**Actividad: Elaboración de un video con un tema desarrollado en el curso de Bases de Datos**

Selecciona el lugar adecuado, con la iluminación adecuada, determina tus herramientas, como cámara, micrófono, software para animación, tripié, prompter. Procede a redactar el guion toma en cuenta las características que debe tener y se enuncian a continuación.

Características:

* Producción de un Video referente a alguno de los temas del curso.
* Duración del video entre 10 y 15 minutos.
* Con introducción y musicalización.
* Con cortinas para cambios de pantalla.
* Con efectos musicales de cambio de pantalla en forma gradual incremental o en decrecimiento.
* Elaborar un guion, con presentación del tema, desarrollo y conclusión.
* Elaborar un guion gráfico (storyboard), que incluya las imágenes a incorporar con sus enlaces al guion escrito.
* Escribir como si estuvieras charlando con un amigo.
* Revisar el guion para encontrar si se puede incorporar una imagen al video.
* Realiza el video donde aparezcas como protagonista hablando a la cámara.
* Los ojos del personaje que habla deberá ver directamente a la cámara sin mover los ojos y sin parpadear.
* El personaje que aparezca en la escena debe ocupar al menos un 60% de la ventana de la pantalla.
* Utilizar un prompter para seguir el guion tal cual lo elaboraste, existen aplicaciones en teléfonos inteligentes que son de uso libre los puedes utilizar.
* La iluminación es importante, debes cuidar que se vea con claridad y luminosidad.
* El fondo sobre el cual se graba puedes ser en una tela verde liso de tal forma que el fondo se pueda intercambiar al momento de editar.
* Editar el video para incorporar las imágenes, gráficas o interfaces en otro software.

Puedes obtener sugerencias de cómo elaborarlo el La Fonda Filosófica, un sitio del DR Darin McNabb, profesor de filosofía y teoría política de la Universidad Veracruzana.

Bibliografía.

McNabb, Darinn (2010). *Como elaboro mis videos* [video]. Disponible en:

<https://www.youtube.com/watch?v=o9NyMM0k4bI> Recuperado el día 5 de mayo de 2016.

Galué Javier canal en youtubecom [www.javiergalue.com](http://www.javiergalue.com)

**Script Carpeta Clase 25**

|  |  |  |
| --- | --- | --- |
| Tema | En I / D / C | Persona T / M |
| ¿Qué es una base de Datos? | I | T |
| ¿Qué es SQL? | I | M |
| ¿Qué es una tabla? (Fila, columna, tupla) | I | T |
| Modelo Entidad Relación EER | I | M |
| Llave primaria | D | T |
| Llave Foránea | D | M |
| Llave sustituta | D | T |
| Clase | D | M |
| Subclase | D | T |
| Super Clase | D | M |
| Relación  Abarcar todas (resumida) Muchos a Muchos  (Cardinalidad N:N) | D | 1 |
| Modelado de Categorías Tipo Unión | D | 1 |
| Fig 8.8 | D | 1 |
| Fig 9.7 | D | 1 |
| ¿Por qué es importante saber realizar un esquema de forma adecuada? | C | 1 |
| Importancia del conocimiento de las cardinalidades en los esquemas para la elaboración de una base de datos. | C | 1 |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | |
|  |  |  | |
|  |  |  |

# ¿Qué es una base de Datos?

Una base de datos es una colección organizada de información estructurada, o datos, típicamente almacenados electrónicamente en un sistema de computadora. Una base de datos es usualmente controlada por un sistema de gestión de base de datos (DBMS). En conjunto, los datos y el DBMS, junto con las aplicaciones que están asociados con ellos, se conocen como un sistema de base de datos, que a menudo se reducen a solo base de datos.

# ¿Qué es SQL?

El Lenguaje de Consulta Estructurado popularmente conocido por sus siglas en inglés como SQL, es un tipo de lenguaje de programación que ayuda a solucionar problemas específicos o relacionados con la definición, manipulación e integridad de la información representada por los datos que se almacenan en las bases de datos.

Algunos aspectos de SQL están basados en el cálculo relacional, algunos en el álgebra relacional que provienen del modelo relacional y otros a ninguno de los dos sino que son parte de SQL.

SQL es un lenguaje de alto nivel orientado a conjuntos de registros. Esto implica que un solo comando SQL puede equivaler a decenas o cientos de líneas de código que se tendrían que utilizar en un lenguaje de más bajo nivel orientado a registros.

# ¿Qué es una tabla? (Fila, columna, tupla)

Las tablas en las bases de datos son todas aquellas que contienen la información de un sistema o un programa. Estas tablas contienen diversas columnas en representación de lo que se quiere dentro de la aplicación que se crea. Por lo general, podemos encontrar bases de datos simples donde se inserta un nombre, un apellido, un correo y un número telefónico.

Dentro de las tablas encontramos la información organizada en filas y columnas y a cada uno de los registros que estas contienen las llamamos tuplas.

# Modelo Entidad Relación EER

El modelo entidad-relación es una herramienta para generar el modelo de datos que describe la estructura y relaciones de una BD. Estos modelos al mismo tiempo están describiendo una situación real, con elementos reales que se relacionan entre sí

El modelo entidad-relación es un diagrama que ayuda a generar la estructura de datos con la que gestionar un problema o actividad real. Una vez este modelo se ha convertido en una estructura dentro la BD, es decir, las tablas con sus claves primarias y foráneas, mediante SQL es posible tanto mantener el funcionamiento de la actividad alimentando la base de datos, como analizar los datos en beneficio de la actividad. Por ejemplo, en el caso del almacén de fruta, la estructura de datos debería permitir registrar pedidos de los clientes, pero también y en consecuencia, obtener las ventas por cliente en un periodo determinado. En el caso de este foro, la estructura de datos permite registrar nuevos usuarios, pero también conocer cuantos usuarios hay registrados hasta la fecha, o cuantos de ellos están online en un momento dado.

Entidades fuertes y débiles

Existen dos tipos de entidades, las fuertes, en ocasiones llamadas maestros, que de forma independiente identifican sus registros con un clave propia, y las débiles que dependen de una entidad fuerte para identificar sus registros

--Resumen--

El modelo entidad-relación es una herramienta en forma de diagrama que ayuda a generar la estructura de datos con la que gestionar un problema o actividad real, es decir las tablas con sus claves en una BD relacional.

Cuanto mayor sea el grado de conocimiento de la actividad a gestionar tanto mejor para desarrollar el modelo entidad-relación. Esta es un ejercicio creativo donde la teoría al respecto ayuda pero no enseña a desempeñarlo con soltura, solo se adquiere con la práctica y experiencia.

Una vez se tiene un modelo desarrollado la traducción a objetos de BD es directa aplicando una serie de pasos.

En un modelo entidad-relación encontraremos esencialmente relaciones de dos tipos: de 1 a N y, de N a M. También encontraremos dos tipos de entidades: fuertes y débiles. Una entidad débil necesitará la clave de la entidad fuerte para identificar sus registros. La cardinalidad de una relación de una entidad débil con su maestro o entidad fuerte siempre será de 1 a N.

# Llave primaria

Conjunto de uno o más atributos de una tabla, que tomados colectivamente nos permiten identificar un registro como único, es decir, en una tabla podemos saber cual es un registro en específico sólo con conocer la llave primaria.

En una arquitectura entidad-relación la llave primaria permite las relaciones de la tabla que tiene la llave primaria, con otras tablas que van a utilizar la información de esta tabla.

# Llave Foránea

La clave externa o FOREIGN KEY, es una columna o varias columnas, que sirven para señalar cual es la clave primaria de otra tabla.

La columna o columnas señaladas como FOREIGN KEY, solo podrán tener valores que ya existan en la clave primaria PRIMARY KEY de la otra tabla.

# Llave sustituta / Candidata

Llamamos llave candidata de una relación (o simplemente llave) al atributo o conjunto de atributos que tienen la propiedad de identificar unívocamente a una tupla dentro de la relación.

# Clase

A UML **class** (ER term: **entity type**) is used to model any “thing” in the enterprise that is to be represented in our database. As mentioned, it could be a physical “thing” (e.g., a store, an animal, etc) or simply a fact about the enterprise or an event that happens in the real world (e.g., the lending of a book). We are interested in the following important aspects of a UML class.

Una clase UML (término ER: tipo de entidad) se utiliza para modelar cualquier "cosa" en la empresa que se va a representar en nuestra base de datos. Como se mencionó, podría ser una "cosa" física (por ejemplo, una tienda, un animal, etc.) o simplemente un hecho sobre la empresa o un evento que ocurre en el mundo real (por ejemplo, el préstamo de un libro). Estamos interesados en los siguientes aspectos importantes de una clase UML.

# Subclase

El conjunto de subclases se define basándonos en características diferenciadoras de las ocurrencias de entidad de la superclase. Por ejemplo, el conjunto se subclases {SECRETARIA, INGENIERO, TECNICO} es una especialización de la superclase EMPLEADO mediante la distinción del tipo de trabajo en cada ocurrencia de entidad. Podemos tener varias especializaciones de una misma entidad basándonos en distintos criterios. Por ejemplo, otra especialización de EMPLEADO podría dar lugar a las subclases ASALARIADO y SUBCONTRATADO, dependiendo del tipo de contrato.

# Super Clase

A superclass is the class from which many subclasses can be created. The subclasses inherit the characteristics of a superclass. The superclass is also known as the parent class or base class.

# Relación

Es una asociación o correspondencia existente entre una o varias entidades. La relación puede ser regular, si asocia tipos de entidad regulares, o débil, si asocia un tipo de entidad débil con un tipo de entidad regular. Dentro de las relaciones débiles se distinguen la dependencia en existencia y la dependencia en identificación.

Se dice que la dependencia es en existencia cuando las ocurrencias de un tipo de entidad débil no pueden existir sin la ocurrencia de la entidad regular de la que dependen. Se dice que la dependencia es en identificación cuando, además de lo anterior, las ocurrencias del tipo de entidad débil no se pueden identificar sólo mediante sus propios atributos, sino que se les tiene que añadir el identificador de la ocurrencia de la entidad regular de la cual dependen.

Además, se dice que una relación es exclusiva cuando la existencia de una relación entre dos tipos de entidades implica la no existencia de las otras relaciones.

Una relación se caracteriza por:

* Nombre: que lo distingue unívocamente del resto de relaciones del modelo.
* Tipo de correspondencia: es el número máximo de ocurrencias de cada tipo de entidad que pueden intervenir en una ocurrencia de la relación que se está tratando.

# Abarcar todas (resumida)

Es el número máximo de ocurrencias de cada tipo de entidad que pueden intervenir en una ocurrencia de la relación que se está tratando.  
Conceptualmente se pueden identificar tres clases de relaciones:

* 1. Relaciones 1:1: Cada ocurrencia de una entidad se relaciona con una y sólo una ocurrencia de la otra entidad.
  2. Relaciones 1:N: Cada ocurrencia de una entidad puede estar relacionada con cero, una o varias ocurrencias de la otra entidad.
  3. Relaciones M:N: Cada ocurrencia de una entidad puede estar relacionada con cero, una o varias ocurrencias de la otra entidad y cada ocurrencia de la otra entidad puede corresponder a cero, una o varias ocurrencias de la primera.

Cardinalidad: representa la participación en la relación de cada una de las entidades afectadas, es decir, el número máximo y mínimo de ocurrencias de un tipo de entidad que pueden estar interrelacionadas con una ocurrencia de otro tipo de entidad. La cardinalidad máxima coincide con el tipo de correspondencia.

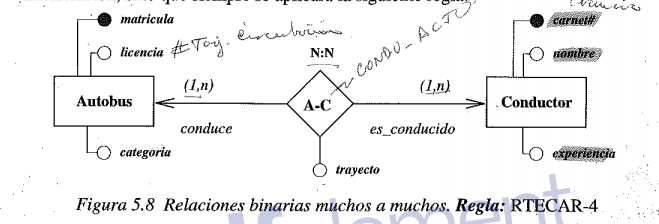
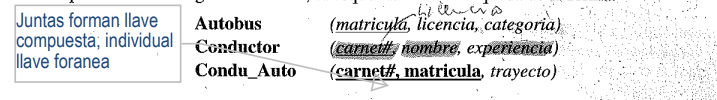
# (Cardinalidad N:N)

**TRANSFORMACIÓN DE TIPOS DE INTERRELACIÓN MUCHOS A MUCHOS (N:N)**

**NO DEPENDE DE LA CARDINALIDAD MÍNIMA, SIEMPRE SE APLICA LA SIGUIENTE REGLA**

**RTECAR – 4 (1,N) (1,N)**

* Cada tipo de entidad se transforma en una tabla por aplicación de **RTECAR-1**
* Se genera una nueva tabla para representar al tipo de interrelación
* **Identificadores** de los tipos de entidad que intervienen y por los atributos asociados al tipo de interrelación.

**Clave principal** Agregación de los atributos **identificadores** correspondientesa los tipos de entidad

**ENTIDAD DÉBIL**

* Su existencia depende de otra entidad
* Posee una llave primaria parcial (esta es tipo provisional)

La tabla de esta entidad tiene:

* Llave primaria: Llave primaria de entidad fuerte con el que se relaciona más llave parcial

**ATRIBUTO MULTÍVALUADO**

* Genera una tabla nueva
* Llave Primaria Compuesta: Propia y la llave primaria de la entidad a la que pertenece.
* Llave Foránea: Llave primaria de la entidad a la que pertenece

# Normalización

Los atributos de una entidad se separan y se añaden a entidades recién creadas mediante un proceso que se denomina normalización. El resultado de este proceso es que cada entidad tiene una clave primaria. Este es un atributo que identifica de forma única cualquier instancia de esa entidad. Todos los atributos de la clave deben depender por completo de esta. La clave primaria debe estar formada por una combinación de atributos. Esto se denomina clave compuesta.

# Modelado de Categorías Tipo Unión

# Fig 8.8

Dentro de esta figura podemos observar la existencia de 5 entidades: **Person, Bank, Company, Car, Truck,** las cuales poseen distintos atributosy en especial el que esta subrayado es aquel que identificaremos como atributo llave, pues es el que caracteriza de forma única a cada una de las entidades que participan en el diagrama.

Observamos que existe una relación de muchas a muchos

# Fig 9.7

# ¿Por qué es importante saber realizar un esquema de forma adecuada?

Un diagrama de entidad-relación (ERD por sus siglas en inglés) es parte del proceso de diseño de una base de datos relacional. Un analista comienza por recolectar todos los tipos de datos que encuentra en una organización y los representa como cajas en un diagrama. Las cajas se relacionan entre sí mediante enlaces, representados por una línea, que puede terminar en una pata de gallo, a fin de indicar una relación de una con muchas.

La base de datos relacional se crea con el ERD. Las entidades se convierten en tablas y los atributos son columnas de esas tablas. Las claves primarias se convierten en índices de la tabla y las externas le indican al programador cómo debe enlazar dos tablas. El ERD que diseñó la base de datos se mantiene como referencia para utilizar como un mapa que le indique cómo llegar de una tabla a la otra.

Es por eso que se debe tener una planificación anticipada, pues como vemos al momento de realizar la base de datos ya se debe tener un mapeo de cómo se van a ir relacionando las tablas y mediante que “conectores / llaves” esto con el propósito de tener todos los datos como debe ser y poder acceder a ellos cuando esto sea necesario, así como evitar la redundancia en los datos o la generación de las llamadas tuplas “espuria”

# Importancia del conocimiento de las cardinalidades en los esquemas para la elaboración de una base de datos.

La importancia de este tema radica principalmente en que la cardinalidad es un aspecto fundamental, pues este nos indicará de acuerdo a las reglas de la normalización si es correcto o bien necesario la generación de una nueva tabla dentro de nuestro esquema o la forma en que dos tablas serán relacionadas, así como el atributo que cada una heredaría o sería establecido como llave primaria o foránea

Guion Elaboración Vídeo Base de Datos

Actividad: **Modelado de Categorías tipo unión**: considere el tipo propietario, que se subdivide en los conjuntos compañía, banco y persona, que se conecta con una interrelación propietario con cardinalidades M: N con vehículo registrado que puede ser un carro o un camión, fig 8.8 Elmasry & Navathe. Fig 9.7

**Elmasry & Navathe.**

Elaborar diagrama entidad relación extendido, el esquema relacional, poblar la base con datos ejemplo. Hacer consultas al esquema.

|  |  |  |
| --- | --- | --- |
| PERSONA | DIALOGO | TEMA |
| TONY | Clase Base de Datos, profesor Sergio Salinas  Actividad realización de un Vídeo hablando acerca del **Modelado de Categorías tipo unión**: considere el tipo propietario, que se subdivide en los conjuntos compañía, banco y persona, que se conecta con una interrelación propietario con cardinalidades M: N con vehículo registrado que puede ser un carro o un camión, fig 8.8 Elmasry & Navathe. Fig 9.7.  A cargo del equipo conformado por Torres Carrillo Josefh Miguel Angel y su servidor Mora Ayala José Antonio. | Introducción |
| MIGUEL | Para abarcar el tema de forma correcta es necesario tener un conocimiento previo acerca de conceptos básicos que se manejan en las Bases de Datos es por eso por lo que comenzaremos con la definición de ¿Qué es una base de Datos? |  |
| TONY | Una base de datos es una colección organizada de información estructurada, o datos, almacenados electrónicamente en un sistema de computadora. Es usualmente controlada por un sistema de gestión de base de datos (DBMS), para nuestro caso estaremos utilizando Oracle  Este está ligado principalmente a un Lenguaje de Consulta Estructurado conocido como SQL | **¿Qué es una base de Datos?** |
| MIGUEL | Pero ahora surge una nueva pregunta ¿Qué es SQL?, bien, como se menciona SQL o mejor conocido por sus siglas en ingles Structured Query Language es un lenguaje de consulta estructurado es decir es un lenguaje de programación que ayuda a solucionar problemas relacionados con la definición, manipulación e integridad de la información representada por los datos almacenados en un Base de datos como le llamaremos a partir de ahora “BD”, | **¿Qué es SQL?** |
| TONY | Una base de datos relacional almacena datos en tablas, estas **contienen datos sobre una sola materia (un algo, objeto)** cada tabla tiene un único nombre que define el contenido de sus datos  **COLUMNAS**  Las columnas en una tabla organizan los datos y **una tabla contiene al menos una columna**, cada columna **representa un único detalle acerca de un set particular de datos.**  **TUPLAS O FILAS**  Cada tupla usualmente **representa un único ser de datos dentro de la tabla** | **Tabla, Fila, Columna** |
| MIGUEL | Como ya hemos visto el concepto de las Tablas y su relación dentro de las Bases de Datos podemos hablar del Modelo Entidad Relación  El modelo entidad-relación o “ER” es una herramienta para generar el modelo de datos que describe la estructura y relaciones de una BD  Básicamente es un diagrama que ayuda a generar la estructura de datos con la que gestionar un problema o actividad real, una vez pasado el diagrama a una BD mediante SQL es posible tanto mantener el funcionamiento de la actividad alimentando la base de datos | **Modelo Entidad Relación** |
| TONY | Cabe destacar que el modelo toma en cuenta las entidades o es decir los objetos de la vida real, estas pueden ser fuertes o débiles, las fuertes son llamadas en ocasiones entidades “maestro” que de forma independiente identifican sus registros con una clave propia por otro lado las débiles dependen de una entidad fuerte para identificar sus registros.  En un modelo "ER” encontraremos esencialmente relaciones de dos tipos: de 1 a N y, de N a M es decir de una entidad a n entidades o de n entidades a m entidades |  |
| MIGUEL | También debemos tener en cuenta las cardinalidades de una relación de una entidad débil con su maestro o entidad fuerte que por lo general o casi siempre será de 1 a N. ya que es como si fueran nodos, un nodo padre (entidad fuerte) puede tener M entidades hijas (entidades débiles) | **Imagen numero 13** |
| TONY | Ya que conocemos el Esquema que vamos a utilizar es importante tomar en cuenta que para el correcto funcionamiento de este son necesarias algunas “restricciones” que deben ser especificadas al momento de crear una tabla, tal como las “llaves primarias” las cuales son: Aquellas que nos ayudan a identificar de manera única los datos dentro de la tabla, el dato de la columna es único para cada tupla.  Pero, aquí surge una interrogante, ¿Cómo es que mediante estas podemos relacionar nuestras tablas? Si la llave primaria como tal solo sirve para dicha tabla donde fue especificada, pues para eso existen las llaves foráneas | **Llave Primaria** |
| MIGUEL | Así es, las FOREIGN KEY o llaves foráneas a veces llamadas claves externas son las que sirven para señalar cual es la llave primaria de otra tabla  Las entidades con este tipo de llave, solo podrán tener valores que ya existan en otra tabla y que también tengan una llave primaria asociada | **Llave Foránea** |
| TONY | Llamamos llave candidata de una relación (o simplemente llave) al atributo o conjunto de atributos que tienen la propiedad de identificar unívocamente a una tupla dentro de la relación. | **Llave Sustituta/Candidata** |
| MIGUEL | También es importante conocer que son las clases y sus respectivas formas, una clase UML se usa para modelar cualquier cosa en la tabla que se va a representar en nuestra BD, esta puede ser una cosa física, un hecho sobre la tabla o incluso un evento | **Clase** |
| TONY | Una superclase es aquella de la cual pueden derivar distintas subclases, las cuales heredan sus atributos, pues depende de qué tipo sea, será como esta subclase será tratada en relación a la superclase | **Superclase** |
| MIGUEL | El conjunto de subclases se define basándonos en características diferenciadoras de las ocurrencias de entidad de la superclase.  Por ejemplo, una Superclase Empleado tiene subclases como secretaria, Ingeniero o Técnico, un empleado puede ser cualquiera de estas 3 subclases | **Subclase** |
| TONY | Una relación es una asociación o correspondencia existente entre una o varias entidades. La relación puede ser regular, si asocia tipos de entidad regulares, o débil, si asocia un tipo de entidad débil con un tipo de entidad regular. Dentro de las relaciones débiles se distinguen la dependencia en existencia y la dependencia en identificación.  Una relación se caracteriza por:   * Nombre: que lo distingue unívocamente del resto de relaciones del modelo. * Tipo de correspondencia: es el número máximo de ocurrencias de cada tipo de entidad que pueden intervenir en una ocurrencia de la relación que se está́ tratando. | **Relación** |
| MIGUEL | Dentro de las relaciones existen las cardinalidades las cuales representan la participación en la relación de cada una de las entidades afectadas, es decir, el número máximo y mínimo de ocurrencias de un tipo de entidad que pueden estar interrelacionadas con una ocurrencia de otro tipo de entidad. La cardinalidad máxima coincide con el tipo de correspondencia. | **Cardinalidad** |
| TONY | a. Relaciones 1:1: Cada ocurrencia de una entidad se relaciona con una y solo una ocurrencia de la otra entidad. |  |
| MIGUEL | b. Relaciones 1:N: Cada ocurrencia de una entidad puede estar relacionada con cero, una o varias ocurrencias de la otra entidad. |  |
| TONY | c. Relaciones M:N: Cada ocurrencia de una entidad puede estar relacionada con cero, una o varias ocurrencias de la otra entidad y cada ocurrencia de la otra entidad puede corresponder a cero, una o varias ocurrencias de la primera. |  |
| TONY | La siguiente figura extraída del libro de Fundamentals Database Systems de Elmsari Navathe representa un esquema que procederé a explicar:  En el esquema podemos observar la existencia de 3 tipos de entidades, las cuales se encuentran bajo el nombre de **PERSON, BANK y COMPANY.** Ahora en la base de datos para el registro de un vehículo (REGISTERED\_VEHICLE), el propietario de un vehículo puede ser de **cualquier tipo de las entidades anteriormente mencionadas.** Para lo cual será necesario crear una Clase (colección de Entidades) la cual incluye las 3 entidades que pueden desarrollar el papel de **propietario de vehículo.** Por lo que crearemos una Categoría de tipo unión llamada **OWNER** la cual es una **subclase de la UNION** de **3 entidades (COMPANY, BANK y PERSON).** Como podemos observar en nuestro diagrama estas 3 superclases están conectadas mediante una línea que llega a un círculo con la letra U en el medio (la cual representa la operación unión).  Nuestro diagrama posee 2 categorías OWNER la cual es una subclase de la unión de PERSON, BANK y COMPANY y también la denominada **REGISTERED\_VEHICLE,** la cual es una subclase que esta constituida por la unión de (**CAR y TRUCK**)  Tenemos una restricción muy importante: OWNER forzosamente debe ser 1 de las 3 entidades anteriormente mencionadas  Cabe destacar que OWNER hereda atributos de PERSON, BANK y COMPANY dependiendo de la superclase a la cual la entidad pertenece.  Podemos observar 2 casos bastante interesantes en este diagrama, las superclases de una categoría pueden poseer diferentes atributos llave, tal como vemos en la categoría OWNER, o bien compartir el atributo llave como se observa en el caso de REGISTERED\_VEHICLE  El diagrama nos presenta las categorías de forma parcial, esto lo sabemos por que esta representado con una sola línea recta, lo cual significa que no todos los carros o camiones involucrados en la base de datos tienen que estar registrados, sucede lo mismo en el caso de OWNER | Fig 8.8 |
| TONY | A continuación, se está presentando la ejecución de las creaciones de las tablas ocupadas para esta base de datos, proporcionamos un formato a la consola para una correcta visualización en las tablas estamos indicando de que atributos estará compuesta cada una de las entidades.  Veamos ahora la descripción de cada una de las tablas para ver que tipo de datos pueden contener.  Ahora procedemos a declarar las restricciones, ósea declarar nuestras llaves primarias y las foráneas para poder relacionar nuestras tablas.  Realizamos la población de cada tabla, mediante **insert into nombre de la tabla y los datos que va a contener.**  Por ultimo visualicemos que el contenido haya sido agregado de forma exitosa | Explicación SQL |
| MIGUEL | La siguiente figura del libro de Fundamentals Database Systems de Elmsari Navathe representa el mapeo de las categorías tipo unión en la figura 8.8  El diagrama consiste en la representación de las tablas unidas por las llaves que estas entidades poseen, Estas llaves van unidas a nuestra entidad **OWNER** la cual vincula directamente a las entidades **PERSON, BANK, COMPANY Y OWNS**, a su vez la entidad **OWNS** también tiene como llave primaria a **VEHICLE\_ID** la cual es referenciada en las entidades faltantes siendo **CAR** y **TRUK** completando así la unión entre nuestras entidades de esta manera tenemos a nuestras entidades conectadas entre sí mediante estos atributos denominados Surrogated Keys o llaves sustituta  El uso de estas llaves es principalmente unificar e identificar todas la entidades en la categoría tipo unión es decir nuestra llave primaria **OWNER\_ID** en la tabla relación **OWNER** es la llave sustituta en **PERSON, BANK Y COMPANY**, mientras que en nuestra otra entidad de relación **REGISTED\_VEHICLE** tenemos la llave primaria **VEHICLE\_ID,** siendo la llave sustituta de **CAR Y TRUK** además estas llaves primarias de las relaciones están ligadas a nuestra superclase **OWNS**, hay que destacar que si un miembro de las tablas donde exista una llave sustituta no pertenece o no es miembro de **OWNER** tendrá un valor NULL para su **OWNER\_ID** en dicha tabla, también es favorable añadir un atributo particular en la tabla **OWNER** que indique a la entidad en particular a la que pertenece dicha id ya sea **PERSON, BANK, COMPANY** | Fig 9.7 |
| MIGUEL | Ahora procederemos a realizar unas consultas a la base de datos previamente creada, una vez ya declaradas las llaves primarias y las tablas ya estando relacionadas mediante las llaves foráneas correspondientes.  Primero obtendremos los datos del vehículo con una identificación de vehículo sea igual al numero 5001, mediante esta consulta obtendremos el (id del vehículo, id de la persona, nombre, numero de licencia y la marca, esto es posible debido a la relación que existe entre ellas)  Ahora accedamos a los vehículos registrados que sean carros, o ahora los que sean un camión.  Para concluir, despleguemos aquellas personas que posean un carro |  |
| TONY | Un diagrama de entidad-relación (ERD por sus siglas en inglés) es parte del proceso de diseño de una base de datos relacional. Un analista comienza por recolectar todos los tipos de datos que encuentra en una organización y los representa como cajas en un diagrama. Las cajas se relacionan entre sí mediante enlaces, representados por una línea, que puede terminar en una pata de gallo, a fin de indicar una relación de una con muchas.  La base de datos relacional se crea con el ERD. Las entidades se convierten en tablas y los atributos son columnas de esas tablas. Las claves primarias se convierten en índices de la tabla y las externas le indican al programador cómo debe enlazar dos tablas. El ERD que diseñó la base de datos se mantiene como referencia para utilizar como un mapa que le indique cómo llegar de una tabla a la otra.  Es por eso por lo que se debe tener una planificación anticipada, pues como vemos al momento de realizar la base de datos ya se debe tener un mapeo de cómo se van a ir relacionando las tablas y mediante que “conectores / llaves”, esto con el propósito de tener todos los datos como debe ser y poder acceder a ellos cuando esto sea necesario, así como evitar la redundancia en los datos o la generación de las llamadas tuplas “espuria” | Importancia de Realizar Esquema de Forma Correcta |
| MIGUEL | Por último, cabe destacar que la importancia de este tema radica principalmente en que la cardinalidad es un aspecto fundamental, pues este nos indicará de acuerdo con las reglas de la normalización si es correcto o bien necesario la generación de una nueva tabla dentro de nuestro esquema o la forma en que dos tablas serán relacionadas, así como el atributo que cada una heredaría o sería establecido como llave primaria o foránea | Importancia de las Cardinalidades |
|  | Con esto damos cierre al tema abordado en este vídeo, esperando haya sido de su agrado y los conceptos hayan sido lo suficientemente claros para comprender mejor las Bases de Datos, así como las figuras usadas para la explicación de dichos conceptos. | Conclusión |