MySQL



Programa del Curso

Clase 1: Introducción a BD + Mysql + Usos + Sentencias

Clase 2: Consultas Avanzadas + Joins + Llaves

Clase 3: Experticia en querys

Clase 4: Más Practicas + DML

Clase 5: Tipos de Datos + DER + Constraints + DDL

Diagrama Entidad Relación

Herramienta para el modelado de datos que permite representar las entidades relevantes de un sistema de información así como sus interrelaciones y propiedades.

- Entidad
- Atributos
- Claves
- Relaciones

Representa una "cosa" del mundo real. Se representa como un rectángulo.

Puede ser un objeto con existencia física o conceptual

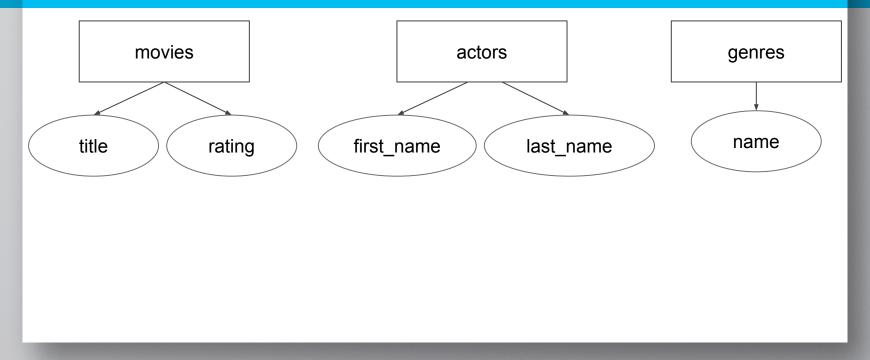
- Persona
- Animal
- Casa
- Materia
- Inscripto
- Película
- Actor

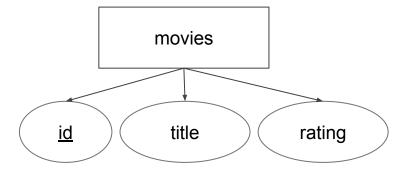
movies

actors

genres

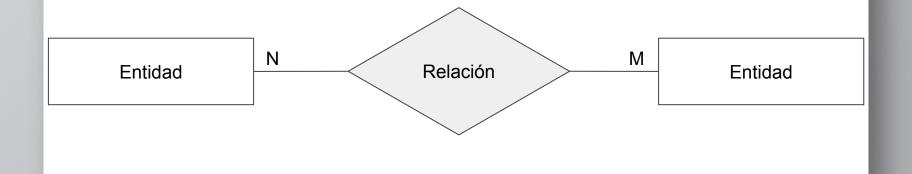
6





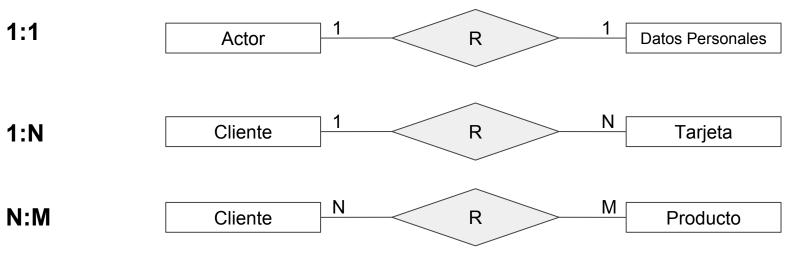
movies
id
title
rating

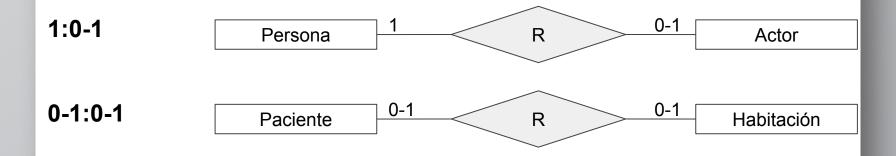
- Conexión lógica entre entidades
- Se representa con un rombo
- Puede tener atributos propios



DER - Cardinalidad

Número de entidades con la cual otra entidad puede asociarse mediante una relación.

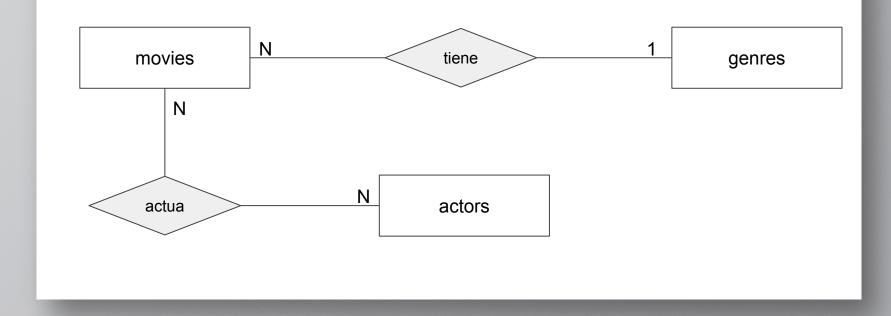




movies

genres

actors



Debate:

¿Qué cosas deberíamos tener en cuenta a la hora de diseñar una base de datos?

Guía de Ejercicios: DER - Diseño de base de datos

Tipos de Datos

- Tipos Numéricos
- Tipos de Fecha
- Tipos de Texto (String)

Tipos de Datos - Numericos

- INTEGER (INT)
 - 0 -2,147,483,648 2,147,483,647
 - Unsigned 0 4,294,967,295
- TINYINT
 - o -128 127
 - Unsigned 0 255
- BOOLEAN → No lo usen, mejor TINYINT UNSIGNED
- SMALLINT, MEDIUMINT, BIGINT
- INTEGER AUTO_INCREMENT

Tipos de Datos - Numéricos

```
SELECT *
FROM movies
WHERE awards > 5;

SELECT *
FROM movies
WHERE awards > 5 AND awards < 8;
```

Tipos de Datos - Numericos

- DECIMAL (dígitos totales, dígitos decimales)
- FLOAT
 - Puede derivar en problemas de precisión:
 - https://dev.mysql.com/doc/refman/5.7/en/problems-with-float.html

Tipos de Datos - Numéricos

```
SELECT *
FROM movies
WHERE rating > 9;

SELECT *
FROM movies
WHERE rating >= 9.1;
```

Tipos de Datos - Texto

- CHAR(100) → 100 caracteres (siempre)
- VARCHAR(100) → 100 caracteres (como máximo)
- TEXT → Texto libre (post de un blog)
- ENUM ('mal', 'regular', 'bien', 'muy bien', 'sobresaliente')

Tipos de Datos - Texto

```
SELECT *
FROM movies
WHERE title = 'Carrozas de fuego';

SELECT *
FROM movies
WHERE titles LIKE '% de %';
```

Tipos de Datos - Fecha

- DATE
- TIME
- DATETIME

Tipos de Datos - Fecha

```
SELECT *
FROM movies
WHERE release_date >= '2010-01-01';

SELECT *
FROM movies
WHERE YEAR(release_date) = 2010;
```

Constraints (restricciones)

Son restricciones a nivel tabla. Se especifica via DDL.

El servidor analiza a la hora de modificar registros (INSERT, UPDATE, DELETE)

- PRIMARY KEY
- UNIQUE
- DEFAULT
- NOT NULL

Constraints- Unique Keys

Una unique key es una restricción que solo permite valores únicos para uno (o múltiples) campos.

Ejemplos:

- DNI
- ISBN (libros)
- Número de vuelo + Fecha/hora

Constraints- Primary Key (PK) - Clave Primaria

- Una Primary Key es una UNIQUE KEY
- Una PK es un campo (o a una combinación de campos) que identifica a cada fila de una tabla de forma única.
- No puede haber dos filas en una tabla que tengan la misma clave primaria.
- No es obligatorio, pero es altamente recomendable que cada tabla tenga su PK

Constraints- NULL / NOT NULL (nulos)

- Significa que el valor para ese campo no existe o no se conoce
- NULL NO es vacío ' ' (blank) ni cero (0)
- Cualquier campo puede declararse NULL / NOT NULL
- Cuando se arma la estructura de la tabla, se define si el campo puede aceptar valores nulos.

Constraints- DEFAULT

- Se asigna dicho valor cuando no se especifica ninguno en el INSERT.
- NULL es un caso particular de valor DEFAULT

id	DNI	Apellido
1	30653013	Fejgelis
2	38222555	Susnisky
3	38222555	Weschler

id	DNI	Apellido
1		Fejgelis
2		Susnisky
3		Weschler

http://dev.mysql.com/doc/refman/5.7/en/create-table.html

```
id         INT UNSIGNED PRIMARY KEY AUTO_INCREMENT,
         titulo VARCHAR(200) NOT NULL
);
```

```
id INT UNSIGNED PRIMARY KEY AUTO_INCREMENT,
title VARCHAR(500) NOT NULL,
rating DECIMAL(3,1) UNSIGNED NOT NULL,
awards INT UNSIGNED DEFAULT 0,
release_date DATE NOT NULL,
length INT UNSIGNED NULL
);
```

DDL - Borrado de tablas

DROP TABLE [IF EXISTS] nombre_de_tabla;

ALTER TABLE nombre_de_tabla

ADD columna3 TIPO_DE_DATO... [FIRST | AFTER columna2]

MODIFY columna1 TIPO_NUEVO_DE_DATO

DROP columna4

http://dev.mysql.com/doc/refman/5.7/en/alter-table.html

```
ALTER TABLE movies
ADD rating DECIMAL(3,1) UNSIGNED NOT NULL;
```

```
ALTER TABLE movies
MODIFY rating DECIMAL(4,1) UNSIGNED NOT NULL;
```

ALTER TABLE movies DROP rating;

Ejecutemos sentencias

Hay Ejercitación Adicional para el que quiera practicar... Guía de Ejercicios: DDL (armar estructura de la BD diseñada antes) ¿Que vimos hoy?

¿Preguntas?