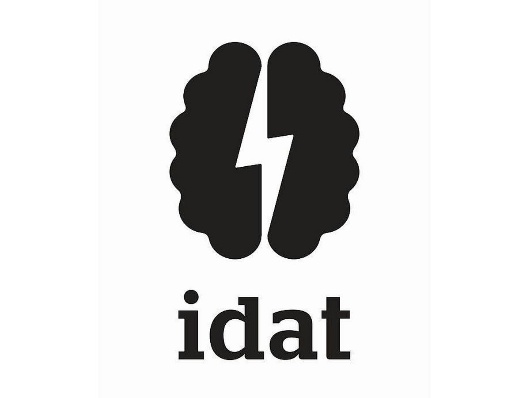
**INSTITUTO SUPERIOR TECNOLÓGICO**

**“IDAT”**



**INFORME N° 001-2025-EJLD/IST “IDAT”- C**

**“BASE DE DATOS RELACIONALES”**

**MÓDULO IV:**

Crea la información de cualquier motor de base a través del lenguaje SQL y NoSQL.

**PROGRAMA DE ESPECIALIZACIÓN:**

DESARROLLO BACK-END

**PRESENTADO POR:**

EMERSON JERZY LEÓN DURÁN

**MARZO – 2025**

# Índice

[Índice 2](#_Toc194168333)

[Introducción 4](#_Toc194168334)

[1. Las 12 reglas de Codd 5](#_Toc194168335)

[1.1. Regla 0: El fundamento: 5](#_Toc194168336)

[1.2. Regla 1: La información: 5](#_Toc194168337)

[1.3. Regla 2: Acceso garantizado: 5](#_Toc194168338)

[1.4. Regla 3: Tratamiento sistemático de valores nulos: 5](#_Toc194168339)

[1.5. Regla 4: Catálogo en línea basado en el diccionario de datos relacional: 6](#_Toc194168340)

[1.6. Regla 5: La regla del sublenguaje de datos completo: 6](#_Toc194168341)

[1.7. Regla 6: La regla de actualización de la vista: 6](#_Toc194168342)

[1.8. Regla 7: Inserción, actualización y eliminación de alto nivel: 6](#_Toc194168343)

[1.9. Regla 8: Independencia física de los datos: 7](#_Toc194168344)

[1.10. Regla 9: Independencia lógica de los datos: 7](#_Toc194168345)

[1.11. Regla 10: Independencia de la integridad: 7](#_Toc194168346)

[1.12. Regla 11: Independencia de la distribución: 7](#_Toc194168347)

[1.13. Regla 12: La regla de la no subversión: 7](#_Toc194168348)

[2. Normalización de base de datos 8](#_Toc194168349)

[2.1. Objetivos principales de la normalización 8](#_Toc194168350)

[2.2. Niveles de normalización 8](#_Toc194168351)

[2.2.1. Primera Forma Normal (1FN) 8](#_Toc194168352)

[2.2.2. Segunda Forma Normal (2FN) 9](#_Toc194168353)

[2.2.3. Tercera Forma Normal (3FN) 9](#_Toc194168354)

[2.3. Requisitos de la normalización 9](#_Toc194168355)

[2.4. ¿Por qué se normalizan las bases de datos? 9](#_Toc194168356)

[3. Modelo relacional para el modelado y gestión de base de datos 10](#_Toc194168357)

[3.1. Características 10](#_Toc194168358)

[3.2. Objetivos 11](#_Toc194168359)

[3.3. Diseño e implementación de un modelo de datos relacional 11](#_Toc194168360)

[3.3.1. Importancia 12](#_Toc194168361)

[3.4. Representación y diagramas 12](#_Toc194168362)

[3.4.1. Diagrama entidad relación 12](#_Toc194168363)

[3.4.2. Diagrama relacional 13](#_Toc194168364)

[4. Línea de tiempo de la evolución de las bases de datos 14](#_Toc194168365)

[5. Conclusión 18](#_Toc194168366)

[6. Referencias 19](#_Toc194168367)

# Introducción

En el mundo de la gestión de bases de datos, el modelo relacional ha sido una de las estructuras más utilizadas debido a su capacidad para organizar y manejar grandes volúmenes de información de manera eficiente. El concepto de bases de datos relacionales fue propuesto por Edgar F. Codd en 1970, quien formuló un conjunto de reglas y principios fundamentales para el diseño y manejo de bases de datos. Estas reglas, conocidas como las *12 reglas de Codd*, establecieron un marco teórico que mejoró la comprensión y aplicación del modelo relacional. Además, el proceso de normalización es esencial para garantizar la eficiencia y la calidad de la base de datos.

Este informe tiene como objetivo detallar las 12 reglas de Codd, la importancia de la normalización de bases de datos y el funcionamiento del modelo relacional en el contexto del diseño y gestión de bases de datos.

# Las 12 reglas de Codd

Edgar Frank Codd es considerado como el padre de las bases de datos relacionales, quién inventó en 1969 el modelo relacional, siendo esta la más utilizada en la actualidad. Por lo tanto, estableció 12 reglas y/o principios que debe cumplir un Sistema de Gestión de Bases de Datos Relacional (SGBDR) para ser considerado verdaderamente relacional. Estas reglas garantizan la integridad, consistencia y flexibilidad de los datos.

## Regla 0: El fundamento:

Para cualquier sistema que se promocione como un sistema de gestión de bases de datos relacional, ese sistema debe ser capaz de gestionar bases de datos enteramente a través de sus capacidades relacionales.

## Regla 1: La información:

Toda la información en una base de datos relacional es representada unidireccionalmente, por valores en posiciones de las columnas dentro de filas de tablas. Por lo que, se representa explícitamente a nivel lógico exactamente de una forma: valores en tablas.

## Regla 2: Acceso garantizado:

Todos los datos deben ser accesibles sin ambigüedad en donde ha de garantizarse que el acceso a cada dato individual (valor atómico) en una base de datos relacional debe ser lógicamente especificando el nombre de la tabla, el valor de la clave primaria y el nombre de la columna.

## Regla 3: Tratamiento sistemático de valores nulos:

Los valores nulos (distintos de la cadena vacía, una cadena de caracteres vacía o cualquier otro valor) se admiten en las bases de datos relacionales para representar información faltante e inaplicable de forma sistemática, independiente del tipo de datos.

## Regla 4: Catálogo en línea basado en el diccionario de datos relacional:

La descripción de la base de datos (es decir, el diccionario de datos o el catálogo) se representa al nivel lógico de la misma forma que los datos normales, de modo que los usuarios autorizados puedan aplicar el mismo lenguaje relacional a sus consultas que utilizan para los datos normales.

## Regla 5: La regla del sublenguaje de datos completo:

Un sistema de bases de datos relacional puede admitir varios lenguajes y varios modos de uso de terminal (por ejemplo, el modo de relleno de formularios). Sin embargo, debe haber al menos un lenguaje cuyas expresiones estén expresadas como cadenas de caracteres, que se ajusten a la regla de la información (como su sintaxis y al menos soporte):

Definición de datos.

Definición de la vista.

Manipulación de datos (interactiva y por programa).

Restricciones de seguridad e integridad.

Administración de transacciones (inicio, confirmación y reversión).

## Regla 6: La regla de actualización de la vista:

Todas las vistas que sean teóricamente actualizables también se pueden actualizar por el sistema.

## Regla 7: Inserción, actualización y eliminación de alto nivel:

Los datos no solo se pueden recuperar de una base de datos relacional de filas múltiples y/o de tablas múltiples, sino también pueden realizarse inserciones, actualización y borrados sobre varias tuplas y/o tablas al mismo tiempo (no sólo sobre registros individuales).

## Regla 8: Independencia física de los datos:

Los cambios en el nivel físico (almacenamiento) no requieren un cambio en un nivel lógico (aplicación).

## Regla 9: Independencia lógica de los datos:

Los cambios en el nivel lógico (tablas, columnas, filas, etc.) no requieren un cambio en un nivel lógico (aplicación). La independencia lógica de los datos es más difícil de lograr que la independencia física de los datos.

## Regla 10: Independencia de la integridad:

Las restricciones de integridad específicas de una base de datos relacional deben definirse en el sublenguaje relacional y almacenarse en el catálogo, no en los programas de aplicación.

## Regla 11: Independencia de la distribución:

La distribución de partes de la base de datos a varios lugares debe ser invisible para los usuarios de la base de datos. Las aplicaciones existentes deben continuar funcionando correctamente:

* *Cuando una versión distribuida del SGBD se introdujo por primera vez.*
* *Cuando se distribuyen los datos existentes se redistribuyen en todo el sistema*.

## Regla 12: La regla de la no subversión:

Si un sistema relacional tiene un lenguaje de bajo nivel (registro único a la vez), ese bajo nivel no puede usarse para subvertir o eludir las reglas y restricciones relacionales de alto nivel (varios registros a la vez).

# Normalización de base de datos

La normalización es el proceso de organizar los datos en una base de datos para reducir la redundancia y mejorar la integridad de los datos. Implica dividir tablas grandes en tablas más pequeñas y vincularlas mediante relaciones.

## Objetivos principales de la normalización

* Corregir datos duplicados y anomalías en la base de datos.
* Evitar la creación y actualización de relaciones y dependencias no deseadas entre datos.
* Prevenir borrados indeseados de datos.
* Optimizar el espacio de almacenamiento.
* Reducir el tiempo y complejidad de revisión de las bases de datos cuando es necesario introducir nuevos tipos de datos.
* Facilitar el acceso e interpretación de los datos a los usuarios y aplicaciones que los usan.

## Niveles de normalización

Para normalizar una base de datos existen principalmente 3 reglas, las cuales se deberían cumplir para evitar redundancias e incoherencias en las dependencias. A estas reglas se les conoce como "Forma normal" qué va de la 1 a la 3 y si la base de datos cumple con cada regla se dice que está en la "primera o segunda o tercera forma normal"

### Primera Forma Normal (1FN)

* Es la forma más básica de normalizar bases de datos.
* Elimina duplicados y crea tablas separadas para grupos de datos relacionados.
* Cada columna en una tabla de base de datos contiene valores atómicos.
* Cada tabla debe tener una clave primaria única que identifique de manera unívoca cada fila.

### Segunda Forma Normal (2FN)

* Elimina subgrupos de datos en múltiples filas de una tabla y crea tablas nuevas.
* Reduce la redundancia al eliminar las dependencias parciales.

### Tercera Forma Normal (3FN)

* Reduce la duplicación de datos al eliminar las dependencias transitivas.
* Elimina columnas que no dependen de la clave principal.

## Requisitos de la normalización

* + Cada tabla debe tener su nombre único.
  + No puede haber dos filas iguales.
  + No se permiten los duplicados.
  + Todos los datos en una columna deben ser del mismo tipo.

## ¿Por qué se normalizan las bases de datos?

Porque de esa manera logramos evitar la redundancia de los datos, disminuir problemas de actualización de datos, proteger la integridad, facilitar el acceso e interpretación de los datos, reducir el tiempo y complejidad de revisión de las bases de datos, optimizar el espacio de almacenamiento y prevenir borrados indeseados de datos.

# Modelo relacional para el modelado y gestión de base de datos

El modelo relacional es un enfoque para la gestión de bases de datos basado en la lógica de predicados de primer orden y en la teoría de conjuntos. Esta se encarga de organiza los datos en tablas (relaciones) con filas (tuplas) y columnas (atributos) en donde estas relaciones entre tablas se establecen mediante claves primarias y claves externas.

Además, este modelo desarrolla un esquema de base de datos (data base schema) a partir del cual se podrá realizar el modelo físico o de implementación en el SGBD. Asimismo, está basado en que todos los datos están almacenados en tablas (entidades/relaciones) y cada una de estas es un conjunto de datos, por tanto, una base de datos es un conjunto de relaciones.

Este modelo se ocupa de la estructura, manipulación e integridad de los datos en donde las relaciones están formadas por atributos (columnas) y tuplas (conjunto de filas).

## Características

* **Tablas:** Los datos se organizan en tablas con filas y columnas.
* **Claves primarias:** Cada fila tiene una clave primaria única que la identifica.
* **Claves externas:** Las claves externas se utilizan para establecer relaciones entre tablas.
* **SQL (Structured Query Language):** Es el lenguaje estándar para interactuar con bases de datos relacionales.

## Objetivos

* **Independencia Física:** La forma de almacenar los datos no debe influir en su manipulación. Si el almacenamiento físico cambia, los usuarios que acceden a esos datos no tienen que modificar sus aplicaciones.
* **Independencia Lógica:** Las aplicaciones que utilizan la base de datos no deben ser modificadas porque se inserten, actualicen y eliminen datos.
* **Flexibilidad:** En el sentido de poder presentar a cada usuario los datos de la forma en que éste prefiera
* **Uniformidad:** Las estructuras lógicas de los datos siempre tienen una única forma conceptual (las tablas), lo que facilita la creación y manipulación de la base de datos por parte de los usuarios.
* **Sencilles:** Las características anteriores hacen que este modelo sea fácil de comprender y de utilizar por parte del usuario final.

## Diseño e implementación de un modelo de datos relacional

El diseño del modelo relacional se puede dividir en tres niveles: el nivel conceptual, lógico y físico. Estos niveles representan diferentes perspectivas del modelo de datos relacional.

A continuación, se detallan los tres niveles para el diseño del modelo relacional:

* **Modelo de datos conceptual**: En este modelo se define quecontiene el sistema**.**El propósito de crear un modelo de datos conceptual es establecer entidades, sus atributos y sus cardinalidades. Este modelo es independiente de cualquier consideración de implementación o detalles.
* **Modelo de datos lógicos**: En este modelo se define cómo se debe implementar el sistema independientemente del DBMS (modelo genérico). El modelo de datos lógico, va un paso más adelante que el modelo de dato conceptual, definiendo tablas, claves primarias, claves foráneas y las restricciones de integridad referencial.
* **Modelo de datos físico**s: En este modelo se define la implementación concreta utilizando un sistema DBMS. El modelo de datos físico adapta el modelo de datos lógico al DBMS, agregando particularidades propias del gestor, como lo pueden ser la nomenclatura, administración de memoria, almacenamiento en disco y otras definiciones.

### Importancia

La separación en niveles del modelo de datos relacional proporciona claridad conceptual, flexibilidad, mantenibilidad, independencia de la plataforma y facilidad de mantenimiento y evolución a largo plazo. Asimismo, permite a los diseñadores y administradores trabajar en diferentes niveles de detalle y enfoque según sus roles y responsabilidades, lo que conduce a una base de datos más robusta y adaptable.

## Representación y diagramas

### Diagrama entidad relación

Este diagrama describe la realidad mediante un conjunto de representaciones gráficas simples en donde todas las abstracciones se representarán basándose en entidades, las cuales estarán asociadas entre sí mediante cardinalidades**.**

A continuación, se mencionan los componentes básicos de un diagrama entidad relación:

* **Entidad**: Una entidad es cualquier objeto o concepto sobre el cual se recoge la información, puede ser una cosa, persona o suceso.
* **Atributos**: Los atributos son las características o propiedades de las entidades. Describen los detalles o información específica asociada a una entidad.
* **Cardinalidad**: La cardinalidad es el número mínimo y número máximo de correspondencias que una relación puede tomar.

### Diagrama relacional

El diagrama relacional, ayuda a comprender la estructura detallada de la base de datos y a visualizar las relaciones y restricciones entre las tablas. Por lo que, es una herramienta útil para la comunicación entre los diseñadores y desarrolladores de la base de datos debido a que proporciona una representación visual clara de la estructura lógica de la base de datos.

A continuación, se mencionan los componentes básicos de un diagrama relacional:

* **Tablas**: Representan las entidades y se identifican por su nombre.
* **Columnas o atributos**: Representan las características o propiedades de las tablas, y se especifican junto a su tipo de dato.
* **Claves primarias**: Se identifican las columnas o combinaciones de columnas que funcionan como claves primarias, subrayándolas o resaltándolas.
* **Claves foráneas**: Se indican las relaciones entre tablas mediante claves foráneas que hacen referencia a las claves primarias de otras tablas.
* **Restricciones y reglas**: Se pueden incluir restricciones de integridad, como restricciones de clave primaria, restricciones de clave foránea y restricciones de valores permitidos.

# Línea de tiempo de la evolución de las bases de datos

**Siglo XIX:**

* **Año de 1884: Tarjetas perforadas**
* Se dio origen a la máquina automática de perforación de tarjetas inventadas por Herman Hollerith. Esta tarjeta perforada era una pieza que contenía la información digital representada mediante la presencia o ausencia de agujeros en posiciones predeterminadas.

**Siglo XX:**

* **Año de 1946: Primer ordenador digital de uso comercial**
* El UNIVAC I diseñado por John William Mauchly y J. Presper Eckert fue la primera computadora comercial de Estados Unidos que, procesaba cintas magnéticas y tenía un convertidor de tarjetas perforadas a cintas magnéticas.
* **Década de 1950: Cintas magnéticas**
  + En el año de 1950 Oberlin Smith Son creó las cintas magnéticas, las cuales eran un tipo de medio o soporte de almacenamiento de datos que se graba en pistas sobre una banda plástica con un material magnetizado. Por lo que, el tipo de información que se podía almacenar en las cintas magnéticas era variado, tales como vídeo, audio y datos.
* **Década de 1960:** **Inicio de las bases de datos computarizadas**
  + Se desarrollan los primeros sistemas de gestión de bases de datos jerárquicos y de red.
  + El uso de computadoras era más accesible al público.
  + Se populariza el uso de los discos (CD), la cual fue un adelanto muy efectivo, ya que por medio de este soporte se podía consultar la información directamente logrando ahorrar tiempo en donde no era necesario saber exactamente donde estaban los datos en los discos debido a que en milisegundos era recuperable la información.
  + Alianza de IBM y American Airlines para desarrollar SABRE, un sistema operativo que manejaba transacciones e información sobre los pasajeros de la compañía.
* **Año de 1961: IDS (Integrated Data Store)**
  + Charles Bachman supuso la creación de un nuevo tipo de sistema de bases de datos conocido como modelo en red que permitió la creación de un standard en los sistemas de bases de datos gracias a la creación de nuevos lenguajes de sistemas de información.
* **Año de 1963: Primera mención**
  + Se da la primera mención del término bases de datos, dada en un simposio celebrada en california, entendiéndolas como un conjunto de información relacionada agrupada o estructurada.
* **Década de 1970: Modelo relacional**
  + Edgar Frank Codd definió el modelo relacional y a la par publicó una serie de reglas para los sistemas de datos relacionales dando paso a la segunda generación de los Sistemas Gestores de Base de Datos en las cuales, surgió el lenguaje SQL.
* **Año de 1976: Modelo entidad-relación**
  + Peter Chen propone un modelo de base de datos nueva llamada Entidad-Relación, o ER. Este modelo permitió a los diseñadores centrarse en la aplicación de los datos, en lugar de estructura de la tabla lógica.
* **Década de 1980:** **Bases de datos Orientadas a Objetos y lenguaje SQL**
  + Los datos e información se representan en forma de objetos.
  + Las estructuras orientadas a objetos como las jerarquías, agregaciones y punteros son introducidas.
  + Permitieron manejar de manera eficiente una gran cantidad de datos de distintos tipos.
  + Los objetos con comportamientos complejos fueron fáciles de manejar utilizando herencia y polimorfismo.
  + Redujeron el número de relaciones al crear objetos.
  + Structured Query Language (SQL) se convirtió en el lenguaje de consulta estándar.
* **Década de 1990: Microsoft Access y Excel** 
  + Los programas Access y Excel de la compañía Microsoft Office marcan el comienzo de las bases de datos orientadas a objetos.
* **Año de 1993: World Wide Web (www)**
  + La compañía CERN puso el software de la World Wide Web en el dominio público en donde gracias a este software se facilitó la consulta a bases de datos.

**Siglo XXI:**

* **Año de 2008: MySQL**
  + Existe la versatilidad de combinar el estándar SQL con la accesibilidad que proporciona el Internet y se desarrolla MySQL.
* **Año de 2010: Sistemas XML**
  + Las bases de datos XML (Extensible Markup Language) forman un subconjunto de bases de datos utilizando el lenguaje XML, que es abierto, legible para humanos y maquinas, ampliamente usado.
* **Actualidad:**
  + En la actualidad, las tres grandes compañías que dominan el mercado de las bases de datos son IBM, Microsoft y Oracle. Por su parte, en el campo de Internet, la compañía que genera gran cantidad de información es Google. Aunque existe una gran variedad de software que permiten crear y manejar bases de datos con gran facilidad.

# Conclusión

El modelo relacional y las 12 reglas de Codd han sido fundamentales para el desarrollo de sistemas de bases de datos eficientes y robustos. La normalización, como parte integral del diseño de bases de datos, asegura que los datos sean almacenados de manera eficiente, eliminando redundancias y mejorando la integridad. A lo largo de los años, el modelo relacional ha evolucionado, pero sigue siendo una de las arquitecturas más utilizadas en la gestión de bases de datos en la actualidad.

Este informe ha presentado una visión detallada de estos tres temas clave, proporcionando una base sólida para entender el impacto de estos avances en el mundo de las bases de datos.

# Referencias

Codd, E. F. (1970). *A relational model of data for large shared data banks*. Communications of the ACM, 13(6), 377–387.

Elmasri, R., & Navathe, S. B. (2016). *Fundamentals of Database Systems* (7th ed.). Pearson Education.

Chen, P. P. (1976). *The Entity-Relationship Model—Toward a Unified View of Data*. ACM Transactions on Database Systems, 1(1), 9–36.