares** localización

edición

LKJ789JKAS KCO3100KJH NSDJOIH128 09KSIHBD71

Viaje al cent... El señor de ... De la tierra... Amor en tie...

... ... ... ...

JKE7823CLK JKFE1093JD 82938ISHDIK 838439JHDUI

pasillo 1 pasillo 1 pasillo 1 pasillo 1

1 1 3 1

**Diagrama ER: Platziblog** usuarios comentarios

posts

categorías

**Entidades Platziblog**

titulo

id

**posts**

fecha\_publicacion

etiquetascontenido estatus

**Entidades Platziblog**

login

id

**usuarios**

email

password apodo

**Relaciones**

Espacio para imagen

tiene

**Relaciones** laptops

tiene

discos\_duros

**Cardinalidad**

Espacio para imagen

**Cardinalidad: 1 a 1**

**1:1**

**1 1**

persona tiene datos\_contacto **1 1**

**Cardinalidad: 0 a 1**

**0:1**

sesion\_actual tiene usuario

**Cardinalidad: 1 a N**

**1:N**

**1 N**

persona tiene automovil **1 1**

**Cardinalidad: 0 a N**

**0:N**

paciente tiene hab\_hospital

**Cardinalidad: N a N**

**N:N**

**1 N**

alumno pertenece clase **N 1**

**Diagrama ER**

Espacio para imagen

**Diagrama ER: Platziblog**

**1:N**

usuarios comentarios escribe

escribe

**1:N**

tiene

**1:N**

posts

categorias

**1:N**

tiene

**N:N**

tiene etiquetas

**Diagrama Físico**Espacio para imagen

**Tipos de dato**

Texto

CHAR(n)

VARCHAR(n) TEXT

Números

INTEGER

BIGINT

SMALLINT

DECIMAL(n, s) NUMERIC (n, s)

Fecha/hora DATE

TIME

DATETIME TIMESTAMP

Lógicos

BOOLEAN

**Constraints (Restricciones)**

Constraint Descripción

**NOT NULL** Se asegura que la columna no tenga valores nulos **UNIQUE** Se asegura que cada valor en la columna no se repita **PRIMARY KEY** Es una combinación de NOT NULL y UNIQUE **FOREIGN KEY** Identifica de manera única una tupla en otra tabla

**CHECK** Se asegura que el valor en la columna cumpla una condición dada

**DEFAULT** Coloca un valor por defecto cuando no hay un valor especificado

**INDEX** Se crea por columna para permitir búsquedas más rápidas

**Normalización**

Espacio para imagen

**Sin normalizar**

**alumno nivel\_curso nombre\_curso materia\_1 materia\_2** Juanito Maestría Data

engineering MySQL Python

Pepito Licenciatura Programación MySQL Python

Primera forma normal (1FN) Atributos atómicos (Sin campos repetidos)

**alumnos**

**alumno\_id alumno nivel\_curso nombre\_curso materia** 1 Juanito Maestría Data engineering MySQL 1 Juanito Maestría Data engineering Python 2 Pepito Licenciatura Programación MySQL 2 Pepito Licenciatura Programación Python

Segunda forma normal (2FN)

Cumple 1FN y Cada campo de la tabla debe depender de una clave única

**alumnos**

**alumno\_id alumno nivel\_curso nombre\_curso** 1 Juanito Maestría Data engineering 2 Pepito Licenciatura Programación

**materias**

**materia\_id alumno\_id materia**

1 1 MySQL

2 1 Python

3 2 MySQL

4 2 Python

Tercera forma normal (3FN)

Cumple 1FN y 2FN y los campos que NO son clave NO deben tener dependencias.

**alumnos**

**cursos**

| **alumno\_id** |
| --- |

**alumno curso\_id**

**curso\_id nivel\_curso nombre\_curso**

1 Juanito 1 2 Pepito 2

1 Maestría Data engineering 2 Licenciatura Programación

**materias**

**materia\_id alumno\_id materia** 1 1 MySQL 2 1 Python 3 2 MySQL 4 2 Python

Cuarta forma normal (4FN)

Cumple 1FN, 2FN y 3FN los

campos multivaluados se

identifican por una clave única.

| **alumno\_id alumno** |
| --- |

**alumnos**

**curso\_id**

**cursos**

**curso\_id nivel\_curso nombre\_curso**

1 Juanito 1 2 Pepito 2

**materias**

**materia\_id materia** 1 MySQL 2 Python

1 Maestría Data engineering 2 Licenciatura Programación

**materias\_por\_alumno**

**mpa\_id materia\_id alumno\_id** 1 1 1

2 2 1

3 1 2

4 2 2

**Diagrama ER: Platziblog**

**1:N**

usuarios comentarios escribe

escribe

**1:N**

tiene

**1:N**

posts

categorias

**1:N**

tiene

**N:N**

tiene etiquetas

usuarios

id: INTEGER (PK)

login: VARCHAR(30) N~~N~~

password: VARCHAR(32) NN

nickname: VARCHAR(40) NN

email: VARCHAR(40) NN UNIQUE

posts

id: INTEGER (PK)

titulo: VARCHAR (150)

fecha\_publicacion: TIMESTAMP

contenido: TEXT

comentarios id: INTEGER (PK)

comentario: TEXT

usuarios\_id: INTEGER FK posts\_id: INTEGER FK

categorias id: INTEGER (PK)

estatus: CHAR(8) CHECK( IN (’activo’, ‘inactivo’) usuarios\_id: INTEGER FK

categorias\_id: INTEGER FK

etiquetas

id: INTEGER (PK)

categoria: VARCHAR(30)

nombre\_etiqueta VARCHAR(30)

posts

id: INTEGER (PK)

titulo: VARCHAR (15~~0)~~

fecha\_publicacion: TIMESTAMP

contenido: TEXT

estatus: CHAR(8) CHECK( IN (’activo’, ‘inactivo’) usuarios\_id: INTEGER FK

categorias\_id: INTEGER FK

etiquetas

id: INTEGER (PK)

nombre\_etiqueta VARCHAR(30)

posts

id: INTEGER (PK)

titulo: VARCHAR (15~~0)~~

fecha\_publicacion: TIMESTAMP

contenido: TEXT

estatus: CHAR(8) CHECK( IN (’activo’, ‘inactivo’)

usuarios\_id: INTEGER FK

categorias\_id: INTEGER FK

posts\_etiquetas

post\_id: INTEGER (PK, FK)

etiqueta\_id: INTEGER (PK, FK)

etiquetas

id: INTEGER (PK)

nombre\_etiqueta VARCHAR(30)

**RDBMS o cómo hacer lo anterior de manera práctica**

**RDB ¿Qué?**

**RDBMS**

**R**elational **D**atabase

**M**anagement **S**ystem







**Instalación local de un RDBMS (Mac)**

**Descargar el instalador** https://dev.mysql.com/downloads/mysql/5.6.html

**Instalación local de un RDBMS**

**(Windows)**

**Descargar el instalador** https://dev.mysql.com/downloads/mysql/5.6.html

**Servicios**

**administrados**

**Clientes gráficos**

**Descargar el instalador** https://dev.mysql.com/downloads/workbench/

**SQL hasta en la sopa**

**Historia SQL**

Espacio para imagen

**SQL**

**S**tructured **Q**uery

**L**anguage

**NOSQL**

**N**ot

**O**nly

**S**tructured **Q**uery

**L**anguage

**SQL Hoy**

****

DDL

Data Definition Language

Create

Alter

Drop

**Create**

Database Table View

Create Database 

CREATE DATABASE test\_db; USE test\_db;

Create Table 

CREATE TABLE people ( person\_id int,

last\_name varchar(255), first\_name varchar(255), address varchar(255), city varchar(255)

);

Create View 

CREATE VIEW v\_brasil\_customers AS

SELECT customer\_name,

contact\_name

FROM customers

WHERE country = "Brasil";

Alter Table 

ALTER TABLE people

ADD date\_of\_birth date;

ALTER TABLE people

ALTER COLUMN date\_of\_birth year;

ALTER TABLE people

DROP COLUMN date\_of\_birth;

Drop 

DROP TABLE people; DROP DATABASE test\_db;

DML

Data Manipulation Language

Insert

Update

Delete

Select