

**CFP**  
**Programador**  
**full-stack**

# Base de Datos

## CFP Programador full-stack

*Modelo de Entidad-Relación*

# Introducción

- La información valiosa debe tener una representación simbólica adecuada e inteligible para que pueda ser *comunicada*

## ¿Qué es un dato?

- Una pieza atómica que se utilizará para construir el concepto de información
- Una pieza atómica de dato es un valor, que debe ser interpretado como una propiedad perteneciente a un hecho o cosa del mundo real
- La interpretación es necesaria para ubicar el **dato** en **contexto**

# Modelo de Datos

- **Modelado de Datos:** organizar los datos para que representen una situación del mundo real con la mayor fidelidad posible, con el objetivo de poder manejarlos computacionalmente
- **No es posible tener un conocimiento completo del mundo real:** centrar la atención en la información relevante, ocultando o ignorando otra que **NO** sea relevante o que resulte inadecuada para el propósito requerido

# Modelo de Datos

- Comprender las características de la información relevante **para determinar cómo van a ser organizados y procesados los datos:** pueden ser descriptas a través de enunciaciones generales, por ej. lenguaje natural
- Un conjunto formal y consistente de tales enunciaciones define un **modelo conceptual de datos**

# Modelo de Datos

- La correcta estructura de la base de datos se obtiene como resultado del **modelado de los datos**
- **Modelo de datos:** Es una colección de herramientas conceptuales para describir los datos, las relaciones entre ellos, la semántica que tienen en el Universo del Discurso y las restricciones referidas a sus valores
- El **modelo de datos relacional** es uno de los paradigmas más populares, subyacente a la mayoría de las BD actuales: considera los datos bajo la forma de conjuntos, que se representa habitualmente de forma tabular

# Modelo de Datos

- El **modelo de Entidades y Relaciones (Extendido - MERE)** brinda una representación gráfica sencilla para ver los datos, relaciones y restricciones, de acuerdo al modelo relacional
  - Representa **Entidades, Atributos y Relaciones**
  - Reglas de transformación: esquema lógico según modelo relacional

# ¿Cómo Construir el Modelo de Datos?

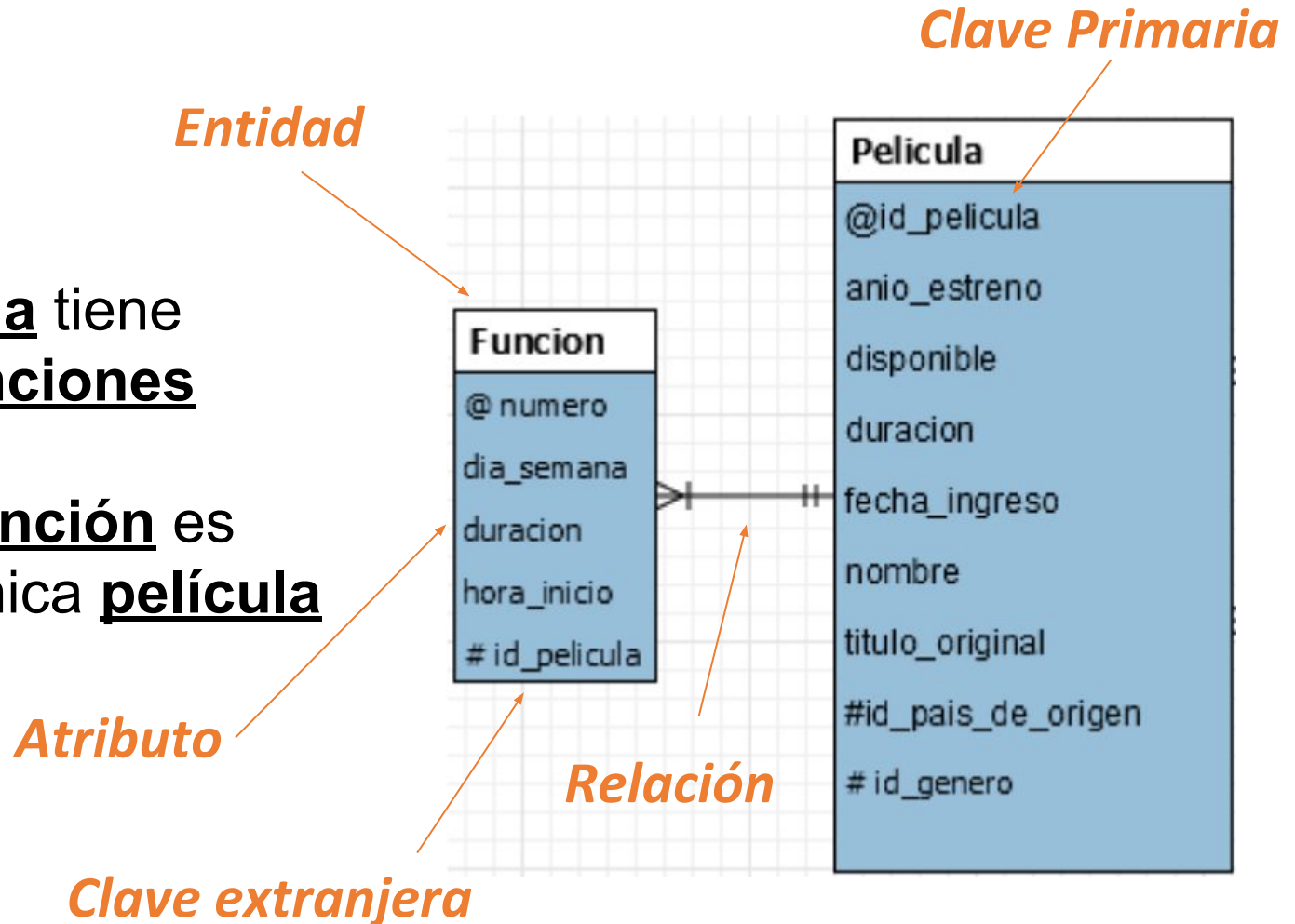
- Etapas para encontrar una forma eficaz de representar la información en el mundo computacional:
  - **Identificación de los datos de la realidad:** actores, recursos, objetos, etc. del mundo real de los cuales interesa guardar información
  - **Identificación de relaciones entre datos:** detección de los vínculos significativos que se dan entre los elementos de las etapas anteriores
  - **Abstracción de datos y relaciones:** representación simbólica sólo de los elementos detectados en las etapas anteriores



# Modelo de Datos

## Ejemplo

Una película tiene muchas funciones asignadas, pero una función es para una única película



# Entidad

- Objeto real o abstracto que existe en la realidad y del cual se desea almacenar información
- Cualquier objeto (real o abstracto) que existe en la realidad y acerca del cual se desea almacenar información en la base de datos
- Algo con realidad objetiva que existe o puede ser pensado
- Una persona, lugar, cosa, concepto o suceso, real o abstracto, de interés para la empresa
- Objetos (hechos, cosas, personas,...) que tienen propiedades en común y una existencia autónoma

# Entidad

- El elemento básico representado por el modelo entidad relación es una **entidad**, que es una cosa del mundo real con una existencia independiente
- Una **entidad** puede ser un elemento con una existencia física (por ejemplo, una persona en particular, un coche, una casa o un empleado) o puede ser un elemento con una existencia conceptual (por ejemplo, una venta, un trabajo o un curso universitario)
- Cada entidad tiene atributos (propiedades particulares que la describen)

# Entidad

- Una **entidad** es una abstracción de un conjunto de cosas del mundo real tal que:
  - Las cosas de ese conjunto tienen las mismas características o comportamiento
  - Las cosas de ese conjunto están sujetas y conformes a las mismas reglas
  - Las **entidades** que se tendrán en cuenta al modelar un sistema son aquellas que representan "cosas" de las que el sistema necesita almacenar ciertos datos

# Identificación de Entidades

- Cosas tangibles: Artículo, Repuesto, Rodado
- Roles desempeñados por personas u organizaciones: Cliente, Proveedor, Personal
- Incidentes: Usado para representar la ocurrencia de un hecho (en un sistema de una compañía de seguros: Siniestros; en una empresa de transporte: Viajes)
- Interacciones: Representan alguna transacción (Compra, Pedido, Venta, Pago)
- Es importante una **buena elección del nombre** dado a una entidad para la legibilidad y el entendimiento del modelo de datos

# Entidad

## *Ejemplo*

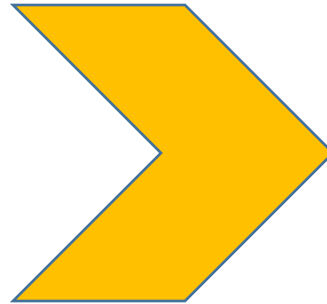
**Ejemplo:** Todos los Clientes representan la Entidad **CLIENTE**

- Todas las entidades de un conjunto tienen el mismo conjunto de atributos que interesa modelar. Todos los ejemplares de CLIENTE, tendrán Identificador de cliente, CUIT, Nombre, Apellido y demás datos descriptores
- Cada conjunto entidad necesita un **identificador**: atributo o conjunto de ellos que permita identificar a cada uno de los ejemplares que componen el conjunto entidad. La Entidad CLIENTE tiene como atributo identificador de los ejemplares a identificador de cliente

# Entidades

## *Ejemplo*

**Entidad: PELICULA**



Pelicula
anio_estreno
disponible
duracion
fecha_ingreso
nombre
titulo_original

# Atributo

Un atributo es una abstracción que identifica características, propiedades que posee una entidad. Los atributos de una entidad deben ser:

- ✓ **Completos:** capturar toda la información que interesa del objeto, desde el punto de vista del sistema
- ✓ **Plenamente elaborados:** cada atributo captura un aspecto separado de la entidad
- ✓ **Mutuamente independientes:** cada atributo debe tomar un valor independientemente de los valores asumidos por otros atributos



# Clasificación de Atributos

- **Atributos identificadores:** el o los atributos que permiten identificar unívocamente a una instancia de una entidad. Constituyen la "clave primaria"
- **Atributos descriptivos:** son las características intrínsecas de cada instancia de la entidad; como lo dice su nombre, describen a la entidad, representan sus propiedades
- **Atributos referenciales:** son atributos que sirven para relacionar entidades entre sí. Se denominan **REFERENCIALES** ya que hacen referencia al **ATRIBUTO IDENTIFICADOR** de la entidad con que se relacionan

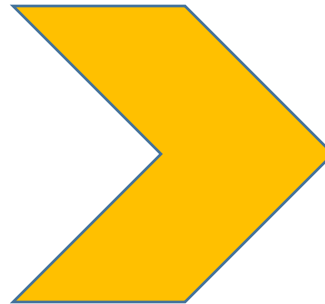
# Atributos

## *Ejemplo*

- **Entidad: PELICULA**

- **Atributos:**

- Nombre
- título original
- año de estreno,
- disponible,
- duración,
- fecha de ingreso



Pelicula
anio_estreno
disponible
duracion
fecha_ingreso
nombre
titulo_original

# Instancias de Entidad

Así podemos tener como ejemplo de entidades **PELICULA**, a dos instancias de película, con los siguientes atributos:

- **Película 1**: {2013, true, 143, 11/07/2013, "El Gran Gatsby", "The Great Gatsby"}
- **Película 2**: {2014, true, 122, 01/08/2014, "Relatos Salvajes", "Relatos Salvajes"}

A cada película de la entidad **PELICULA**, se las denomina genéricamente **instancias** de dicha entidad

# Atributo Identificador Único

- Se denomina identificador a uno o más atributos que identifican unívocamente cada instancia de una entidad; es conocido también como "clave candidata", "clave primaria"..
- Para elegir el atributo identificador debemos tener en cuenta dos reglas:
  - Que la clave sea **mínima**: Es decir elegir la alternativa en la que se necesiten menos atributos para conformar la clave
  - Elegir el atributo **más significativo** dentro del dominio del problema que se está modelando

# Atributo Identificador Único

- El atributo A, o el conjunto de atributos, de una entidad, es un posible atributo identificador si y solo si satisface dos propiedades:
  - ✓ **Unicidad:** en cualquier momento dado no existen 2 instancias con el mismo valor de A
  - ✓ **Minimalidad:** Si A es compuesto (es decir el atributo identificador está formado por más de un atributo) no será posible eliminar ningún componente de A sin destruir la propiedad de unicidad
- Toda **entidad** tiene por lo menos un atributo como posible atributo identificador. El o los atributos identificadores se señalan con el símbolo "@"(arroba), o de lo contrario con la sigla PK (clave primaria)
- Si no hay un atributo identificador, se crea uno **ficticio**

# Atributo Identificador

## *Ejemplo*

Pelicula
anio_estreno
disponible
duracion
fecha_ingreso
nombre
titulo_original

**Sin Atributo Identificador**

Pelicula
@id_pelicula
anio_estreno
disponible
duracion
fecha_ingreso
nombre
titulo_original

**Con Atributo Identificador**

# Atributo Identificador

## *Representación de un Entidad*

id_pelicula	anio_estreno	disponible	duracion	fecha_ingreso	nombre	titulo_original
1	2013	true	143	11/07/2013	"El Gran Gatsby"	"The Great Gatsby"
2	2014	true	122	01/08/2014	"Relatos Salvajes"	"Relatos Salvajes"

Este atributo agregado **@id\_película**, no es más que un número identificador que crece secuencialmente a medida que se agregan nuevas películas: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10,...n.

# Relación





- Asociación o vinculación entre **entidades**
- Una relación es la abstracción de un conjunto de asociaciones que existen entre las instancias de dos **entidades**, por ejemplo, existe una relación entre Película y PaisDeOrigen
- Las relaciones existen ya que las **entidades** representan aspectos del mundo real y en este mundo los componentes no están aislados, sino que se relacionan entre sí; es por esto que es necesario que existan las relaciones entre las **entidades**



# Cardinalidad y Opcionalidad de las Relaciones

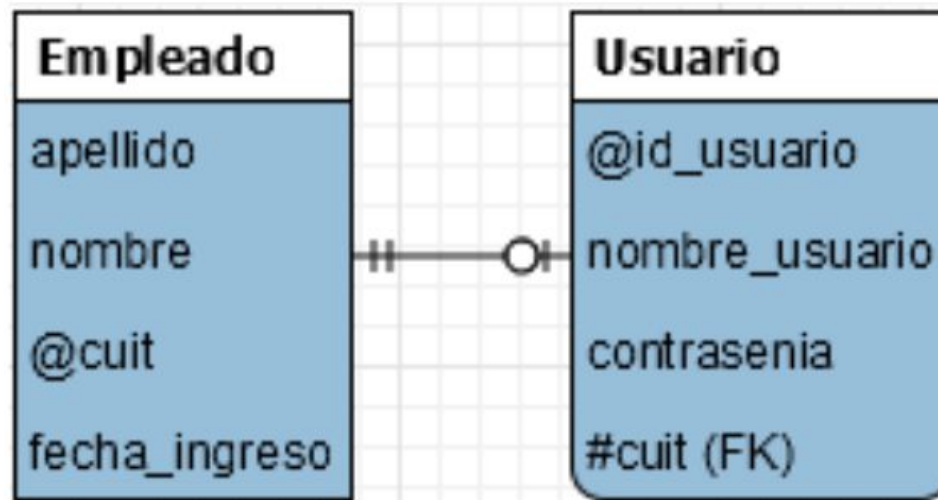
- **Cardinalidad:** Indica para una instancia de una entidad A con cuántas instancias de la entidad B se relaciona. Las posibilidades son: 0, 1 o muchos
- **Opcionalidad:** Indica para una instancia de una entidad A, si la relación con instancias de la entidad B, es opcional u obligatoria. Las posibilidades son: 0 o 1

# Simbología en Diagramas de Entidad - Relación

Tipo de Cardinalidad / Opcionalidad	Simbología	Descripción de Finalidad y Observaciones
0 ó 1	 (0,1)	Ninguna o una única ocurrencia servirá de asociación entre dos entidades cualquiera. Si la relación existe, será una única ocurrencia de la Entidad.
1 y 1	 (1,1)	Una y solo una ocurrencia estará asociando una Entidad A con una Entidad B.
0 a N	 (0, N)	Ninguna, una o varias ocurrencias asocian dos entidades del modelo de datos. Si existe podrá ser efectivizado por una o varias ocurrencias de la Entidad.
1 a N	 (1,N)	Una o varias ocurrencias servirán siempre de asociaciones entre dos entidades cualquiera del modelo de Datos.

# Caso Uno a Uno

- Un empleado puede tener o no un usuario y si ese usuario existe, es para un único empleado



- Un empleado puede tener opcionalmente un único usuario o no tener ningún usuario asociado, un usuario está asociado de manera obligatoria a un único empleado

# Caso Uno a Uno

## *Ejemplo*

### Empleado

Cuit (PK)	Nombre	Apellido	Fecha_Ingreso
220000000001	Pepe	Flores	1-1-1990
320000000002	Nene	Sanchez	1-1-1995
420000000003	Lalo	Peralta	1-1-1998

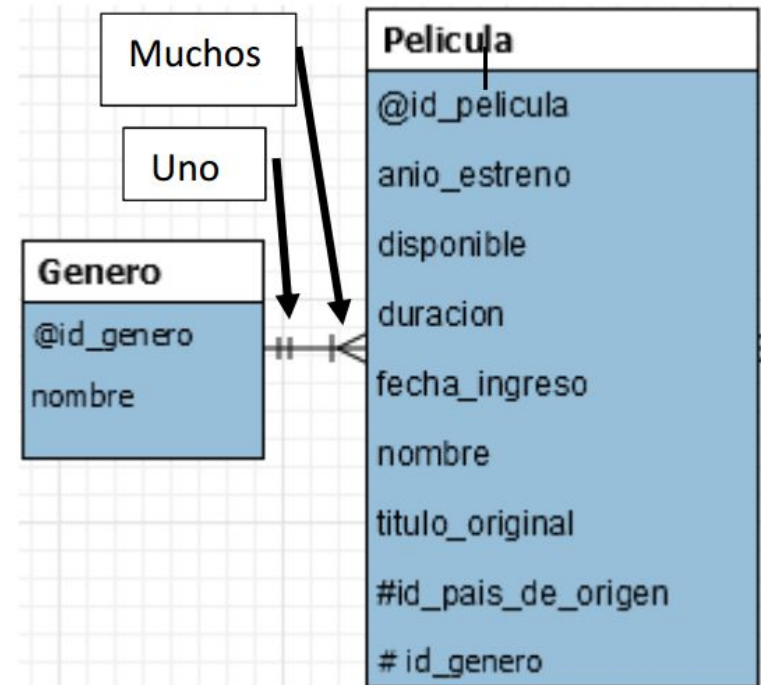
### Usuario

Id_usuario	Cuit (FK)	Nombre_usuario	Contraseña
user1	220000000001	user1@company.com	1234
user2	320000000002	user2@company.com	1432
user3	420000000003	user3@company.com	4321

# Caso Uno a Muchos

- Una película tiene un único género, pero un género como “Drama” puede estar asignado a muchas películas:

*“Un género puede estar asignado a muchas películas, pero una película tiene un único género”*



# Caso Uno a Muchos

## *Ejemplo*

### Genero

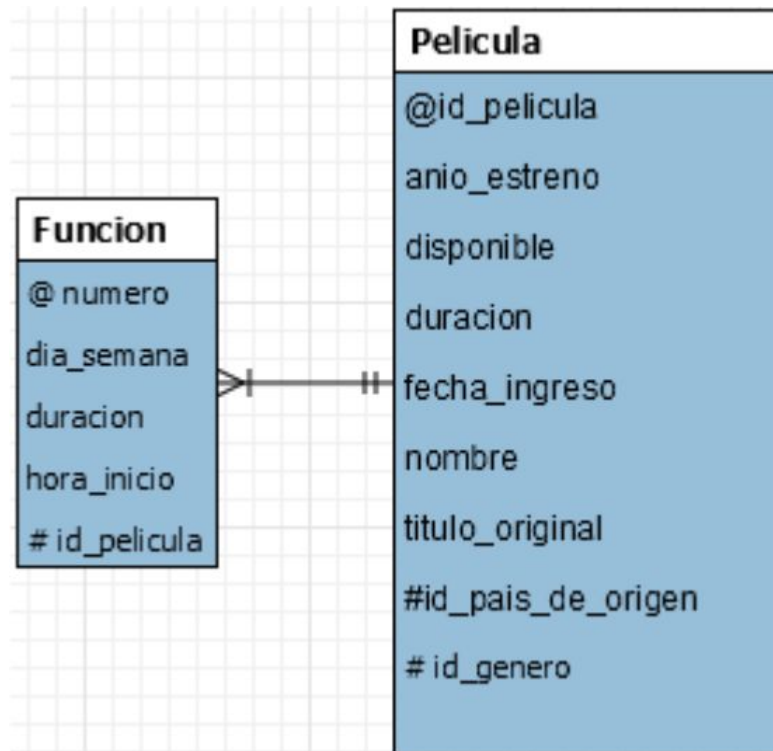
Id_genero(PK)	Nombre
gen1	Terror
gen2	Comedia
gen3	Accion

### Pelicula

id_pelicula (PK)	id_genero(FK)	anio_estreno	disponible	titulo	....
pel11	gen1	2009	true	Cenicienta	...
pel12	gen1	2008	false	Amélie	...
pel13	gen3	2012	true	Wanted	....

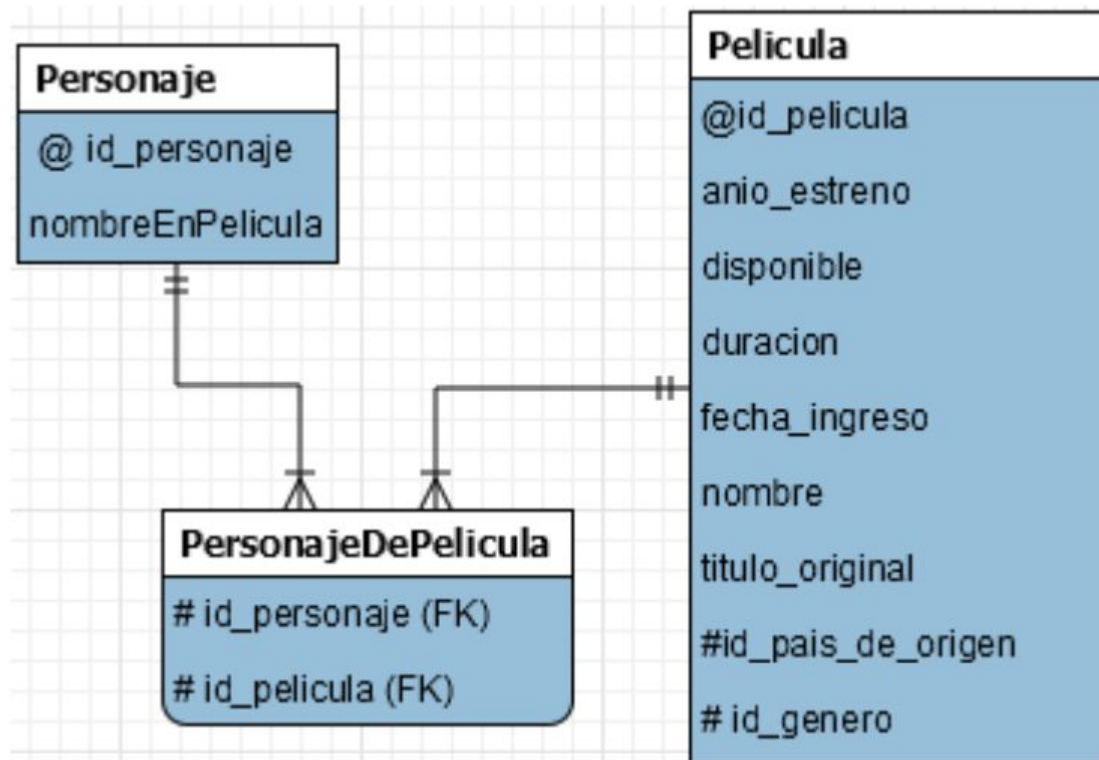
# Caso Muchos a Uno

- Una película tiene muchas funciones asignadas, pero una función es para una única película



# Casos Muchos a Muchos

Un personaje puede pertenecer a varias películas, por ejemplo: Harry Potter y la piedra filosofal, Harry Potter y la cámara secreta, Harry Potter y el prisionero de Azkaban, etc; y una película puede tener varios personajes





# Casos Muchos a Muchos

## *Ejemplo*

**Personaje**

id_personaje(PK)	Nombre
per1	ActorX
per2	ActorY
per3	ActorZ

**PersonajeDePelicula**

id_personaje(PK)	id_pelicula(PK)
per1	pel1
per1	pel2
per2	pel3

**Pelicula**

id_pelicula (PK)	id_genero(FK)	anio_estreno	disponible	titulo	....
pel1	gen1	2009	true	Cenicienta	...
pel2	gen1	2008	false	Amélie	...
pel3	gen3	2012	true	Wanted	....

# Atributo Referencial

- Un **atributo referencial/clave extranjera/clave derivada** se utiliza para establecer relaciones

entre diferentes entidades de un **modelo entidad-relación**

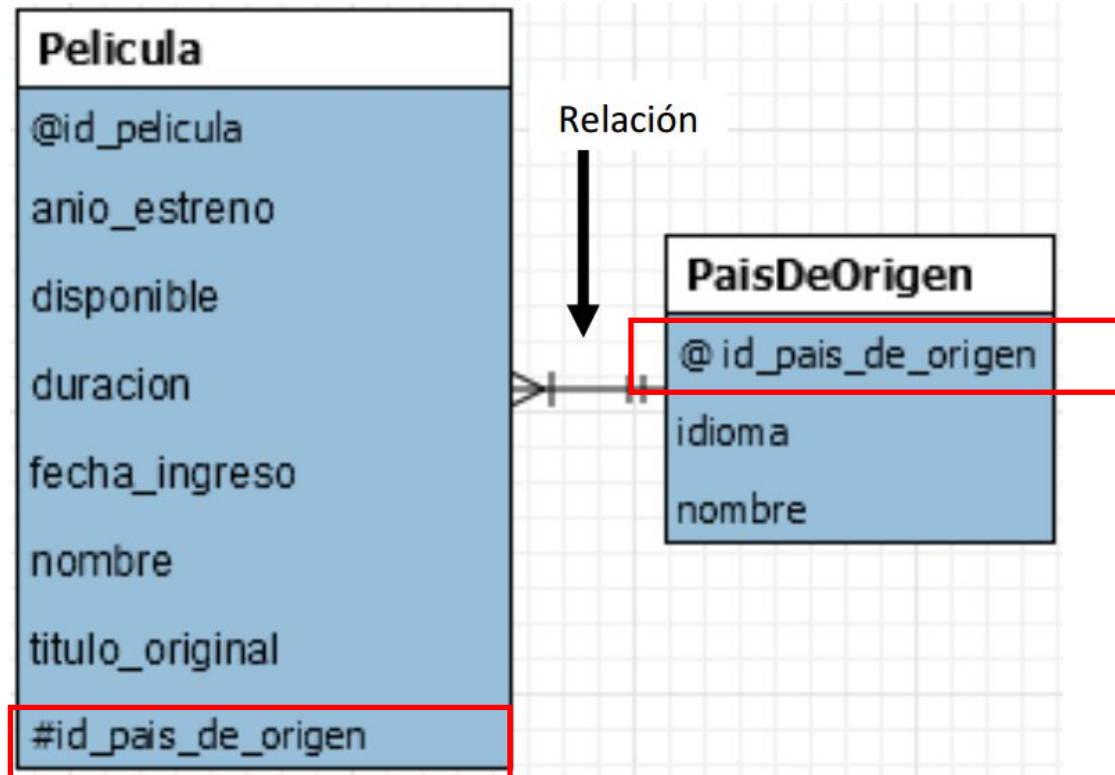
- Se dice que un atributo  $j$ , ó un conjunto de atributos, de una entidad B es un atributo referencial si y sólo si satisface dos propiedades:

- Cada valor  $j$  es nulo del todo (es decir, no existe) o no nulo del todo. En caso de ser un atributo compuesto, formado por más de un atributo
- Existe una entidad A con atributo identificador  $j$  tal que cada valor no nulo de  $j$  es en la entidad B idéntico al valor  $j$  en alguna instancia de la entidad A. Es decir que si en B el atributo  $j$  tiene valor es porque existe ese mismo valor de  $j$  en la entidad A

**Ejemplo: PELICULA y PAIS DE ORIGEN.** Se utiliza el símbolo “#” para señalar que un **atributo es referencial**, o de lo contrario con la sigla FK (clave foránea)

# Atributo Referencial

## *Entidades y sus relaciones*



# Atributo Referencial

Si tuviéramos las siguientes entidades en PaísDeOrigen:

- **País de Origen 1:** {1, "Español", Argentina}
- **País de Origen 1:** {2, "Inglés", Estados Unidos}
- **País de Origen 1:** {3, "Francés", Francia}

Entonces la entidad película, tendría en el **atributo referencial** `id_pais_de_origen`, el valor 1, que referencia a la instancia Argentina de la **entidad** PaísDeOrigen, como se ve a continuación:

- **Película 2:** { 1, 2014, true, 122, 01/08/2014, "Relatos Salvajes", "Relatos Salvajes", 1 }

# Reglas de Integridad

## *Integridad de Entidades*

- Ningún componente del atributo identificador en una entidad aceptará NULOS (nulo se considera que es inexistente, es decir, ausencia de valor)

### Ejemplo:

- **Película:** {NULL, 2014, true, 122, 01/08/2014, "Relatos Salvajes", "Relatos Salvajes",1} → **Mal**
- **Película:** { 2, 2014, true, 122, 01/08/2014, "Relatos Salvajes", "Relatos Salvajes",1} → **Bien**

# Atributo Referencial

## Reglas de Integridad Referencial

### Película

id_pelicula	id_pais_de_origen	anio_estreno	disponible	duracion	fecha_ingreso	nombre	titulo_original
1	1	2013	true	143	11/07/2013	El Gran Gatsby	The Great Gatsby
2	4	2014	true	122	01/08/2014	Relatos Salvajes	Relatos Salvajes

### País de Origen

id_pais_de_origen	idioma	pais
1	Español	Argentina
2	Inglés	Estados Unidos
3	Francés	Francia

# Reglas de Integridad

## *Conceptos*

- La regla de integridad referencial está relacionada con el concepto de **clave foránea**. Concretamente, determina que todos los valores que toma una **clave foránea** deben ser **valores que existen** en la **clave primaria** que referencia
- La necesidad de la **regla de integridad** referencial proviene del hecho de que las **claves foráneas** tienen por objetivo establecer una **conexión con la clave primaria** que referencian. Si un valor de una clave foránea no estuviese presente en la clave primaria correspondiente, representaría una referencia o una conexión incorrecta

# Reglas de Integridad

## *Conceptos - Tipos*

Esta política se puede aplicar en las siguientes operaciones de actualización que **violarían la regla de integridad**:

- a) **Borrado** de una tupla que tiene una clave primaria referenciada
- b) **Modificación** de los valores de los atributos de la clave primaria de una tupla que tiene una clave primaria referenciada

En los casos anteriores, algunas de las políticas que se podrán aplicar serán las siguientes:

- Restricción
- Actualización en cascada
- Anulación



# Valores Atómicos en Cada Celda

- **Caso 1:** Relación entre Función y Película, con Función referenciando a Película. En este caso la función tiene una única película asociada porque se proyecta una película por función

ENTIDAD FUNCIÓN				
@numero	dia_semana	duración	Hora_inico	#id_pelicula
1	3	150	22:30	2
2	4	150	18:15	2

ENTIDAD PELICULA				
@id_pelicula	Nombre	duración	#id_pais_de_origen	#id_genero
2	Relatos Salvajes	122	1	1

# Valores Atómicos en Cada Celda

- **Caso 2:** Relación entre Función y Película, con Película referenciando a Función. En este caso como la película puede proyectarse en muchas funciones, deberíamos poner en la columna identificada como **#id-función**, más de un atributo referencias, rompiendo la propiedad de atomicidad de los atributos

ENTIDAD FUNCIÓN			
@numero	dia_semana	duración	Hora_inico
1	3	150	22:30
2	4	150	18:15

ENTIDAD PELICULA					X
@id_pelicula	Nombre	duración	#id_pais_de_origen	#id_genero	#id_funcion
2	Relatos Salvajes	122	1	1	1,2

# Modelo Lógico Relacional

- El Modelo Lógico Relacional es un modelo de datos conceptual de alto nivel
- Este modelo y sus variaciones se utilizan con frecuencia para el diseño conceptual de las aplicaciones de base de datos
- Este modelo es una forma de representar los datos (mediante tablas), y la manera para manipular esa representación (utilizando operadores)
- Se ocupa de tres aspectos de los datos: su **estructura**, su **integridad** y su **manipulación**

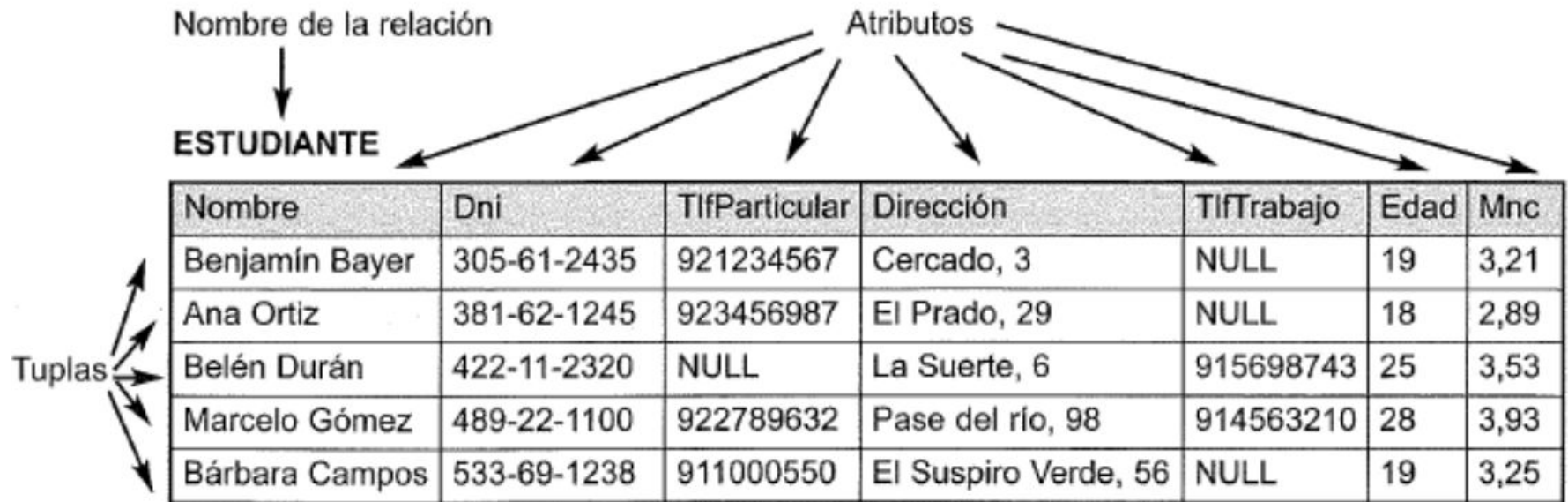
# Modelo Lógico Relacional

*Correspondencia entre modelos*

Modelo Lógico	Modelo de Datos
Relación	Entidad
Atributo	Atributo
Tupla	Instancia de la Entidad
Clave Primaria	Identificador Único de la Entidad
Dominio	Conjunto de Atributos de la Entidad

# Estructura del Modelo Lógico

En la sección anterior se presentaron los conceptos de **entidad** y de **relación** como conceptos para el **modelado de datos reales**. En el **modelo lógico**, cada **fila de la tabla** representa un hecho que, por lo general, se corresponde con **una instancia de la entidad**



# Terminología en el Modelo Lógico

- Una **relación** corresponde a lo que conocemos como **tabla**, que se utiliza para representar los datos que queremos almacenar en nuestra base de datos. En el modelado de entidad relación es la **Entidad**
- Un **atributo** corresponde a una **columna** o **campo**. El número de atributos se denomina **grado**
- Una **tupla** corresponde a una **fila** o **registro** de esa **tabla**. En el modelado de entidad relación es lo que denominamos **instancia** de la Entidad. El número de tuplas de una tabla se denomina **cardinalidad**
- La **clave primaria** es un identificador único para la tabla. Nunca existen dos filas de la tabla con el mismo valor en esa columna o combinación de columnas
- Un **dominio** es una colección de valores, de los cuales uno o más atributo (columnas) obtienen sus valores reales

# Propiedades de las Relaciones

- No existen tuplas repetidas
- Las tuplas no tienen que estar ordenadas, necesariamente, (de arriba hacia abajo)
- Los atributos no tienen que estar ordenados, necesariamente, (de izquierda a derecha)
- Todos los valores de los atributos son atómicos (esto significa que debe tener un único valor, por ejemplo si es un atributo que contendrá un número de teléfono, solo puede contener un número de teléfono)

# Integridad en las Bases de Datos Relacionales

- El objetivo de las reglas de integridad es informar al sistema administrador de Bases de Datos (DBMS), de ciertas restricciones del mundo real, (por ejemplo, los pesos de las piezas no pueden ser negativos)
- Se debe vigilar las operaciones de inserción y modificación y rechazar cualquier entrada que no cumpla con las especificaciones
- El Modelo Relacional incluye dos reglas generales de integridad: *Reglas de Integridad para Clave Primaria* y *Reglas de Integridad para Clave Foránea*



# Reglas de Integridad para Clave Primaria

- Un atributo  $a$  (posiblemente compuesto) de la relación  $R$  es una clave candidata de  $R$ , sí y solo sí satisface las siguientes propiedades:
  - **Unicidad:** no existen dos tuplas de  $R$  con el mismo valor de  $a$ , en un momento dado
  - **Minimalidad:** si  $a$  es un atributo compuesto, no puedo eliminar un componente de  $a$  sin destruir la propiedad de unicidad
- En el caso de que existan varias claves candidatas, debemos escoger una y las demás serán claves Alternativas (claves candidata que no son clave primaria)

# Reglas de Integridad para Clave Primaria

## *Ejemplo*

PersonajeDePelicula

id_personaje(PK)	id_pelicula(PK)
per1	pel1
per1	pel2
per2	pel3

De este modo, se garantiza que todas las filas de la tabla **ProductVendor** tengan una combinación de **ProductID** y **VendorID**. Esto impide la inserción de filas duplicadas.

# ¿Por Qué son Importantes las Claves Primarias?

- Son importantes porque constituyen el mecanismo de direccionamiento a nivel de tuplas, básico en un sistema relacional
- Es el único modo garantizado por el sistema para localizar una tupla específica
- Ningún componente de la clave primaria de una relación base puede aceptar nulos. Se entiende por nulo a un valor o representación que, por convención, no representa valor real del atributo aplicado

# ¿Por Qué son Importantes las Claves Primarias?

En una base de datos relacional, nunca registraremos información de algo que no podamos identificar. Consideraciones para esta regla:

- Para las claves primarias compuestas: cada valor individual de la clave primaria debe ser no nulo en su totalidad
- Esta regla se aplica a las relaciones base únicamente
- Se aplica sólo a las claves primarias
- En caso de que la clave primaria fuera compuesta, se utiliza la abreviación PK para cada atributo que forma parte de la clave primaria

# Reglas de Integridad para Clave Foránea

- La **clave foránea** (FOREIGN KEY) es un atributo de una **relación** R2 cuyos valores deben concordar con los de la clave primaria de alguna otra **relación** R1 (no se requiere el caso inverso).
- La **integridad referencial** se formula en términos de estados de la base de datos. Para cada clave foránea es necesario responder tres preguntas:
  - ¿Puede aceptar nulos esa clave ajena?
  - ¿Qué deberá suceder si hay un intento de eliminar el objetivo de una referencia de clave ajena?
  - ¿Qué deberá suceder si hay un intento de modificar la clave primaria del objetivo de una referencia?

# Reglas de Integridad para Clave Foránea

## *Ejemplo - Restringido*

- Si aplicamos la restricción en caso de borrado y, por ejemplo, queremos borrar al **género *gen1***, no podremos hacerlo porque **películas** que lo referencian.

**Género**

Id_genero(PK)	Nombre
gen1	Terror
gen2	Comedia
gen3	Accion

**Película**

id_pelicula (PK)	id_genero(FK)	anio_estreno	disponible	titulo	....
pe1i1	gen1	2009	true	Cenicienta	...
pe1i2	gen1	2008	false	Amélie	...
pe1i3	gen3	2012	true	Wanted	....

# Reglas de Integridad para Clave Foránea

## *Ejemplo - Cascada*

- Si aplicamos la actualización en cascada en caso de borrado y, por ejemplo, queremos borrar el **género gen3**, se borrarán también la **película peli3** que hay en la tabla **película**

Género

Id_genero(PK)	Nombre
gen1	Terror
gen2	Comedia
gen3	Accion

Película

id_pelicula (PK)	id_genero(FK)	anio_estreno	disponible	titulo	....
pel1	gen1	2009	true	Cenicienta	...
pel2	gen1	2008	false	Amélie	...
pel3	gen3	2012	true	Wanted	....

# Reglas de Integridad para Clave Foránea

## *Ejemplo - Anulación*

- Si aplicamos la anulación en caso de borrado y, por ejemplo, queremos borrar al **género gen3**, se modificarán todas las **películas** que lo tenían asignado, y pasarán a tener un valor **nulo** en **id\_genero**.

Género

Id_genero(PK)	Nombre
gen1	Terror
gen2	Comedia
gen3	Accion

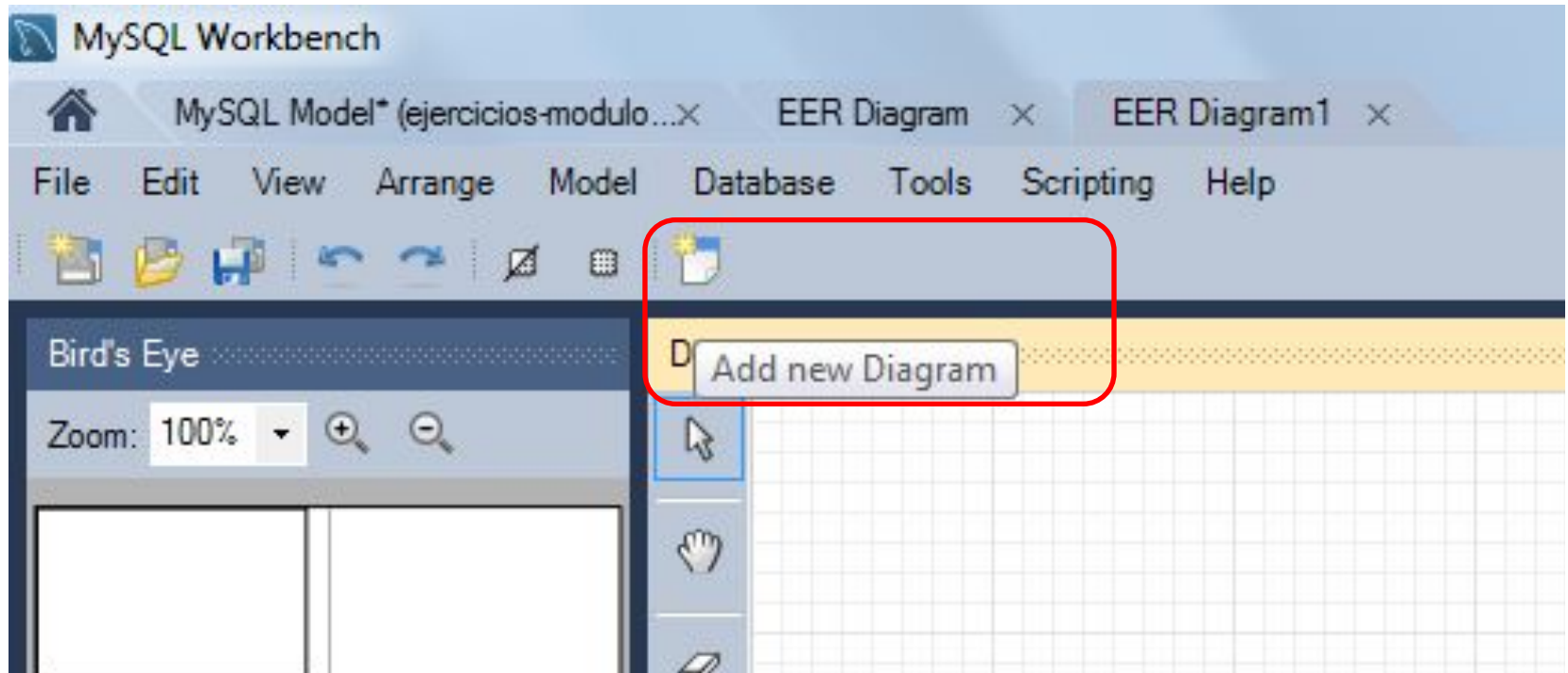
Pelicula

id_pelicula (PK)	id_genero(FK)	anio_estreno	disponible	titulo	....
pe1i1	gen1	2009	true	Cenicienta	...
pe1i2	gen1	2008	false	Amélie	...
pe1i3	<b>NULO</b>	2012	true	Wanted	....



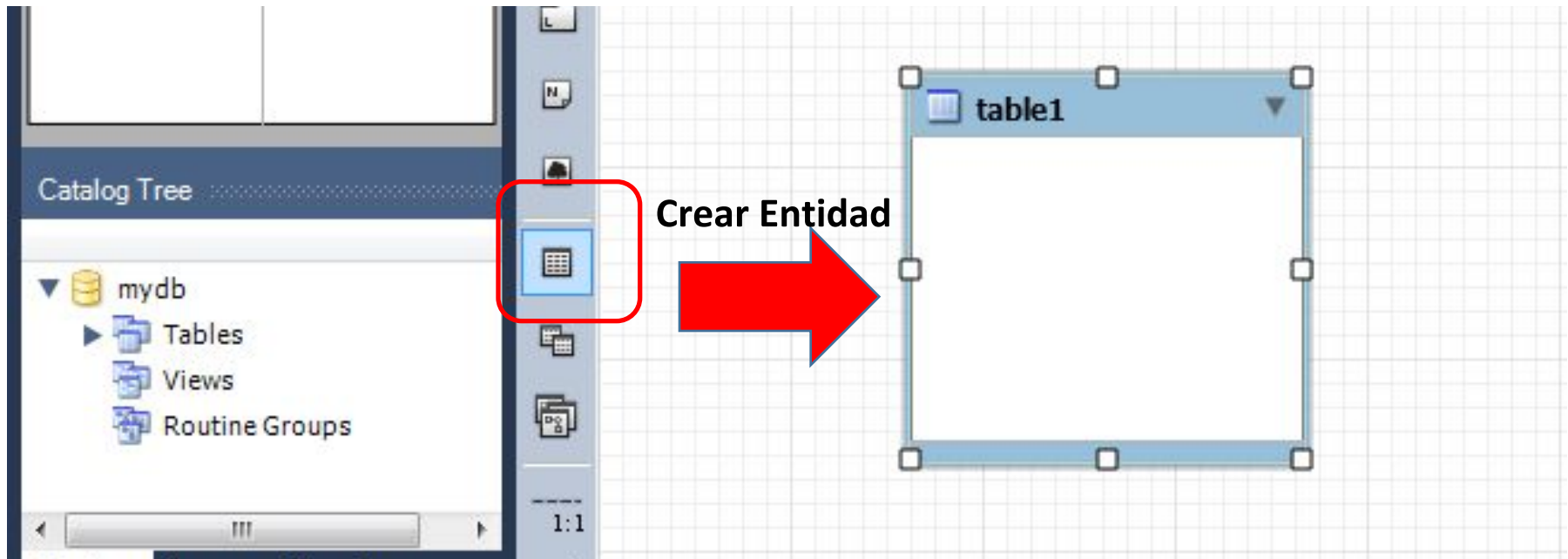
# Creación de un DER

## *Tutorial de MySQL Workbench*



# Creación de un DER

## *Tutorial de MySQL Workbench*



# Creación de un DER

## *Tutorial de MySQL Workbench*

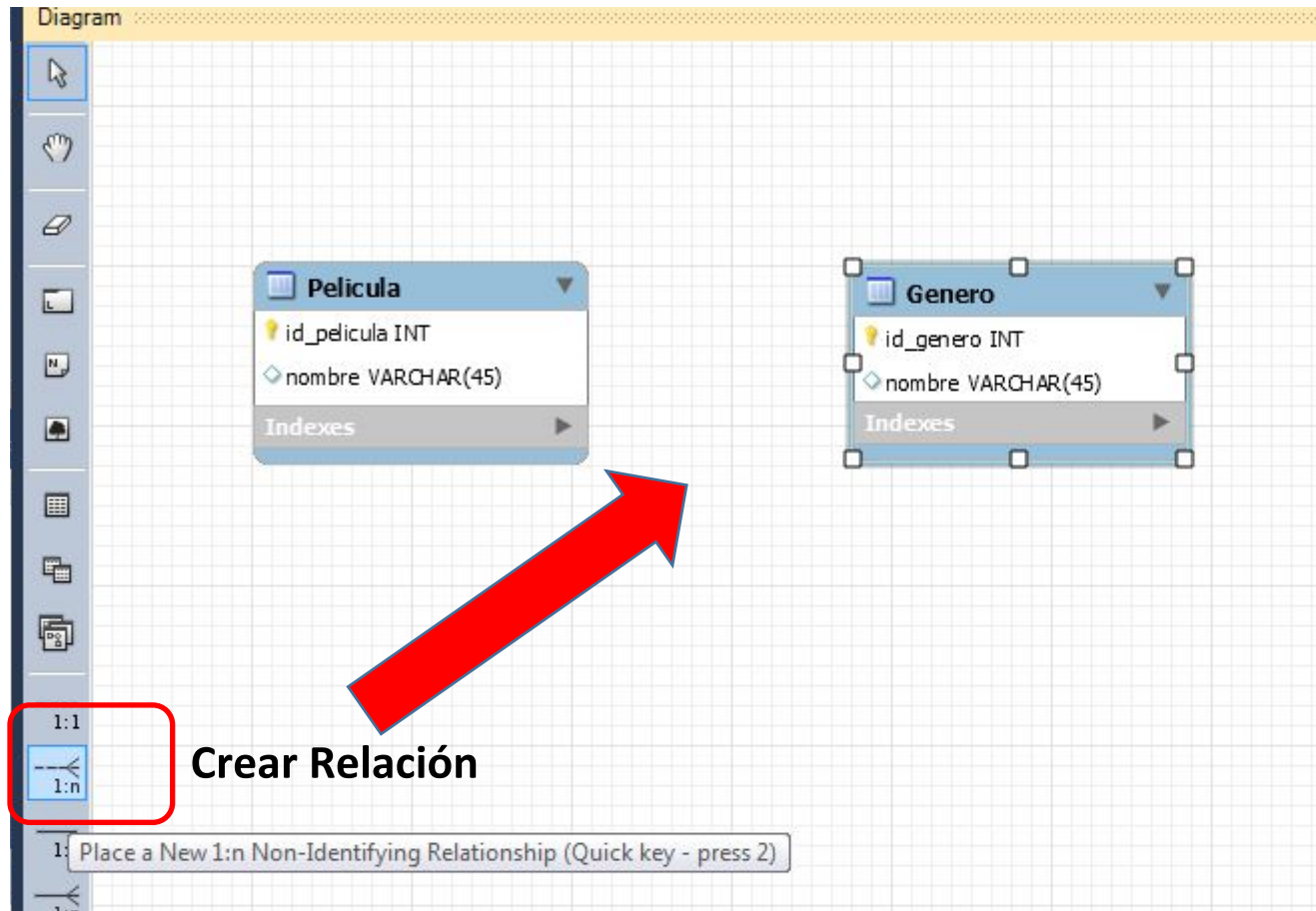
**Edición de la entidad**  
**Creación de atributos**

**Table Editor: Pelicula - Table**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Exp
id_pelicula	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
nombre	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

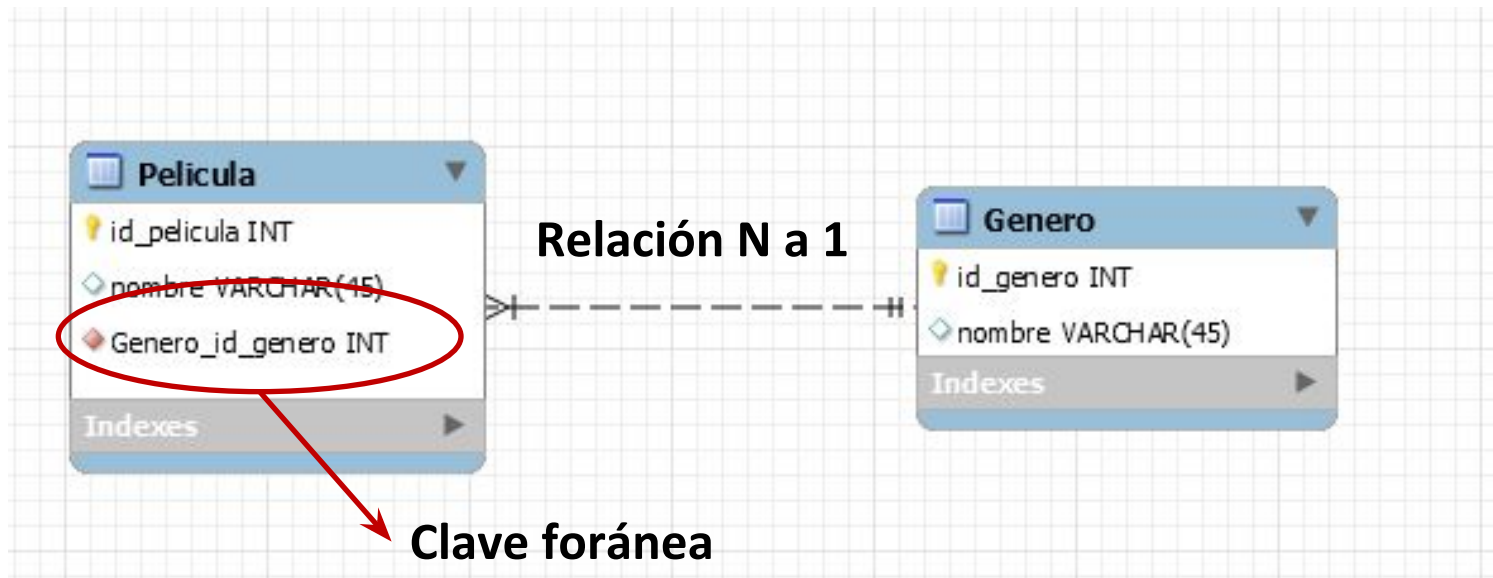
# Creación de un DER

## *Tutorial de MySQL Workbench*



# Creación de un DER

## *Tutorial de MySQL Workbench*



Base de Datos

**CFP**  
**Programador**  
**full-stack**

*Modelo de Datos (Ejercicios)*

# Modelo de Entidad-Relación

1. Identifique las entidades y sus relaciones, para éstas últimas indique su cardinalidad mínima y máxima. Agregue los atributos de cada entidad o relación (si los tuviera) con todas sus características:
  - Cada empleado está representado por un número de empleado, su nombre y dirección. Se debe indicar además a qué departamento pertenece
  - Cada departamento está representado por su nombre y se saben que empleados trabajan en él, quién es el jefe del departamento y los productos que vende
  - Cada producto está representado por su nombre, fabricante, precio, número de producto asignado por el fabricante y número de producto asignado por el almacén
  - Cada fabricante está representado por su nombre, dirección, productos que suministra al almacén y precios de estos productos

# Relaciones y su Cardinalidad

1. A partir de los siguientes enunciados, modele con un DER las relaciones con su cardinalidad. Incluya el modelado completo de entidades
  - A. Un empleado puede tener o no un usuario y si ese usuario existe, es para un único empleado
  - B. Una película tiene un único género, pero un género como “Drama” puede estar asignado a muchas películas
  - C. Una película tiene muchas funciones asignadas, pero una función es para una única película



# Entidades

- *Dado el siguiente enunciado identifique las entidades:*

Una empresa aeronáutica tiene empleados, de los cuales se sabe su legajo. Existen 3 tipos de empleados: operarios, gerentes y auditores. Los operarios son evaluados periódicamente por su performance. Los gerentes aprueban fondos para proyectos de construcción de aviones y los auditores controlan la calidad de los procesos de construcción de aviones. Los empleados pueden pertenecer a distintos equipos de trabajo, los cuales requieren al menos 2 empleados. Estos equipos son los encargados de construir aviones. Cada equipo siempre está activo construyendo al menos un avión. Cada avión está formado por distintos componentes. En caso de dar de baja un avión por desperfectos, el auditor toma la decisión de descartar la reutilización de sus componentes para minimizar riesgos.

# Atributos y Relaciones

1. En base al enunciado anterior, identifique los atributos de las entidades reconocidas
2. En base al enunciado anterior, identifique las relaciones indicando cardinalidad y opcionalidad

# Base de Datos

## CFP Programador full-stack

*Modelo de Datos (Repaso)*

# Modelo de Datos

## *Repaso*

- Un modelo de datos es una colección de herramientas conceptuales para describir los datos, las relaciones entre ellos, la semántica que tienen en el Universo del Discurso y las restricciones referidas a sus valores
- Una entidad es un objeto real o abstracto que existe en la realidad y acerca del cual se desea almacenar información
- Un atributo es una abstracción que identifica características, propiedades que posee una entidad.
- Una relación es la abstracción de un conjunto de asociaciones que existen entre las instancias de dos **entidades**

# Relaciones, Cardinalidad y Opcionalidad

- La cardinalidad indica para una instancia de una entidad A con cuántas instancias de la entidad B se relaciona. Las posibilidades son: 0, 1 o muchos
- La opcionalidad indica para una instancia de una entidad A, si la relación con instancias de la entidad B, es opcional u obligatoria. Las posibilidades son: 0 o 1
- El modelo lógico es una forma de representar los datos (mediante tablas), y la manera para manipular esa representación (utilizando operadores)