



Clase 02. SQL

Introducción a las Bases de datos relacionales

***RECUERDA PONER A GRABAR LA
CLASE***





OBJETIVOS DE LA CLASE

- Definir base de datos relacional y sus aspectos generales
- Identificar los componentes de una base de datos relacional

CRONOGRAMA DEL CURSO

Clase 1



Introducción a BD



KICK OFF



¡PRESENTATE!

Clase 2



Introducción a las Bases de Datos Relacionales



BASES DE DATOS RELACIONALES, CONCEPTOS



DISEÑO DE UN DIAGRAMA ENTIDAD-RELACIÓN



TEMÁTICAS POSIBLES PARA EL PROYECTO FINAL

Clase 3



Lenguaje SQL



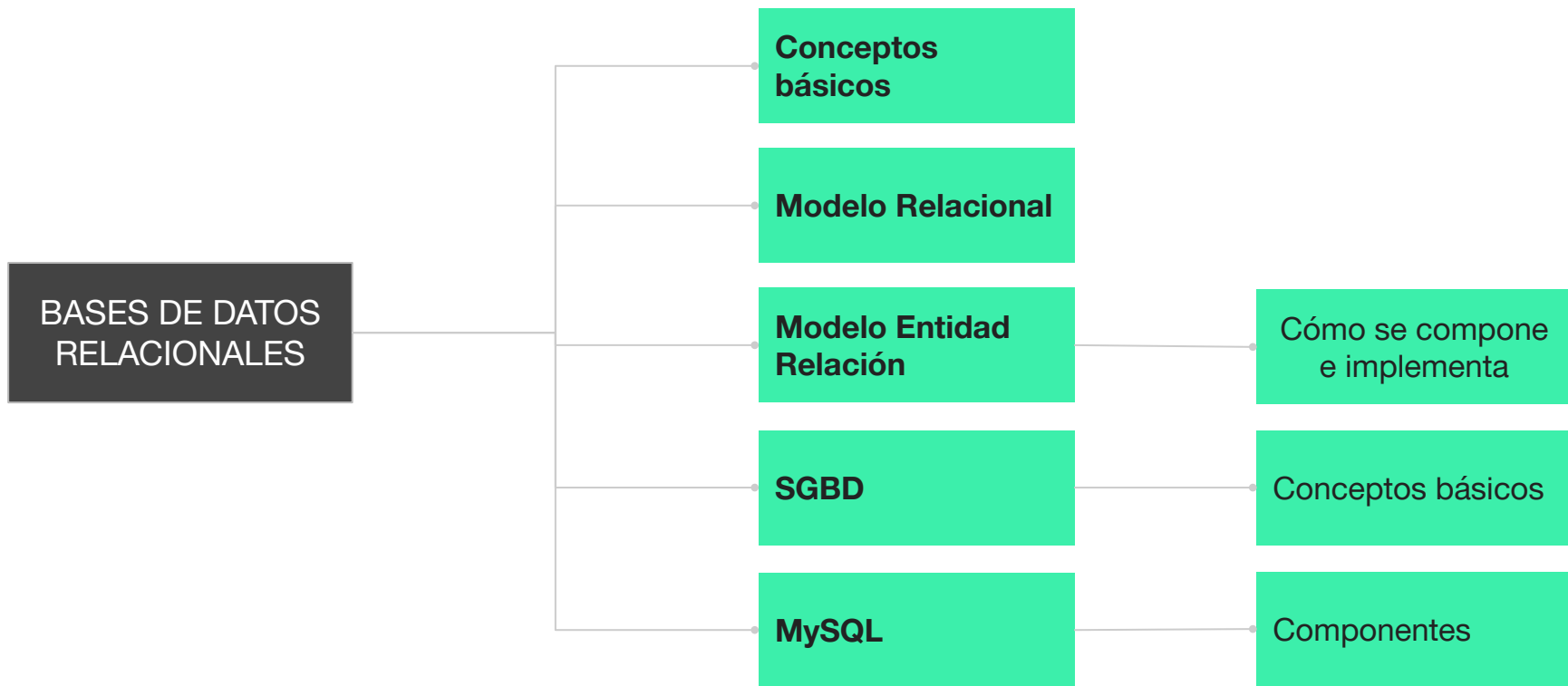
SELECT - FROM



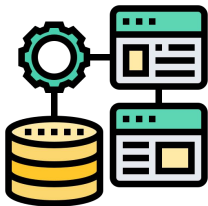
PRÁCTICAS CON LA SECUENCIA WHERE

MAPA DE CONCEPTOS CLASE 2

¡Para
recordar!



BASES DE DATOS RELACIONALES



CONCEPTO

Las bases de datos relacionales se basan en el **Modelo Relacional** usando **tablas** para representar los datos y las relaciones entre ellos.

Es poco frecuente encontrar casos de una DB con una sola tabla pero, en el caso de que se dé esta situación, se la denomina **DB plana**.

BASE DE DATOS PLANA



PROVEEDOR	FECHA	PEDIDO	IMPORTE PAGADO
MAYORISTA DE KIOSCOS S.A.	21/3/2021	1,234	1456,44
BEBIDAS DON PEDRO S.L.	30/3/2021	2,233	2451,87
MAYORISTA DE KIOSCOS S.A.	4/4/2021	1,299	2021,59
MAYORISTA DE KIOSCOS S.A.	10/4/2021	1,344	2163,71
BEBIDAS DON PEDRO S.L.	21/4/2021	2,398	3122,81
MAYORISTA DE KIOSCOS S.A.	22/4/2021	1,400	2022,77

Este es un ejemplo de cómo se suele estructurar la información en una base de datos plana

Si pensamos en un proyecto a gran escala, la redundancia de los datos que encontramos en la columna **PROVEEDOR**, generaría con el tiempo que la DB crezca en ocupación de disco, y se vuelva lenta para buscar información o para recorrerla.

BASE DE DATOS RELACIONAL



COMPRAS			
PROVEEDOR	FECHA	PEDIDO	IMPORTE PAGADO
1	21/3/2021	1,234	1456,44
2	30/3/2021	2,233	2451,87
1	4/4/2021	1,299	2021,59
1	10/4/2021	1,344	2163,71
2	21/4/2021	2,398	3122,81
1	22/4/2021	1,400	2022,77
3	30/4/2021	8,422	3002,41

PROVEEDORES	
CODIGO	NOMBRE
1	MAYORISTA DE KIOSCOS S.A.
2	BEBIDAS DON PEDRO S.L.
3	MINUTAS Y POSTRES S.A.
4	GASEOSAS Y AGUAS S.R.L.

Ejemplo de datos
correctamente
normalizados.

Para transformar la base de datos plana en una base de datos relacional, esta sería una forma mucho más correcta de “normalizar” la información que representamos.



Ventajas



Evitar datos duplicados



Optimiza el espacio de almacenamiento



Reduce la complejidad del usos de las tablas



INVOICE

LOGO

East Repair Inc.
1912 Harvest Lane
New York, NY 12210

BILL TO
John Smith
2 Court Square
New York, NY 12210

SHIP TO
John Smith
3787 Pineview Drive
Cambridge, MA 12210

INVOICE # US-001
INVOICE DATE 11/02/2019
P.O.# 2312/2019
DUE DATE 26/02/2019

QTY	DESCRIPTION	UNIT PRICE	AMOUNT
1	Front and rear brake cables	100.00	100.00
2	New set of pedal arms	15.00	30.00
3	Labor 3hrs	5.00	15.00
Subtotal			145.00
Sales Tax 6.25%			9.06
TOTAL			\$154.06

Basándonos en el ejemplo de **Compras** - **Proveedores**, ¿te puedes imaginar rápidamente cuántos datos que conforman una factura necesitan normalizarse o separarse?

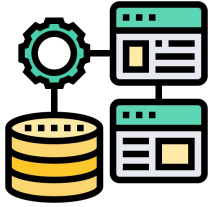
MODELO RELACIONAL



SOBRE LO MENCIONADO HASTA AQUÍ...

Recapitulando sobre lo repasado en la primera clase y lo mencionado en las diapositivas anteriores, habremos notado que **tabla**, es un término frecuentemente nombrado

Profundicemos un poco más este y otros términos de uso frecuente en el universo de las DB.



CONCEPTO

El **Modelo Relacional**, en el cual se apoyan las **Bases de Datos Relacionales** almacenan la información en un **conjunto de tablas**, y a su vez, las aprovechan para representar tanto los **datos** como también las **relaciones** entre cada una de ellas

TABLAS

CODER HOUSE

TABLA

system_user				
id_system_user	first_name	last_name	email	password
1	Tyson	Besse	tbesse0@aboutads.info	VA3hLGlp
2	Tam	Fransewich	tfransewich1@who.int	raX9Jch
3	Rosamund	Barcroft	rbarcroft2@furl.net	YLskgD
4	Bent	Husher	bhusher3@cbc.ca	k4I22qmvY
5	Averell	Alliot	aalliot4@opensource.org	0DtH1nK1G
6	Aurora	Hannabus	ahannabus5@uiuc.edu	U0H0CPUNLv6
7	Somerset	Fairlamb	sfairlamb6@webnode.com	gmsBTHSScu
8	Victor	Maughan	vmaughan7@bloglovin.com	AJOQWDFbrW

Nos referimos a un **tipo de elemento** que permite guardar la información en **registros** y cumple con el paradigma de **modelo relacional**.

FILAS Y COLUMNAS

Esquema de la
tabla o cabecera

Nombre de tabla

Columna o
atributo

system_user				
id_system_user	first_name	last_name	email	password
1	Tyson	Besse	tbesse0@aboutads.info	vA3nLGip
2	Tam	Fransewich	tfransewich1@who.int	raX9Jch
3	Rosamund	Barcroft	rbarcroft2@furl.net	YLskgD
4	Bent	Husher	bhusher3@cbc.ca	k4l22qmvv
5	Averell	Alliot	aalliot4@opensource.org	0DtH1nK1G
6	Aurora	Hannabus	ahannabus5@uiuc.edu	U0H0CPUNLv6
7	Somerset	Fairlamb	sfairlamb6@webnode.com	gmsBTHSScu
8	Victor	Maughan	vmaughan7@bloglovin.com	AJOQWDFbrW

FILAS Y COLUMNAS

Filas

La primer fila de la tabla corresponde al nombre de los campos que la integran y se denomina esquema de tabla o cabecera. El resto de las filas de la tabla son los registros de datos

Columna o atributo

Corresponde al nombre del campo y a todos los datos que almacena para cada registro en dicha posición

REGISTRO

system_user				
id_system_user	first_name	last_name	email	password
1	Tyson	Besse	tbesse0@aboutads.info	VA3hLGlp
2	Tam	Fransewich	tfransewich1@who.int	raX9Jch
3	Rosamund	Barcroft	rbarcroft2@furl.net	YLskgD
4	Bent	Husher	bhusher3@cbc.ca	k4I22qmw
5	Averell	Alliot	aalliot4@opensource.org	0DtH1nK1G
6	Aurora	Hannabus	ahannabus5@uiuc.edu	00H0CP0NLv6
7	Somerset	Fairlamb	sfairlamb6@webnode.com	gmsBTHSScu
8	Victor	Maughan	vmaughan7@bloglovin.com	AJOQWDFbrW

Registro o tupla

Valor almacenado en el campo password del 5to registro de la tabla

Un **registro** es el dato alojado en forma de fila, dentro de una tabla

ESTRUCTURA DE TABLAS



TABLA, RELACIÓN, RELACIONALES...

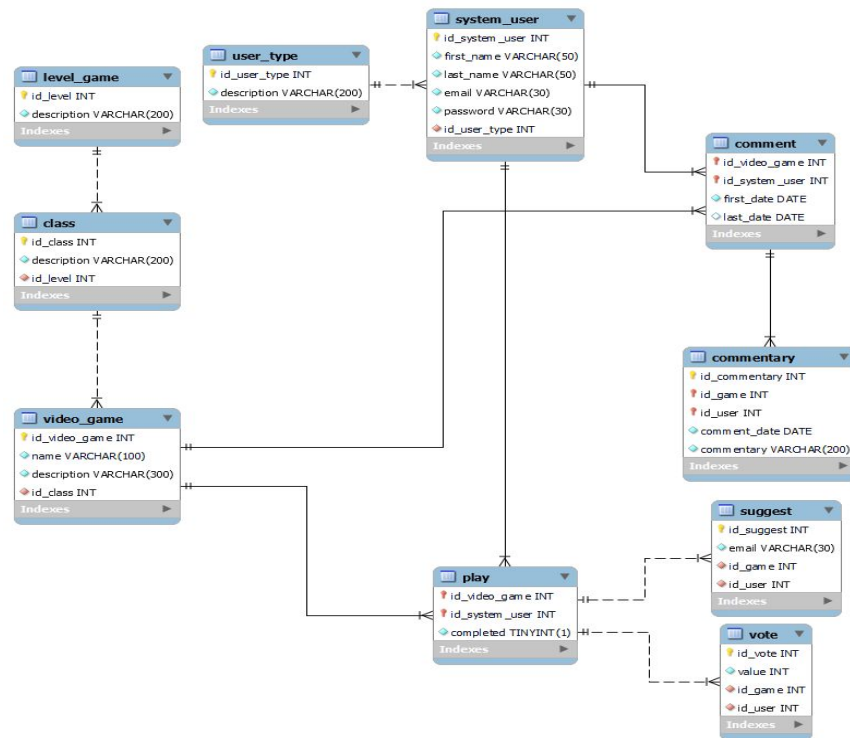
A partir de aquí es cuando comienza a tener más sentido los términos **Relación** y **Relacionales** que hemos venido mencionando hasta ahora.

Veamos entonces cuál es su propósito...

ESTRUCTURA DE TABLAS

Es una **estructura de tablas**, las cuales a su vez **se relacionan con otras** tablas.

Su principal característica es **no poseer información repetida** de forma **innecesaria**, lo que permite adicionar más información sin llegar a afectar la otra almacenada.



VENTAJAS DEL MODELO RELACIONAL



Ventajas

- 👉 Favorece el proceso de **normalización**, el cual permite eliminar la redundancia **evitando la duplicidad** de información en los registros guardados
- 👉 Permite realizar **consultas** y obtener **reportes** de **forma ágil y rápida** por medio de SQL u otro lenguaje de base de datos estructurado



Ventajas



Se pueden crear **una o varias** relaciones entre las tablas

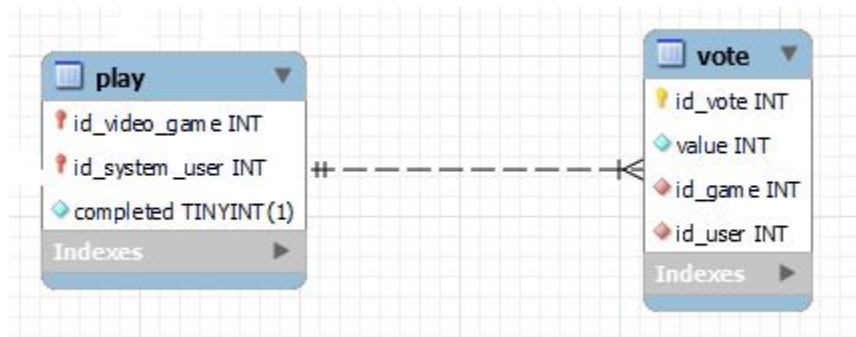


Garantiza la **integridad referencial**; si un **registro se relaciona con otro registro de otra tabla, no permite que** el mismo **sea eliminado**.

Asimismo, si se quiere borrar, también pasará con todos los datos relacionados.

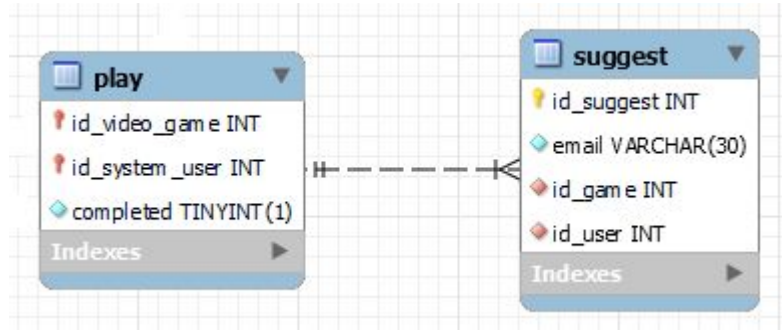
TIPOS DE RELACIONES

UNO A UNO



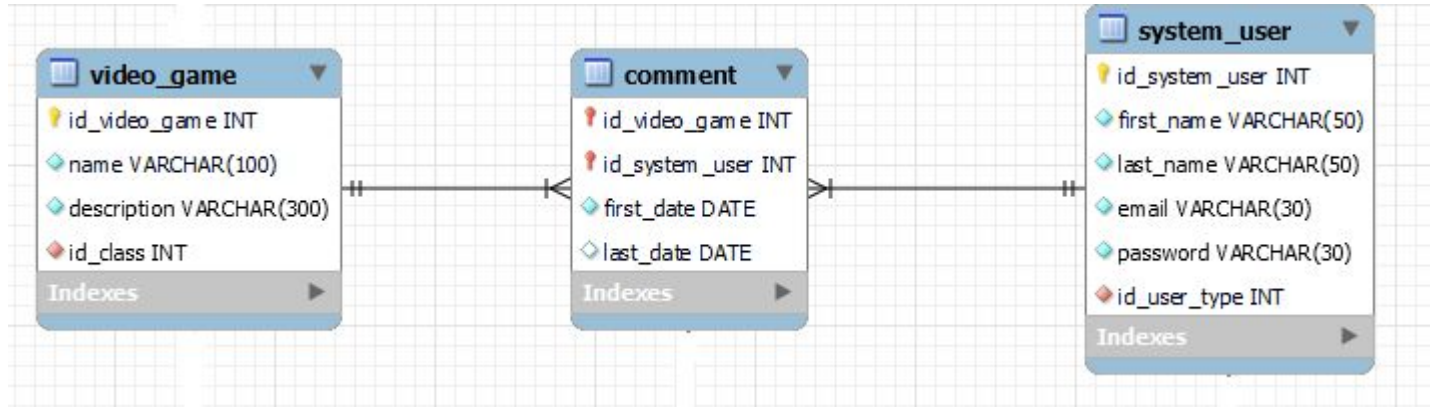
Cada registro, en cada tabla, aparece solo una vez, y tienen una relación unívoca

UNO A MUCHOS



Un registro en una tabla puede tener relación con varios elementos de otra tabla

MUCHOS A MUCHOS



Uno o más registros en una tabla pueden tener relación con uno o más elementos de otra tabla.

CLAVES EN LAS TABLAS

Lógicas



Clave primaria PK: también llamada llave primaria o primary key, hace que el **registro** sea **unívoco** y obligatoriamente **no nulo**

Lógicas



Clave foránea FK: también llamada foreign key, clave secundaria o clave externa, puede ser -o no- una clave primaria dentro de la tabla. Su característica es que es el punto de enlace con otra tabla donde ésta es primary key.

Lógicas

INDEX

Clave índice: es un campo que **facilita la búsqueda** dentro de una tabla. Generalmente son campos primary key.

Conceptuales



- **Clave concatenada CK:** ayuda a encontrar la singularidad en una tabla **combinando dos campos**, ya que no hay una llave primaria.
- **Claves candidatas:** cuando existe más de una clave primaria dentro de la tabla, por ejemplo legajo y dni.

MODELO ENTIDAD-RELACIÓN

EL MODELO DE DATOS ENTIDAD-RELACIÓN

¿De qué se trata?



El modelo de datos entidad-relación (E-R) se basa en la percepción del mundo real que consiste en un **conjunto de objetos básicos**, denominados **entidades**, y de las **relaciones entre esos objetos**.

¿De qué se trata?

Al llevar este modelo al terreno de las bases de datos comenzamos a ver que cada una de las tablas(*entidades o relaciones*) posee **registros** (*instancias*), y que éstos se diferencian del resto de sus pares por alguna característica particular.



¿De qué se trata?



En el ecosistema de DB cada componente posee una forma de identificarse del resto. Además, la forma en la que se relacionan, se basa en un modelo claro que permite la comprensión de la interacción entre cada parte.

Veamos a continuación cómo identificar estos componentes del resto.

COMPONENTES DE BASES DE DATOS

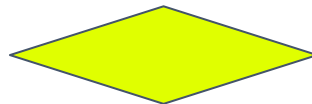
Componentes



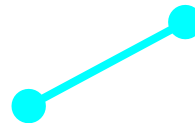
Rectángulos:
representan
conjuntos de
entidades.



Elipses:
representan
atributos; en la
mayoría de los
casos,
la clave
candidata.



Rombos:
representan
acciones que
permiten
relaciones entre
tablas.



Líneas:
representan la
unión entre
acciones y
entidades.

Componentes

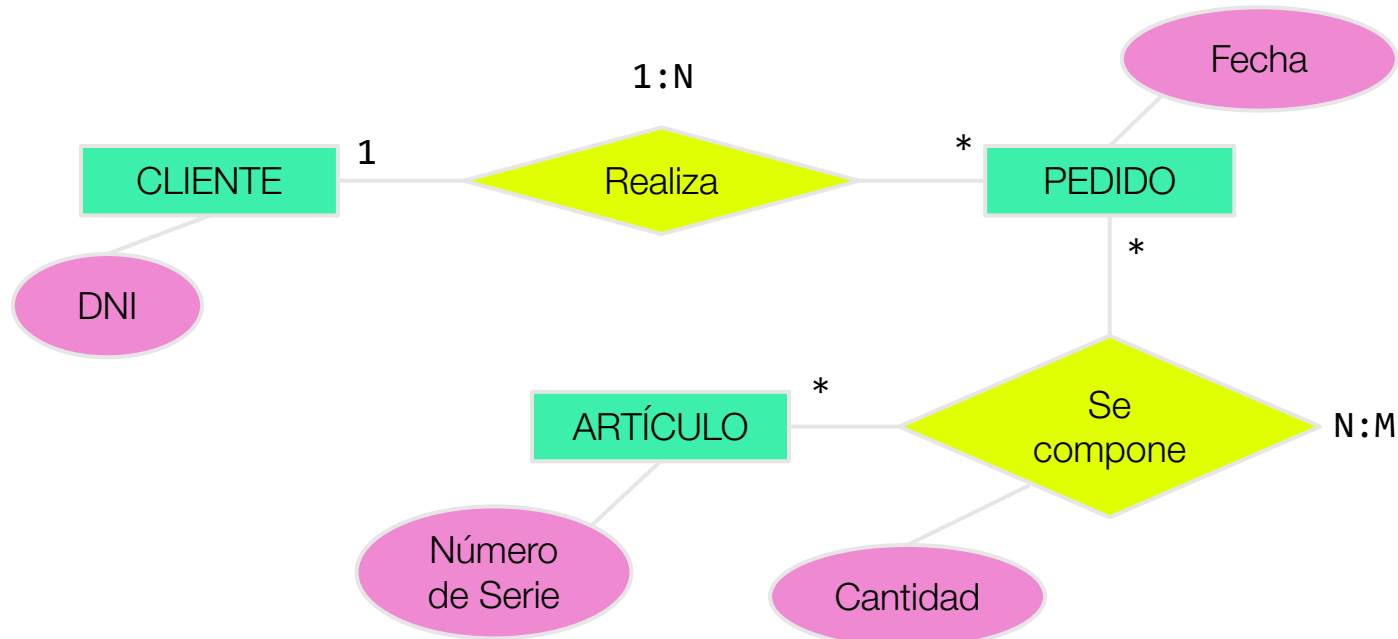


Con estos cuatro componentes, estamos preparados para construir un **Diagrama Entidad-Relación**, que nos permita unir cada elemento de una tabla con uno o más elementos de otras tablas contenidas en la misma base de datos.

Veamos a continuación un ejemplo gráfico de un

Diagrama E-R

Diagrama



Ejemplo
en vivo



¡VAMOS A PRACTICAR LO VISTO!

CODER HOUSE

Diagrama E-R

Ejemplo
en vivo



Estudiantes

Profesores

Asignaturas

- Implementar todos los componentes del diagrama
- Identificar los tipos de relación que se dan

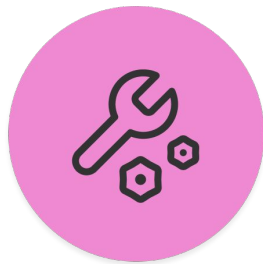


DIAGRAMA DE ENTIDAD-RELACIÓN

Diseña un diagrama de entidad-relación basado en algún modelo de negocios específico.

Tiempo estimado: 20 minutos

DIAGRAMA DE ENTIDAD-RELACIÓN

Tiempo estimado: 20 minutos

Desafío
generico



1. Elegir un modelo de negocios de los presentes en la opción o *download tool* de la web: **<https://businessmakeover.eu/tools/business-model-cards>**
2. Basado en el modelo de negocios elegido, identifica cinco posibles tablas que formarían parte del modelo de datos, y diseña el diagrama E-R con sus componentes.
3. Guarda el archivo del desafío, porque lo usaremos en una próxima clase como complemento

NOTA: Sugerimos diseñes la consigna en formato PPT o GSlides.

CODER HOUSE



BREAK

¡5/10 MINUTOS Y VOLVEMOS!

**¿Sabías que premiamos a nuestros
estudiantes por su dedicación
durante la cursada?**

Conocé los beneficios del TOP10

¡5/10 MINUTOS Y VOLVEMOS!

CODER HOUSE

MYSQL

CODER HOUSE

MySQL



Es un **sistema de gestión de Base de datos Relacional.**

ORACLE es su dueño actual, pero mantiene una **licencia dual**, que le permite tener una opción **bajo el estándar OPEN SOURCE** (*Código Abierto*).



www.mysql.com

CODER HOUSE



Mysql



Fue concebida para usarse bajo el **modelo cliente-servidor**. Debe instalarse en un servidor para que los clientes se conecten y operen con la información de la base de datos.

Soporta múltiples bases de datos bajo un único motor y, para una respuesta efectiva, dependerá del hardware de procesamiento del servidor donde se ejecute.



Componentes Mysql



- mysqld
- mysql
- mysqlimport
- mysqladmin
- mysqldump
- mysqlexport
- mysqlcheck
- myisamchk
- myisampack



Instalación



Guia de instalación de **MySQL**:

👉 Instalación para [Windows](#)

👉 Instalación para [MAC](#)



MYSQL WORKBENCH

Conozcamos este sistema de gestión de base de datos, y sus principales herramientas

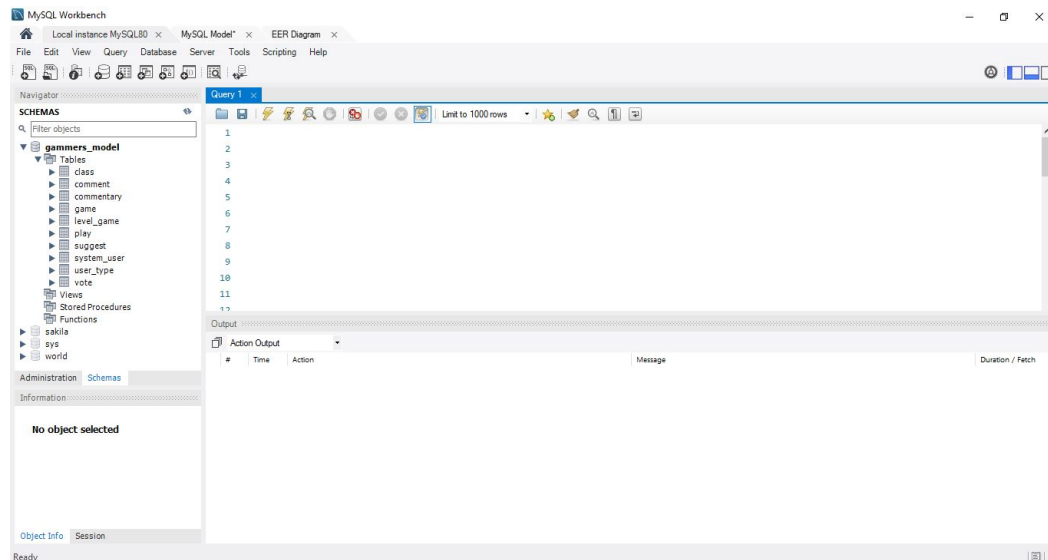
Mysql Workbench



Es una herramienta visual que permite crear y gestionar bases de datos de forma gráfica.

Este software integra:

- Diseño de las DB
- Desarrollo de las DB
- Administración de las DB
- Mantenimiento de las DB



Panel Schemas



El **Panel Schemas** nos lista todas las bases de datos que tengamos, junto a sus objetos.

Podemos navegar cada objeto a través del **menú de árbol**, para ver sus propiedades y acciones.

Al pié, obtenemos **información general del elemento** seleccionado.

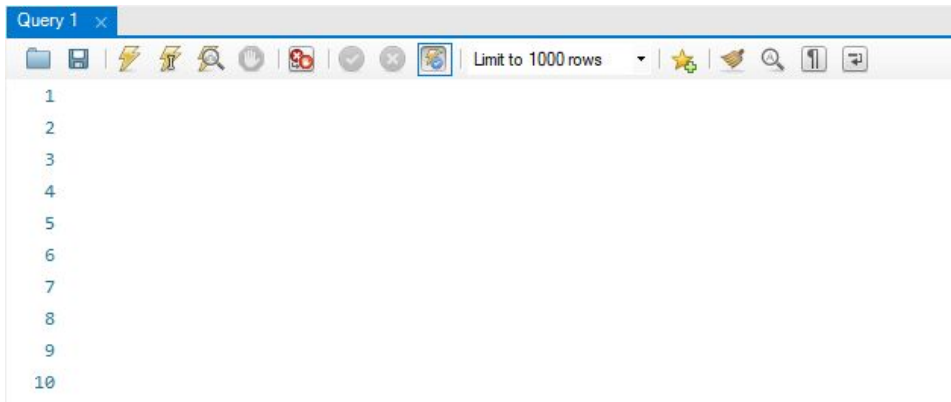
Barra de herramientas superior



Incluye botones para iniciar cualquiera de las principales tareas tales como crear una base de datos, abrir una ventana de script, crear una tabla, una función, un stored procedures y otras opciones más.

Al pié se encuentran **las pestañas de acción** para navegar entre los objetos en uso.

Panel central



Permite visualizar diferentes pestañas para ver o escribir comandos SQL, junto a la ventana de resultados al pie de este panel

Pestaña de acciones

Output				
Action Output				
#	Time	Action	Message	Duration / Fetch
✖	1 22:01:58	SELECT * FROM VIDEO_GAME WHERE id_level IN (SELECT id_level FROM level_...	Error Code: 1146. Table 'gammers_model.video_game' doesn't exist	0.078 sec
✔	2 22:02:04	SELECT * FROM GAME WHERE id_level IN (SELECT id_level FROM level_game ...	100 row(s) returned	0.078 sec / 0.000 sec

Al pie de la ventana está la **pestaña de acciones**, donde veremos el resultado de cada consulta o comando SQL que se dispare; la confirmación de ejecución, los errores, y demás mensajes del sistema.



TEMÁTICAS POSIBLES PARA EL PROYECTO FINAL

Generando un listado de posibles temáticas a abordar

TEMÁTICAS POSIBLES PARA EL PROYECTO FINAL

Formato: PPT o Slides nombrado como “Lista+Apellido”

Sugerencia: En caso de ser un archivo en línea, activar permisos de acceso.

Desafío
entregable



>> Consigna:

- Pensar una serie de temáticas de cara a ir moldeando tu proyecto final y generar un listado de posibles temáticas a abordar para tu base de datos.

>> Aspectos a contemplar:

- Para la **clase 04** comenzaremos a diseñar el **Diagrama Entidad-Relación** de nuestro proyecto final.
- Si bien el mismo puede variar, al menos ya podemos construir las bases para que luego sean modificadas según lo que vayamos agregando en éste.
- Los formatos de entrega para estas presentaciones serán PPT o Slides de Google. También es posible exportar este contenido al formato PDF, según lo que más cómodo sea para los estudiantes.

CODER HOUSE



RESUMEN DEL GLOSARIO TÉCNICO

- Base de datos relacional
- Base de datos plana
- Modelo relacional
- Diagrama Entidad-Relación
- SGBD
- Mysql
- Mysql Workbench
- Clave primaria
- Clave foránea
- Índice único
- Relación uno a uno
- Relación uno a muchos
- Relación muchos a muchos
- Tablas
- Filas y columnas
- Registros

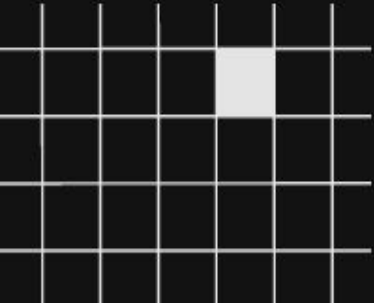
¿PREGUNTAS?





¡MUCHAS GRACIAS!

Resumen de lo visto en clase hoy:

- Bases de datos relacionales
 - Modelo de entidad-relación
 - Diseño de base de datos
- 



OPINA Y VALORA ESTA CLASE

#DEMOCRATIZANDO LA EDUCACIÓN

CODER HOUSE