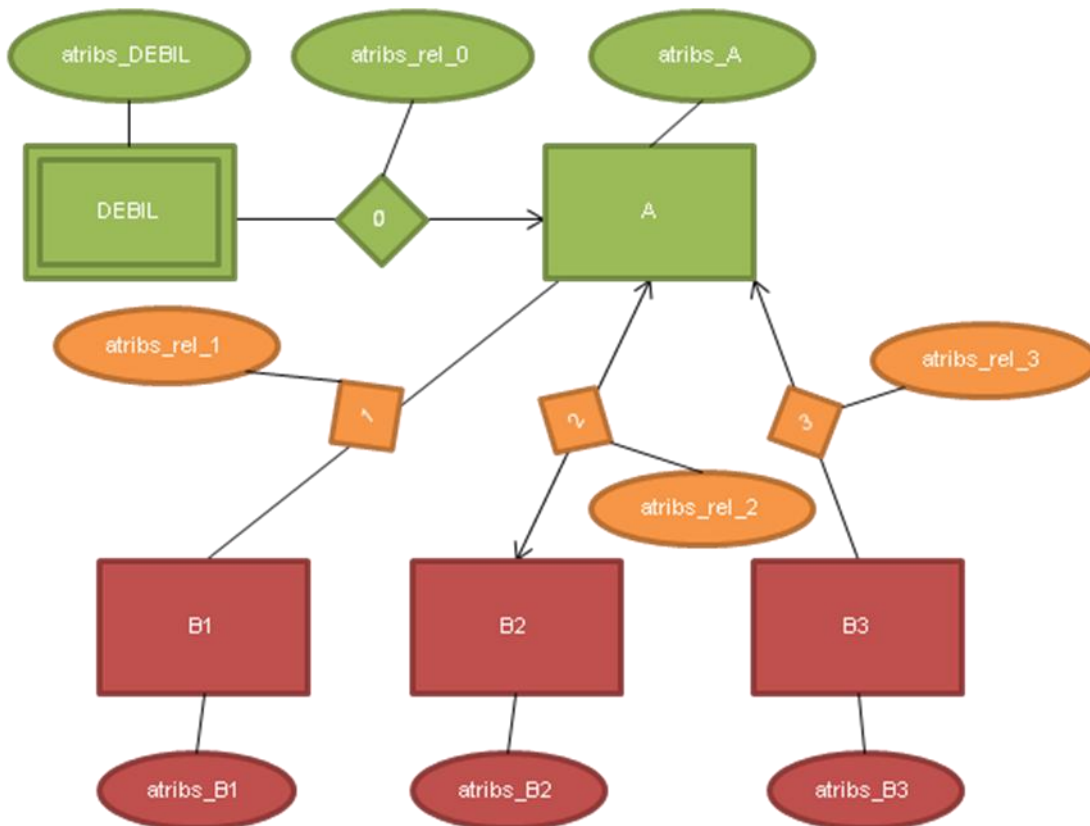




Introducción a la unidad

El modelo relacional de base de datos surge a finales de los 60, no obstante, es hoy en día el modelo más utilizado en sistemas empresariales. Los principales manejadores de bases de datos comerciales o de software libre están basados en este modelo y brindan soluciones tecnológicas robustas para todo tipo de empresas. Es por estas razones que el Licenciado en Informática debe conocer el modelo relacional de base de datos.





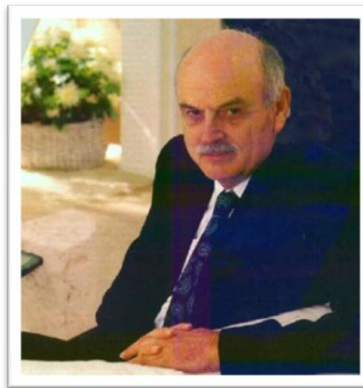
Unidad II. Modelo Relacional



En esta unidad, revisaremos algunas características de los modelos pre-relacionales y pos-relacionales para brindar parámetros de diferenciación con el modelo relacional. También estudiaremos los fundamentos teóricos del modelo a partir de los conceptos de relación, dominio, álgebra y cálculo relacional.

En el proceso de desarrollo de una base de datos relacional, resulta importante evitar problemas de redundancia y de actualización de datos, por esto repasaremos el procedimiento conocido como normalización, el cual se basa en la descomposición sin pérdida de distintas relaciones para ajustarlas a formas normales.

Fue Edgar F. Codd quien puso las bases de este modelo y formuló lo que hoy se conoce como las “12 reglas de Codd”. En esta unidad dedicaremos un apartado a repasar estas reglas. Finalmente, discutidos los fundamentos del modelo, brindaremos aspectos generales del lenguaje estándar de desarrollo de bases de datos relacionales llamado SQL.



Objetivo particular de la unidad

Reconocer las características teóricas que conforman el modelo relacional de base de datos. Además, identificar los elementos del mismo modelo, las reglas propuestas por Edgar Codd y el proceso de normalización de relaciones.



Unidad II. Modelo Relacional



Lo que sé:

Menciona de manera breve qué es una base de datos, un sistema administrador de base de datos y por qué es importante que una empresa cuente con este sistema.

Comparte tus respuestas en el **Foro** de la unidad **Modelo relacional**.

Pulse el botón **Colocar un nuevo tema de discusión aquí**. Escriba en el apartado **Asunto** el título de su aportación, redacte su comentario en el área de texto y de clic en el botón **Enviar al foro**.

Temas de la unidad II

1. Introducción
 - 1.1. Modelos pre-relacionales
 - 1.2. Modelos pos-relacionales
2. Definición de relación
 - 2.1. Partes
3. Propiedades de una relación
4. Dominio y tipos de datos
5. Álgebra relacional y cálculo relacional
6. Normalización
 - 6.1. Formas normales
 - 6.2. Proceso de descomposición sin pérdida
7. Reglas de CODD
8. Estándar SQL



Unidad II. Modelo Relacional



Resumen de la unidad

En esta unidad conocimos el modelo propuesto por Codd, matemático investigador de IBM, el cual ofreció a las bases de datos los fundamentos esenciales para que hoy en día sea el modelo sobre el cual se basan el diseño de los demás. Asimismo conocimos sus antecesores y sus predecesores, así como sus características principales (relaciones) y las reglas propuestas por este investigador que definen los requisitos de un manejador de base de datos relacional.

De igual manera, abordamos los dominios, los tipos de datos y la manera que se relaciona con los atributos; el algebra relacional en función con la operación de las relaciones (normalización).

Por supuesto, se abordó el lenguaje de programación que es un claro ejemplo del modelo de Codd, el lenguaje conocido como SQL (Structured Query Language).

Finalmente se abordarán las etapas de la Normalización aplicadas al diseño de las Bases de Datos.



Unidad II. Modelo Relacional



Tema 1. Introducción a los modelos

Objetivo del tema

Identificar los elementos básicos y estructura de los modelos pre-relacionales y post-relacionales.

Desarrollo

A finales de 1968, Edgar F. Codd, matemático investigador de IBM, propuso el uso de las matemáticas para dar cierto rigor al campo de las bases de datos. Codd puso sus ideas en un artículo que hoy es clásico: “A Relational Model of Data for Large Shared Data Banks (consultar <http://www.seas.upenn.edu/~zives/03f/cis550/codd.pdf>, 31 de marzo del 2009).¹ En ese artículo y subsiguientes, Codd presenta los conceptos fundamentales del modelo y sus beneficios frente a la tecnología de ese tiempo.

Tal como lo mencionamos en la unidad 1, la manera de manejar datos antes del advenimiento de las bases de datos era mediante archivos de datos. El uso de estos sistemas de archivos causaba distintos problemas que provocaban pérdidas de datos, datos duplicados innecesariamente, datos incompletos y erróneos.

Los investigadores en computación de aquellos tiempos comenzaron a proponer nuevas opciones de almacenamiento hasta que surgieron los primeros modelos de bases de datos. A continuación hablaremos de algunos de ellos.

Son básicamente dos modelos los antecesores del modelo relacional de base de datos. Ambos están basados en una estructura de nodos interconectados que almacenan la información.

¹ E. F. Codd. “A Relational Model of Data for Large Shared Data Banks” en CACM 13, no. 16 (junio, 1970).



Unidad II. Modelo Relacional

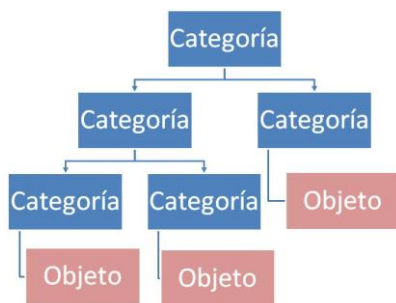


Modelos de bases de datos

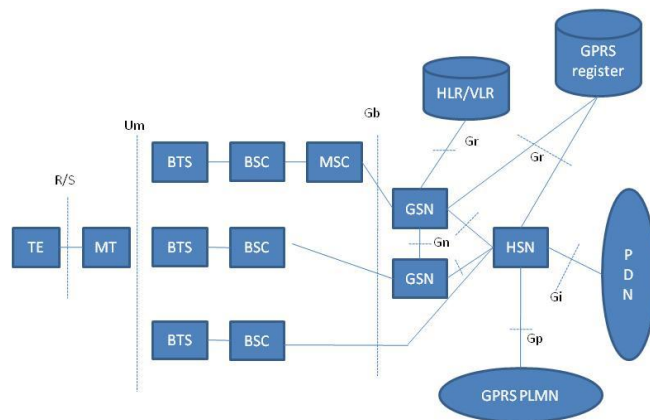
Modelos pre-relacionales

Dentro de estos modelos, podemos observar en un primer plano el modelo jerárquico, que interconecta nodos en una jerarquía estricta de padre e hijos, donde no podía haber relación entre nodos de distintos niveles o entre los del mismo nivel. Este modelo fue útil hasta que se empleó para resolver problemas de almacenamiento más complejos. La interconexión de nodos y, por consiguiente, el uso de apuntadores, comenzó a ser un inconveniente difícil de manejar por los sistemas de aplicación. El otro problema principal de este modelo es que no puede implementar relaciones de M:M (muchos a muchos) entre instancias de entidades del mundo real .

El segundo modelo es el de red. Se trata de una interconexión de nodos mediante apuntadores, sin la restricción del jerárquico, es decir, pueden salir de cada nodo varios arcos apuntando a otros nodos. A diferencia del modelo jerárquico, en este modelo se permiten las conexiones entre nodos de cualquier tipo, por lo que resulta bueno para relaciones M:M. Su principal desventaja radica en los problemas de implementación para lograr un rendimiento óptimo.



Modelo jerárquico simple



Modelo de red

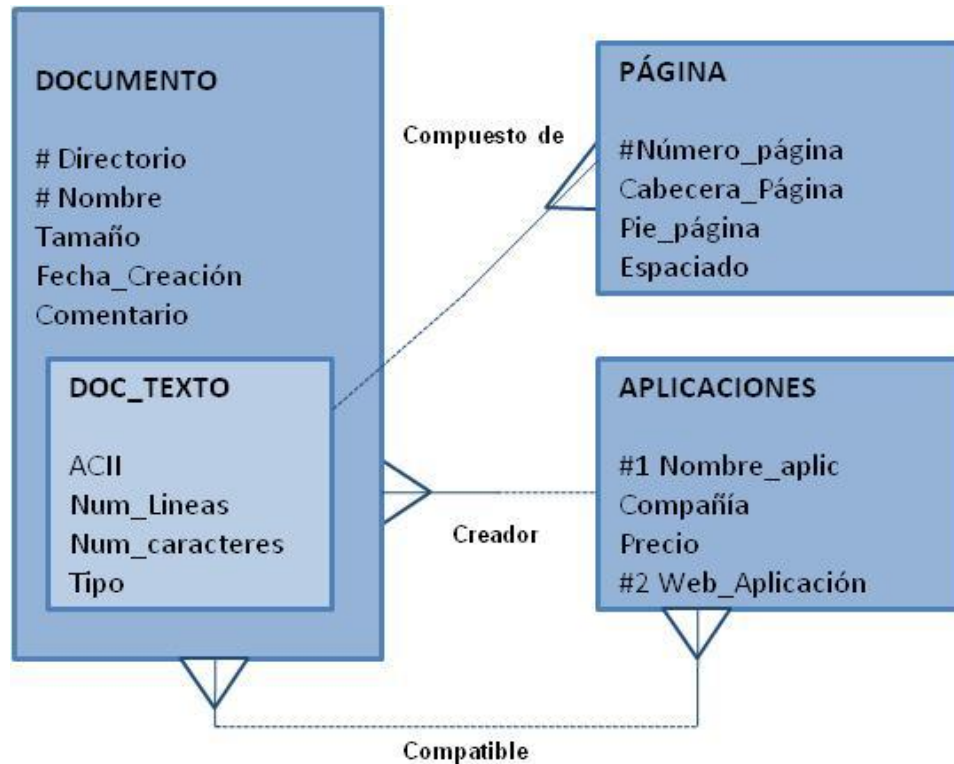


Unidad II. Modelo Relacional



Modelos pos-relacionales

Los modelos pos-relacionales más conocidos son el modelo orientado a objetos, el distribuido, el deductivo, el multidimensional y el modelo semiestructurado.



Tendremos la oportunidad de revisar el modelo orientado a objetos en la unidad 3, así que no diremos nada sobre él en este apartado. El modelo distribuido está basado en el modelo relacional o en el orientado a objetos y su principal característica es que la base de datos está fragmentada en una red. Así, partes de las tablas de usuarios se encuentran en distintos lugares ya sea a nivel horizontal (renglones) o a nivel vertical (columnas). Uno de los retos principales de estos manejadores es que deben presentar la información al usuario como si las tablas estuvieran almacenadas localmente. Tanto la consulta como la actualización de datos deben estar garantizadas desde cualquier punto de la red.



Unidad II. Modelo Relacional



El **modelo deductivo** se apoya en un conjunto de datos y reglas de inferencia. Es conocido también como modelo inferencial o modelo basado en la lógica. Las consultas de datos en este modelo son demostraciones de axiomas deductivos. El objetivo primordial de este tipo de bases de datos es inferir ciertos datos a partir de las reglas.

Las **bases de datos multidimensionales** son llamadas así porque los datos están almacenados en arreglos multidimensionales. Estos arreglos pueden ser interpretados en términos de variables dependientes e independientes. Las últimas determinan las dimensiones sobre las que se guardan valores (variable dependiente). Este modelo forma parte de los sistemas conocidos como MOLAP (procesamiento analítico en línea multidimensional) que tiene por objeto la generación de informes analíticos de datos.

Por último, el **modelo semiestructurado** surge ante la necesidad de que cada elemento almacenado cuente con un conjunto de atributos propio, sin necesidad de que este conjunto sea obligatorio como sucede en el modelo relacional. Este tipo de bases de datos se crean con el lenguaje de marcado extensible XML.



Unidad II. Modelo Relacional



ACTIVIDAD 1

Realizar un cuadro comparativo de las características de los Modelos Pre-relacionales, Pos-relacionales.

Mod. Pre-relacionales	Mod. Pos-relacionales

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.



Unidad II. Modelo Relacional



ACTIVIDAD 2

Anota las características de:

- a) El Modelo Jerárquico.
- b) El Modelo de Red.
- c) El Modelo de Distribuido.
- d) El Modelo de Deductivo.
- e) El Modelo Semiestructurado.

Para enviar tu respuesta, pulsa el botón **Editar mi envío**; se mostrará un editor de texto en el cual puedes redactar tu información; una vez que hayas concluido, salva tu actividad pulsando el botón **Guardar cambios**.



Unidad II. Modelo Relacional



Autoevaluación

Selecciona si las aseveraciones son verdaderas (V) o falsas (F). Al finalizar obtendrás tu calificación de manera automática.

	Verdadera	Falsa
1. El modelo Semiestructurado es Prerrelacional.	()	()
2. El Modelo Jerárquico es Posrelacional.	()	()
3. El Modelo de Red es Prerrelacional.	()	()
4. El Modelo Semiestructurado se crea con el Lenguaje XML.	()	()
5. El Modelo Deductivo es Posrelacional.	()	()

Bibliografía básica

Autor	Capítulo	Páginas

Sitios electrónicos

Sitio	Descripción



Unidad II. Modelo Relacional



Tema 2. Definición de relación

Objetivo del tema

Identificar las diferencias de los diversos modelos de diseño de Bases de Datos.

Desarrollo

El modelo relacional de bases de datos es llamado así porque se basa en estructuras de almacenamiento de datos llamadas relaciones. Las dos características fundamentales del modelo relacional de bases de datos son:

1. Los datos son percibidos por el usuario como relaciones y nada más que relaciones.
2. Para consultar los datos, el usuario cuenta con operadores que generan nuevas relaciones a partir de otras (álgebra relacional).

El modelo relacional se fundamenta en una teoría abstracta de datos que toma ciertos aspectos de las matemáticas (teoría de conjuntos y lógica de predicados). De esta manera, podemos definir una relación como un conjunto de tuplas y atributos, y que **se compone de dos partes: encabezado y cuerpo**.

El **encabezado** de una relación denota un cierto predicado o función valuada como verdadera. Por su parte, cada fila en el cuerpo denota una cierta proposición verdadera obtenida del predicado por medio de la sustitución de ciertos valores, del tipo apropiado, en los indicadores de posición o parámetros de ese predicado. En otras palabras, podemos decir que el cuerpo es un conjunto de instancias o ejemplos del predicado. Veámoslo en un ejemplo:



Unidad II. Modelo Relacional



Predicado

El empleado NUMEMP se llama NOMBRE_EMP, trabaja en el departamento NUMDEPTO y gana un salario SALARIO_EMP

Proposiciones verdaderas obtenidas del predicado:

El empleado E1 se llama Azucena López, trabaja en el departamento D1 y gana el salario 20,000.

El empleado E2 se llama Alberto Juárez, trabaja en el departamento D1 y gana el salario 8,000.

Si representamos gráficamente lo anterior tendríamos una relación como la siguiente:

NUMEMP	NOMBRE_EMP	NUMDEPTO	SALARIO_EMP
E1	Azucena López	D1	20,000
E2	Alberto Juárez	D1	8,000

Ahora bien, desde la perspectiva de la teoría de conjuntos, el encabezado es un conjunto de n atributos necesariamente distintos de la forma NombredeAtributo:Nombre deTipo. Mientras que el cuerpo 1, el número de atributos de una relación se conoce como grado y el número de tuplas como cardinalidad.



Unidad II. Modelo Relacional



ACTIVIDAD 1

De la tabla que se desarrolla en este tema, incrementar un Departamento adicional para alterar la Cardinalidad y comprobar su lógica si se adicionan dos personas más.

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.

ACTIVIDAD 2

De la Tabla anterior, adicionar una columna adicional con el nombre de CATEGORIA, la cual tendrá tres puestos A1, A2 Y A3. Asignárselos a personas distintas y en diferentes NUMDEPTO para comprobar su lógica y su cardinalidad.

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.



Unidad II. Modelo Relacional



ACTIVIDAD 3

Contesta lo siguiente:

1. El predicado se compone de: _____
2. El cuerpo esta compuesto por: _____
3. La Relación de NUMDEPTO a NUMEMP es de: _____

Para enviar tu respuesta, pulsa el botón **Editar mi envío**; se mostrará un editor de texto en el cual puedes redactar tu información; una vez que hayas concluido, salva tu actividad pulsando el botón **Guardar cambios**.



Unidad II. Modelo Relacional



Autoevaluación

Selecciona si las aseveraciones son verdaderas (V) o falsas (F). Al finalizar tendrás tu calificación de manera automática.

	Verdadera	Falsa
1. El Modelo relacional toma de base al modelo virtual de relaciones..	()	()
2. El encabezado de la Tupla está dentro del Cuerpo de la Tabla.	()	()
3. El número de Atributos es igual que el número de campos de una Tabla.	()	()
4. El modelo Relacional se funda en Aspectos matemáticos.	()	()
5. Los Formatos de los Datos influyen en la Cardinalidad.	()	()

Bibliografía básica

Autor	Capítulo	Páginas

Sitios electrónicos

Sitio	Descripción



Unidad II. Modelo Relacional



Tema 3. Propiedades de una relación

Objetivo del tema

Identificar las propiedades de las relaciones y los tipos de claves existentes en las entidades.

Desarrollo

Una relación debe contar con cuatro propiedades que son:

1. No existen tuplas duplicadas.

La teoría de conjuntos descarta los elementos duplicados, además, no podemos afirmar dos veces el mismo hecho verdadero. La siguiente no es una relación:

NUMERO	NOMBRES	APELLIDOS
1	Arturo	Jiménez
1	Arturo	Jiménez

2. Las tuplas están en desorden.

La teoría de conjuntos nos dice que sus elementos están en desorden. Esto implica que no existen los conceptos de “tupla 1”, “siguiente tupla”, “cuarta tupla”; entonces nos debemos referir a una tupla por su proposición verdadera.

3. Los atributos están en desorden.

Esto es porque el encabezado es un conjunto de atributos y los conjuntos no permiten elementos duplicados. Por tanto, tampoco hay conceptos de “siguiente atributo”, “primer atributo”, siempre se hace referencia a ellos por nombre, nunca por su posición



Unidad II. Modelo Relacional



4. Los atributos deben contener valores atómicos.

Esto significa que cada valor para cada tupla en un determinado atributo no debe ser semánticamente divisible y por tanto tiene que representar un valor único. La siguiente no es una relación, debido a que el atributo DEPTO no guarda valores atómicos:

NUMERO	NOMBRES	APELLIDOS	DEPTO
1	Arturo	Jiménez	01 Finanzas
2	Arturo	Jiménez	02 Contabilidad

En una relación es posible encontrar tres tipos de claves:

Clave candidata	Clave principal	Clave foránea
Es un atributo que identifica de manera única a cada tupla de una relación. Una relación puede tener varias claves candidatas y estas pueden ser simples, formadas por un atributo, o compuestas cuando se forman por varios atributos. A las últimas se les llama superclaves.	Una clave principal es una clave candidata seleccionada para ser la clave principal de la relación. En este caso, a diferencia de las claves candidatas, sólo existe una por relación. También pueden ser superclaves.	Una clave foránea es un atributo que hace referencia a una clave principal en otra relación. Esta referencia implica que no pueda existir un valor en la clave foránea que no exista primero en la clave principal.



Unidad II. Modelo Relacional



ACTIVIDAD 1

En la Tabla que se observa en el desarrollo del tema, cambia el Orden de los Campos y determina ahora cuál es la Clave Principal.

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.

ACTIVIDAD 2

Basado en las modificaciones que hiciste en la Tabla anterior, crea una Superclave con los dos primeros campos y determina su formato, orden y relaciones con otra Entidad llamada Departamento.

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.



ACTIVIDAD 3

Responde lo siguiente:

1. Anota el concepto de:

- a) Clave foránea.
- b) Superclave.

2. Anota las cuatro propiedades de una relación.

3. Define qué es una tupla.

4. Indica el número de claves principales que pueden existir en una relación.

Para enviar tu respuesta, pulsa el botón **Editar mi envío**; se mostrará un editor de texto en el cual puedes redactar tu información; una vez que hayas concluido, salva tu actividad pulsando el botón **Guardar cambios**.



Unidad II. Modelo Relacional



Autoevaluación

Selecciona si las aseveraciones son verdaderas (V) o falsas (F). Al finalizar tendrás tu calificación de manera automática.

	Verdadera	Falsa
1. Una Clave de relación es igual a una Llave.	()	()
2. La Clave Candidata, es una SuperClave.	()	()
3. La Clave Principal debe ser un Tipo de Dato Numérico.	()	()
4. El valor atómico es divisible.	()	()
5. La Tupla es divisible.	()	()

Bibliografía básica

Autor	Capítulo	Páginas

Sitios electrónicos

Sitio	Descripción



Unidad II. Modelo Relacional



Tema 4. Dominio y tipos de datos

Objetivo del tema

Identificar los elementos de un dominio y sus atributos para asignarles valores.

Desarrollo

Un dominio es el conjunto de valores válidos para un atributo. De esta manera, podríamos decir que cada atributo tendrá asociado siempre un dominio. Es común relacionar este concepto con el de tipo de dato, pero no son lo mismo. Un dominio es más restrictivo en cuanto a la semántica de lo que guarda, por ejemplo, el dominio CIUDAD es el conjunto de todas las ciudades posibles, misma que se expresan en caracteres alfanuméricos mediante un tipo de dato carácter (`char`, `varchar`: son datos compuestos y de valores grandes como puede ser 200^{34}) que podría aceptar cualquier combinación de estos.

Además, es posible utilizar ciertas restricciones para que los dominios acepten sólo determinados valores, pero siempre sobre la base de un tipo de dato simple como *integer* o *char*. Los tipos de datos están predefinidos en nuestro manejador de base de datos y contamos con la posibilidad de hacer conversiones entre distintos tipos.

Tipos de datos de valores grandes

En versiones anteriores de SQL Server, el trabajo con tipos de datos de valores grandes requería un tratamiento especial. Los tipos de datos de valores grandes son aquellos que superan el tamaño máximo de fila, 8 KB. SQL Server 2005 introduce un especificador máximo para los tipos de datos `varchar`, `nvarchar` y `varbinary` para permitir el almacenamiento de valores de hasta 2^{31} bytes. Las



Unidad II. Modelo Relacional



columnas de tablas y las variables Transact-SQL pueden especificar los tipos de datos `varchar(max)`, `nvarchar(max)` o `varbinary(max)`. Los escenarios principales en los que se trabaja con tipos de valores grandes implican su recuperación de una base de datos o agregarlos a una base de datos. Lo siguiente describe diferentes enfoques para realizar estas tareas.

Recuperación de tipos de valores grandes de una base de datos

Al recuperar un tipo de datos de valores grandes, que no sean binarios, de una base de datos, como el tipo de datos `varchar(max)`, un planteamiento es leer esos datos como una secuencia de caracteres. En el siguiente ejemplo, se usa el método `executeQuery` de la clase `SQLServerStatement` para recuperar datos de la base de datos y devolverlos como un conjunto de resultados. A continuación, se usa el método `getCharacterStream` de la clase `SQLServerResultSet` para leer los datos de valores grandes en el conjunto de resultados.

(*)[http://msdn.microsoft.com/es-es/library/ms378813\(SQL.90\).aspx](http://msdn.microsoft.com/es-es/library/ms378813(SQL.90).aspx)



ACTIVIDAD 1

Crea una Tabla con dos Atributos:

1. Temperatura
2. Escala

Ambos con sus respectivos atributos.

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.

ACTIVIDAD 2

Define las características del atributo Temperatura y sus tres Escalas y enlazarlas con las temperaturas de la tabla anterior.

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.



Unidad II. Modelo Relacional



ACTIVIDAD 3

Responde el siguiente cuestionario:

1. Define qué es un atributo.
2. Define qué es un dominio.
3. Cómo puede ser un dato de tipo carácter.
4. Define que es una restricción.
5. Explica cómo es un tipo de Dato Simple.

Para enviar tu respuesta, pulsa el botón **Editar mi envío**; se mostrará un editor de texto en el cual puedes redactar tu información; una vez que hayas concluido, salva tu actividad pulsando el botón **Guardar cambios**.

Bibliografía básica

Autor	Capítulo	Páginas

Sitios electrónicos

Sitio	Descripción



Tema 5. Álgebra relacional y cálculo relacional

Objetivo del tema

Identificar y aplicar las operaciones del álgebra relacional para aplicarlas a los tipos de consultas sobre una bases de datos.

Desarrollo

Álgebra relacional

El álgebra relacional es el conjunto de operadores que permiten manipular las relaciones (consultar datos). Estos operadores siempre generan nuevas relaciones a partir de otras. Las operaciones de álgebra relacional son:

1. Restricción

Regresa una relación que contiene todas las tuplas de una relación especificada, que satisfacen una condición. Un ejemplo sería:

R(ALUMNO)	
IDALUMNO	NOMBRE
1	Jaime
2	Raúl
3	Aurora

Restricción
 $IDALUMNO = 2$

R(RESPUESTA)	
IDALUMNO	NOMBRE
2	Raúl

2. Proyección

Regresa una relación que contiene un subconjunto de atributos de una relación especificada. Por ejemplo:



Unidad II. Modelo Relacional



R(ALUMNO)

IDALUMNO	NOMBRE
1	Jaime
2	Raúl
3	Aurora

Proyección
NOMBRE

R(RESULTADO)

NOMBRE
Jaime
Raúl
Aurora

3. Producto

Regresa una relación que contiene todas la tuplas posibles obtenidas por la combinación de todas las tuplas de la primera relación con todas las tuplas de la segunda relación. El siguiente es un ejemplo:

R(A)

IDA
1
2

PRODUCTO

R(B)

IDB
10
20

R(RESULTADO)

IDA	IDB
1	10
1	20
2	10
2	20

4. Unión

Regresa una relación que contiene todas las tuplas que aparecen en las dos relaciones, omitiendo las tuplas duplicadas. Mira el siguiente ejemplo:

R(A)

IDA	CVE
1	A
2	B

UNIÓN

R(B)

IDB	CVE
2	B
3	C
4	D

R(RESULTADO)

IDAB	CVE
1	A
2	B
3	C
4	D



Unidad II. Modelo Relacional



5. Intersección

Regresa una relación que contiene todas las tuplas que aparecen en las dos relaciones especificadas. El siguiente es un ejemplo:

R(A)			R(B)			R(RESULTADO)	
IDA	CVE		IDB	CVE		IDAB	CVE
1	A	INTERSECCIÓN	2	B		2	B
2	B		3	C			
			4	D			

6. Diferencia

Regresa una relación que contiene todas las tuplas que aparecen en la primera pero no en la segunda de las dos relaciones específicas. Por ejemplo:

R(A)			R(B)			R(RESULTADO)	
IDA	CVE		IDB	CVE		IDAB	CVE
1	A	DIFERENCIA	2	B		1	A
2	B		3	C			
			4	D			

7. Junta (join)

Regresa una relación que contiene tuplas de dos relaciones a partir de la combinación de valores de uno o varios atributos en común. De no existir una columna en común, no sería posible la junta. Observa el siguiente ejemplo:

R(PROD)			R(VTA)			R(RESULTADO)	
IDP	NOMBRE		IDP	CANT		NOMBRE	CANT
1	Aretes	JUNTA IDP = IDP	2	100		Pulseras	100
2	Pulseras		1	30		Aretes	30
3	Collares		3	250		Collares	250



Unidad II. Modelo Relacional



8. División

A partir de dos relaciones unarias $r(A)$ y $r(B)$ y una relación binaria $r(C)$, regresa una relación que contiene todas las tuplas de $r(A)$ siempre y cuando su combinación con todas las tuplas de $r(B)$ aparezca en $r(C)$. Observa bien el siguiente ejemplo y verifica por qué esas tuplas de $r(A)$ aparecen en la relación RESULTADO.

R(A)	R(B)	R(C)	R(RESULTADO)																
<table border="1"><thead><tr><th>IDA</th></tr></thead><tbody><tr><td>1</td></tr><tr><td>2</td></tr></tbody></table>	IDA	1	2	<table border="1"><thead><tr><th>IDB</th></tr></thead><tbody><tr><td>10</td></tr><tr><td>20</td></tr></tbody></table>	IDB	10	20	<table border="1"><thead><tr><th>IDA</th><th>IDB</th></tr></thead><tbody><tr><td>1</td><td>10</td></tr><tr><td>2</td><td>10</td></tr><tr><td>2</td><td>20</td></tr></tbody></table>	IDA	IDB	1	10	2	10	2	20	<table border="1"><thead><tr><th>IDA</th></tr></thead><tbody><tr><td>2</td></tr></tbody></table>	IDA	2
IDA																			
1																			
2																			
IDB																			
10																			
20																			
IDA	IDB																		
1	10																		
2	10																		
2	20																		
IDA																			
2																			

DIVISIÓN

El resultado es 2 ya que es la única tupla de $R(A)$ que al combinarse con todas las de $R(B)$ (2-10, 2-20) aparece en $R(C)$.

9. Cálculo relacional

Es posible identificar dos tipos de cálculo relacional: el cálculo de tuplas y el cálculo de dominios. En ambos casos no se especifica cómo realizar la consulta, esto significa que se indica únicamente una proposición verdadera que deben cumplir las tuplas o los dominios; en el álgebra relacional se indica el procedimiento para resolver la consulta, en el cálculo no. El principal operador de este tipo de consultas es el operador “existe” (*exists* en SQL).



Unidad II. Modelo Relacional



ACTIVIDAD 1

Desarrolla las operaciones estudiadas en el tema a la Tabla “Libro” con los Atributos Libro, Tema y Autor.

LIBRO	TEMA	AUTOR
Fundamentos de finanzas	Finanzas básicas	Gitman Lawrence
Metodología de la programación	Estructura de datos	Joyanes Aguilar Luis
Introducción a las finanzas	Finanzas básicas	Pedromo Abraham

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.



Unidad II. Modelo Relacional



ACTIVIDAD 2

Desarrolla las operaciones estudiadas en el tema a la Tabla “Temperatura” con los Atributos Ciudad, Temperatura.

CIUDAD	TEMPERATURA MAX	ESTADO	HUMEDAD
Toluca	12	Edo. Mex.	45%
Guadalajara	27	Jalisco	35%
Oaxaca	34	Oaxaca	67%
Puebla	31	Puebla	50%

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.



Unidad II. Modelo Relacional



ACTIVIDAD 3

Contesta de manera breve lo siguiente:

1. Una Operación en el Algebra Relacional es:
2. Indicar cómo funciona una Restricción con relación a las tuplas.
3. La Proyección consiste en:
4. El Producto consiste en:
5. La Unión consiste en:

Para enviar tu respuesta, pulsa el botón **Editar mi envío**; se mostrará un editor de texto en el cual puedes redactar tu información; una vez que hayas concluido, salva tu actividad pulsando el botón **Guardar cambios**.

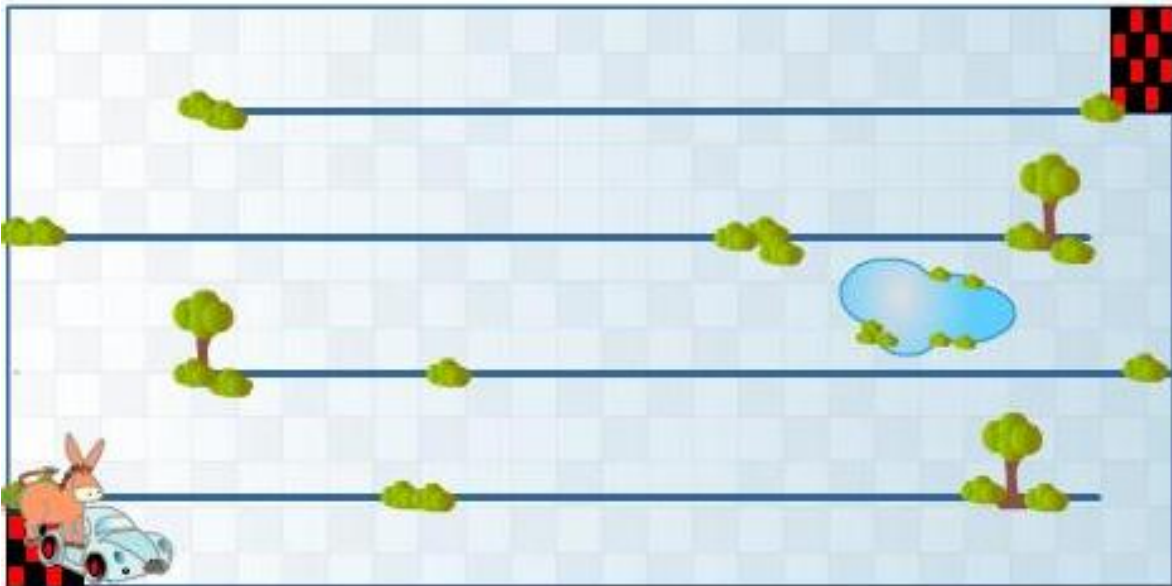


Unidad II. Modelo Relacional



Autoevaluación

Participa en el siguiente Rally. Para avanzar debes responder acertadamente.



1 Un Dominio acepta dos valores.

V () F ()

2 Un manejador de Bases de Datos tiene tipos de datos ilimitados.

V () F ()

3 Los Datos char se pueden convertir en integer.

V () F ()

4 Los Datos fecha constituyen un Dominio.

V () F ()

5 Un Atributo esta asociado a un Dominio.

V () F ()



Unidad II. Modelo Relacional



- 6 En el cálculo relacional se aplica un solo tipo de cálculo.

V () F ()

- 7 Una Restricción devuelve tuplas solicitadas.

V () F ()

- 8 La Proyección regresa una relación entre tuplas.

V () F ()

- 9 Una Unión necesita de una intersección.

V () F ()

- 10 Un Join contiene tuplas de dos relaciones.

V () F ()

Bibliografía básica

Autor	Capítulo	Páginas

Sitios electrónicos

Sitio	Descripción



Unidad II. Modelo Relacional



Tema 6. Normalización

Objetivo del tema

Aplicar las etapas de la normalización en el diseño de las bases de datos.

Desarrollo

La normalización es el proceso de ajuste de relaciones a ciertas reglas llamadas formas normales. Tiene por objetivo reducir los problemas de redundancia y actualización en las relaciones; su ventaja es que lo hace mediante un procedimiento bien formalizado.

A continuación te presentamos la definición de las formas normales. E. F. Codd propuso las tres primeras y con la aportación de posteriores investigadores en computación se establecieron las demás.

Formas normales

Como habíamos mencionado, las formas normales son reglas estrictas que deben cumplir las relaciones para disminuir problemas de redundancia y actualización. Varias de ellas están basadas en el concepto de Dependencia Funcional que explicaremos brevemente.

Dependencias funcionales

Una dependencia funcional (DF) es una relación *muchos a uno* que va de un conjunto de atributos a otro dentro de una determinada relación. En otras palabras: sea R una relación y sean X y Y subconjuntos cualesquiera del conjunto de atributos de R, entonces podemos decir que Y es dependiente funcionalmente de X, en símbolos $X \rightarrow Y$ ², si y sólo si en todo valor válido posible de R, cada valor

² También se puede decir que “X determina funcionalmente a Y.”



Unidad II. Modelo Relacional



X está asociado precisamente con un valor de Y; siempre que dos tuplas coincidan en su valor X, también coincidirán en su valor Y.

La parte izquierda de una DF se denomina determinante y la derecha dependiente.

Dependencias triviales y no triviales

Una dependencia trivial se da si y solo si la parte derecha es un subconjunto de la parte izquierda. Las no triviales son las que no son triviales. Estas son las que realmente importan en el proceso de normalización mientras que las otras no son tomadas en cuenta.

Dependencias transitivas

Supongamos que tenemos una relación R con tres atributos: A, B y C, tales que las DFs $A \rightarrow B$ y $B \rightarrow C$ son válidas para R. Entonces es fácil ver que la DF $A \rightarrow C$ también es válida en dicha relación. Aquí la DF $A \rightarrow C$ es un ejemplo de DF transitiva y decimos que C depende de A transitivamente a través de B.

Cierre de un conjunto de dependencias

Al conjunto de todas las DFs implicadas en una relación se le llama cierre de dependencias y se escribe S^+ . Para obtener este cierre se utilizan reglas de inferencia para obtener nuevas DFs a partir de las ya dadas. Los axiomas de Armstrong nos permiten realizar esta labor y son los siguientes:

Sean A, B y C subconjuntos cualesquiera del conjunto de atributos de la relación dada R, entonces pueden ser aplicados los siguientes axiomas para producir nuevas DFs:



Unidad II. Modelo Relacional



Reflexividad:	si B es subconjunto de A, entonces $A \rightarrow B$.
Aumento:	si $A \rightarrow B$, entonces $AC \rightarrow BC$.
Transitividad:	si $A \rightarrow B$ y $B \rightarrow C$, entonces $A \rightarrow C$.
Autodeterminación:	$A \rightarrow A$.
Descomposición:	si $A (BC$, entonces $A (B$ y $A (C$.
Unión:	si $A (B$ y $A (C$, entonces $A (BC$.
Composición:	si $A (B$ y $C (D$, entonces $AC (BD$.

Te pongo a continuación un ejemplo del uso de estos axiomas. Dadas las siguientes DF:

- 1) $A (BC$
- 2) $B (E$
- 3) $CD (EF$

Se pueden producir nuevas DFs a partir de la aplicación de los siguientes axiomas:

- 4) $A (C$ por descomposición de 1.
- 5) $AD (CD$ por aumento en 4.
- 7) $AD (EF$ por transitividad entre 5 y 3.
- 8) $AD (F$ por descomposición en 7.

Dependencias irreducibles

Una dependencia funcional se considera irreducible si la parte derecha involucra sólo un atributo y no es posible descartar ningún atributo del determinante sin cambiar el cierre. En otras palabras, en el caso de que la relación presente una clave principal compuesta, todo atributo a la izquierda de la DF debe depender de toda la clave y no sólo de una parte.



Unidad II. Modelo Relacional



Primera forma normal (1FN)

Una relación está en 1FN si y sólo si toda tupla contiene exactamente un valor para cada atributo. Este concepto es el que ya habíamos explicado como la propiedad de una relación, por lo que muchos autores proponen que toda relación está en 1FN por definición.

Segunda forma normal

Una relación está en 2FN si y sólo si está en 1FN y todo atributo que no sea clave es dependiente irreduciblemente de la clave primaria. Podemos decir, en otras palabras que la 2FN exige que todas la DFs de una relación sean irreducibles.

Tercera forma normal

Una relación está en 3FN si y sólo si está en 2FN y todos los atributos que no son clave son dependientes en forma no transitiva de la clave primaria. La 3FN elimina la posibilidad de DFs transitivas.

Forma normal de Boyce/Codd

Una relación está en FNBC si y sólo si toda DF no trivial, irreducible a la izquierda, tiene una clave candidata como su determinante.

Proceso de descomposición sin pérdida

Pero qué sucede si una relación no cumple plenamente con una determinada forma normal. Como te habrás dado cuenta, las formas normales también son acumulativas, es decir, no es posible estar en una forma normal más alta sin cumplir las formas normales inferiores.



Unidad II. Modelo Relacional



Bueno, si una relación no cumple una forma normal, debemos descomponerla mediante un proceso conocido como descomposición sin pérdida. Éste consiste en una proyección de la relación para obtener nuevas relaciones que cumplan la forma normal exigida, y decimos que es sin pérdida si al juntar de nuevo las proyecciones regresamos a la relación original, preservando tanto el grado como la cardinalidad.

Revisemos ahora las llamadas **formas normales superiores**: 4FN y 5FN, para lo cual será necesario estudiar dos tipos más de dependencias.

1. Dependencias multivaluadas

Las dependencias multivaluadas se dan entre dos atributos. Uno de ellos caracteriza o determina a un conjunto bien definido de valores del otro atributo. Esta caracterización es independiente de otros atributos. Las dependencias multivaluadas se dan entre atributos multivaluados independientes entre sí donde se establece una combinatoria de todos contra todos. Veamos un ejemplo ilustrativo de De Miguel (2000: 178).

R(ASIGNATURAS)

NOM_ASIGNATURA	PROFESOR	TEXTO
Ficheros y BD	Sr. Sánchez Sra. Hidalgo	Concepción y diseño de BD Fundamentos de BD
BD avanzadas	Sra. Hidalgo Sr. Martín	Diseño de BD avanzadas

Podemos hacer las siguientes observaciones sobre el ejemplo:



Unidad II. Modelo Relacional



Un profesor debe utilizar todos los textos. Por esto, un profesor no puede determinar un texto.

La clave primaria sería una superclave con los tres atributos. Si normalizamos el modelo inicial obtendríamos una relación en FNBC, pero que tendría mucha redundancia.

R(ASIGNATURAS)

NOM_ASIGNATURA	PROFESOR	TEXTO
Ficheros y BD	Sr. Sánchez	Concepción y diseño de BD
Ficheros y BD	Sr. Sánchez	Fundamentos de BD
Ficheros y BD	Sra. Hidalgo	Concepción y diseño de BD
Ficheros y BD	Sra. Hidalgo	Fundamentos de BD
BD avanzadas	Sra. Hidalgo	Diseño de BD avanzadas
BD avanzadas	Sr. Martín	Diseño de BD avanzadas

Una dependencia multivaluada $X \twoheadrightarrow Y$ se da cuando para cada valor de X hay un conjunto de cero o más valores de Y , independientes de los valores de los otros atributos de la relación. La siguiente relación, tomada de De Miguel (2000: 180), no tiene una dependencia multivaluada ya que el texto “Modelo Relacional” no aparece en inglés.



Unidad II. Modelo Relacional



R(CURSOS)

COD_CURSO	TEXTO	IDIOMA
A2783	Introducción a las BD	Español
A2783	Introducción a las BD	Inglés
A2783	Modelo relacional	Español
B2341	Concepción y diseño de BD	Francés
B2341	Modelo relacional	Español

Cuarta forma normal (4FN)

Una relación está en cuarta forma normal si, y sólo si, las dependencias multivaluadas tienen como determinante una clave. En este sentido, la relación ASIGNATURA no está en 4FN, por lo que es necesario descomponerla en dos relaciones:

ASIGNATURA_PROFESOR (Nom_asignatura, Profesor)

ASIGNATURA_TEXTO (Nom_asignatura, Texto)

2. Dependencias de combinación

Este tipo de dependencia se da cuando existe interdependencia entre los atributos de una relación y la descomposición en dos relaciones causa pérdida de información. Por tanto, la descomposición tiene que ser en varias proyecciones. Esta dependencia es también llamada de junta y se expresa así:

DJ * (R1, ..., Rj)



Unidad II. Modelo Relacional



Y significa que $R = R_1 \text{ join } R_2 \text{ join}.. \text{ join } R_j$. Veamos un ejemplo de De Miguel (2000: 190).

R(EDITA)

EDITORIAL	IDIOMA	TEMA
RA-MA	Inglés	BD
RA-MA	Español	CASE
Addison Wesley	Español	BD
RA-MA	Español	BD

Esta relación implica que si se publica un tema de bases de datos (BD) en un determinado idioma, por ejemplo español, entonces todas las editoriales deben publicar ese tema y en ese idioma. Por esto, si Addison Wesley publicara BD en francés, sería necesario que RA-MA también lo hiciera, provocando insertar una tupla adicional.

La descomposición de la relación en dos relaciones provocaría tuplas espurias. Por ejemplo:

R(EDITA1)		R(EDITA2)	
Editorial	Idioma	Idioma	Tema
RA-MA	Inglés	Inglés	BD
RA-MA	Español	Español	CASE
Addison Wesley	Español	Español	BD

R(EDITA1) join R(EDITA2)		
Editorial	Idioma	Tema
RA-MA	Inglés	BD
RA-MA	Español	CASE
RA-MA	Español	BD
Addison Wesley	Español	CASE
Addison Wesley	Español	BD

← Tupla espuria



Unidad II. Modelo Relacional



Para que sea una descomposición sin pérdida sería necesario descomponer en tres relaciones, agregando la siguiente relación:

R(EDITA3)

Editorial	Tema
RA-MA	BD
RA-MA	CASE
Addison Wesley	BD

Quinta forma normal (5FN)

Una relación está en 5FN si, y sólo si, no existen dependencias de combinación. Si dicha relación no está en 5FN se deben hacer tantas proyecciones como descriptores involucrados en la dependencia. En el ejemplo eran tres atributos involucrados en la dependencia de junta y por eso se hicieron tres proyecciones.



ACTIVIDAD 1

Construye la Base de Datos Clientes con las Tablas Facturas, con los Atributos fecha, código de producto, importe, folio de factura, Artículos (atributos: código de producto, descripción de artículo, precio de producto, folio de factura) y Ventas (atributos: fecha de ventas, importe de ven tas, código de producto, folio de facturas).

Construye la Base de Datos Clientes con las siguientes Tablas y Atributos:

1. Facturas, con los atributos: fecha, código de producto, importe, folio de factura.
2. Artículos, con los atributos: código de producto, descripción de artículo, precio de producto, folio de factura.
3. Ventas, con los atributos: fecha de ventas, importe de ven tas, código de producto, folio de facturas.

Posteriormente, normaliza la Bases de Datos Clientes.

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.



ACTIVIDAD 2

Responde las siguientes preguntas:

1. Anote el concepto de Normalizar.
2. Menciona las etapas de la Normalización.
3. Qué es una Dependencia Funcional.
4. Qué es una Dependencia Trivial.
5. Qué es una Dependencia Transitiva.

Para enviar tu respuesta, pulsa el botón **Editar mi envío**; se mostrará un editor de texto en el cual puedes redactar tu información; una vez que hayas concluido, salva tu actividad pulsando el botón **Guardar cambios**.

ACTIVIDAD 3

Elabora un cuadro de dos columnas que resuma las formas normales y el tipo de dependencia funcional que eliminan.

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.



Unidad II. Modelo Relacional



Autoevaluación

Completa las siguientes frases, arrastrando la respuesta correcta al espacio en blanco que le corresponda. Al finalizar obtendrás tu calificación de manera inmediata.

1. La Dependencia Transitiva valida
2. La Dependencia se da si y solo si la parte derecha es un subconjunto de la parte izquierda.
3. Las Formas Normales son
4. La Relación de muchos a uno se llama
5. El Autor de las formas Normales es

Opciones:

otras relaciones entre atributos de una misma relación
reglas estrictas para disminuir problemas de redundancia
E. F. Codd
triviales
dependencia funcional



Unidad II. Modelo Relacional



Tema 7. Reglas Codd

Objetivo del tema

Reconocer las doce reglas de Codd que definen los requisitos del manejador de base de datos.

Desarrollo



E. F. Codd propuso doce reglas que definen los requisitos de un manejador de base de datos relacionales. No obstante la mayoría de los manejadores comerciales no cumplen al 100 por ciento todas las reglas, buena parte de ellas han sido contempladas en los software de base de datos. Las reglas son:

1 Regla de la información

Toda la información de una base de datos relacional está representada explícitamente a nivel lógico mediante valores en tablas.



Unidad II. Modelo Relacional



2 Regla de acceso garantizado

Todo dato (valor atómico) en una base de datos relacional es accesible de manera garantizada mediante la combinación del nombre de tabla, llave primaria y nombre de columna.

3 Regla del tratamiento sistemático de valores nulos

Los valores nulos (Null), que son diferentes a la cadena vacía, el carácter de espacio en blanco y al cero, son manejados por un sistema de bases de datos relacionales de manera sistemática con el objeto de representar la información desconocida o faltante; debe hacerlo de forma independiente del tipo de dato.

4 Regla del catálogo basado en el modelo relacional

La descripción de los datos dentro de una base de datos, es decir, el catálogo (nombres de tablas, nombres de columnas, tipos de datos de cada columna, nombres de restricciones, etc.) debe estar representada a nivel lógico de la misma manera que los datos normales de usuario, es decir, a través de tablas. Esto permitirá utilizar el mismo lenguaje relacional para recuperar datos del catálogo y datos normales de usuario.

5 Regla del sub-lenguaje de datos entendible

Un sistema relacional debe soportar varios tipos de lenguajes y varios modos de uso por parte del usuario. Sin embargo, debe existir al menos un sub-lenguaje relacional que permita expresar sentencias, mediante una sintaxis bien definida, con cadenas de caracteres. Este sub-lenguaje debe ser capaz de soportar las siguientes operaciones:



Unidad II. Modelo Relacional



- Definición de datos.
- Consulta de datos.
- Manipulación de datos.
- Restricciones de integridad.
- Manejo de autorizaciones para los datos.

6

Regla de la actualización de vistas

Todas las vistas que sean teóricamente actualizables deben ser actualizables por el sistema de bases de datos.

7

Regla de inserciones, actualizaciones y eliminaciones de alto nivel

La posibilidad de manejar una relación como un único operador aplica no sólo a la recuperación de datos sino también a la inserción, actualización y eliminación de datos. Debe ser posible realizar estas operaciones para un conjunto de renglones.

8

Independencia física de los datos

Los programas de aplicación no sufren modificaciones a pesar de los cambios en el nivel de físico de almacenamiento o en los métodos de acceso.

9

Independencia lógica de los datos

Los programas de aplicación no sufren modificaciones a pesar de los cambios hechos a las tablas.



Unidad II. Modelo Relacional



10

Regla de independencia de integridad

Las restricciones de integridad especificadas para una relación deben ser definidas con el sub-lenguaje de datos relacional, y almacenadas en el catálogo y no en los programas de aplicación.

11

Independencia de distribución

Un sistema relacional puede estar distribuido en distintos equipos o sitios de una red y las tablas deben ser vistas como si estuvieran localmente.

12

Regla de la no subversión

Ningún lenguaje de bajo nivel puede ser usado para violar las restricciones de integridad expresadas en el lenguaje relacional de alto nivel.



Unidad II. Modelo Relacional



ACTIVIDAD 1

Investiga si los manejadores de Bases de Datos Access © y Fox© cumplen con las reglas estudiadas en este tema.

Para enviar tu respuesta, pulsa el botón **Editar mi envío**; se mostrará un editor de texto en el cual puedes redactar tu información; una vez que hayas concluido, salva tu actividad pulsando el botón **Guardar cambios**.

ACTIVIDAD 2

Responde el siguiente cuestionario.

1. Anota el concepto de manejador de Bases de Datos.
2. Las relaciones a nivel lógico se expresa mediante:
3. ¿Qué son los valores nulos?
4. El Catalogo está constituido por:
5. Un Sistema Relacional debe soportar:

Para enviar tu respuesta, pulsa el botón **Editar mi envío**; se mostrará un editor de texto en el cual puedes redactar tu información; una vez que hayas concluido, salva tu actividad pulsando el botón **Guardar cambios**.



Unidad II. Modelo Relacional



Autoevaluación

Selecciona si las aseveraciones son verdaderas (V) o falsas (F). Al finalizar tendrás tu calificación de manera automática.

	Verdadera	Falsa
1. El Catálogo se representa a nivel lógico.	()	()
2. La Consulta de Datos es una operación.	()	()
3. La Actualización de Vistas se aplican a Interfaces Gráficas.	()	()
4. La Independencia Lógica de los Datos sufren cambios en el Almacenamiento de los Datos.	()	()
5. Si una Bases de Datos solo se puede consultar en dos terminales se estaría cumpliendo con la Regla No. 11.	()	()

Bibliografía básica

Autor	Capítulo	Páginas

Sitios electrónicos

Sitio	Descripción



Unidad II. Modelo Relacional



Tema 8. Estándar SQL

Objetivo del tema

Aplicar comandos del Lenguaje SQL y sus versiones más comunes a operaciones del manejo de Bases de Datos.

Desarrollo

En la sección anterior hemos visto que Codd propuso la utilización de un sub-lenguaje relacional que permitiera operaciones de definición, consulta, manipulación, restricción y control de acceso a datos. Hoy en día contamos con un lenguaje de programación para bases de datos relacionales que cumple en buena medida los requerimientos de Codd. Este lenguaje es conocido como SQL (Structured Query Language).

1. SQL89

La primera versión reconocida por la ANSI como estándar del SQL fue la de 1989, aunque el trabajo para desarrollar un estándar para bases relacionales había comenzado tiempo atrás en los laboratorios de investigación de IBM. Una versión inicial de la ANSI es de 1986, pero sufrió mejoras hasta llegar a la versión que hoy conocemos como SQL89.

Las principales características es que contaba con instrucciones DDL (create y drop), DML (select, insert, delete, update) y DCL (grant y revoke). Se había establecido que toda sentencia SQL debería comenzar con un verbo y después una serie de cláusulas que comienzan por palabras claves. Otra característica importante fue que SQL podía ser integrado a otros lenguajes de aplicación o podría ser ejecutado por una terminal interactiva directamente sobre la base de datos relacional.



Unidad II. Modelo Relacional



2. SQL2 (SQL92)

En 1992 fue revisado el estándar por la ANSI y resueltos varios problemas de la primera versión. Entre las mejoras estuvieron:

- * Soporte a caracteres de longitud variable.
- * Expresiones para cambios de tipo de datos.
- * Operador *join*.
- * Manejo de transacciones.
- * Uso de subconsultas.
- * Uso de operadores de unión, intersección y diferencia.

3. SQL3 (SQL99)

Este nuevo estándar introdujo conceptos de la orientación a objetos y el uso de tipos de datos complejos. Hoy en día los principales sistemas manejadores de bases de datos han adoptado este estándar y se han convertido en lo que podemos llamar manejadores objeto-relacionales de bases de datos.

No es claro si las empresas cambiarán su tecnología de bases de datos en un futuro cercano. Además, es innegable que el modelo relacional sigue siendo el más utilizado y el que ha logrado resolver los requerimientos de información más significativos de distintas organizaciones. Por esto es muy importante que como informático conozcas bien este modelo y seas capaz de utilizarlo para resolver problemas de información.



Unidad II. Modelo Relacional



ACTIVIDAD 1

Elabora un mapa conceptual sobre el modelo relacional de bases de datos que incluya todos los conceptos presentados en la unidad.

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma

ACTIVIDAD 2

Define con tus propias palabras en qué consiste el modelo relacional de bases de datos.

Para enviar tu respuesta, pulsa el botón **Editar mi envío**; se mostrará un editor de texto en el cual puedes redactar tu información; una vez que hayas concluido, salva tu actividad pulsando el botón **Guardar cambios**.



Unidad II. Modelo Relacional



ACTIVIDAD 3

A partir de la siguiente relación **EMBARQUE**, determina si cumple con la primera, segunda y tercera forma normal y explica por qué.

idembarque	iddestino	idproducto	fecha	destino	producto
1	100	V1	01/01/2006	España	Nombre: Vino, Precio: 12,000.
1	100	M1	01/01/2006	España	Nombre: Mueble, Precio: 10,000.
2	100	E1	01/02/2006	España	Nombre: Electrodoméstico, Precio: 13,000
2	100	V1	01/02/2006	España	Nombre: Vino, Precio: 12,000.
3	200	V1	01/03/2006	Argentina	Nombre: Vino, Precio: 12,000.
4	100	M1	01/04/2006	España	Nombre: Mueble, Precio: 10,000.
5	200	M1	01/04/2006	Argentina	Nombre: Mueble, Precio: 10,000.

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.



ACTIVIDAD 4

Responde el siguiente cuestionario.

1. ¿Cuántas son las Reglas de COOD?
2. ¿En cuál regla se incluye la operación Manipulación de Datos?
3. ¿Qué regla trata del sistema relacional distribuido?
4. ¿Qué regla trata de la descripción de los datos dentro de la base de datos?
5. ¿Cuál de las reglas trata del valor atómico?

Para enviar tu respuesta, pulsa el botón **Editar mi envío**; se mostrará un editor de texto en el cual puedes redactar tu información; una vez que hayas concluido, salva tu actividad pulsando el botón **Guardar cambios**.

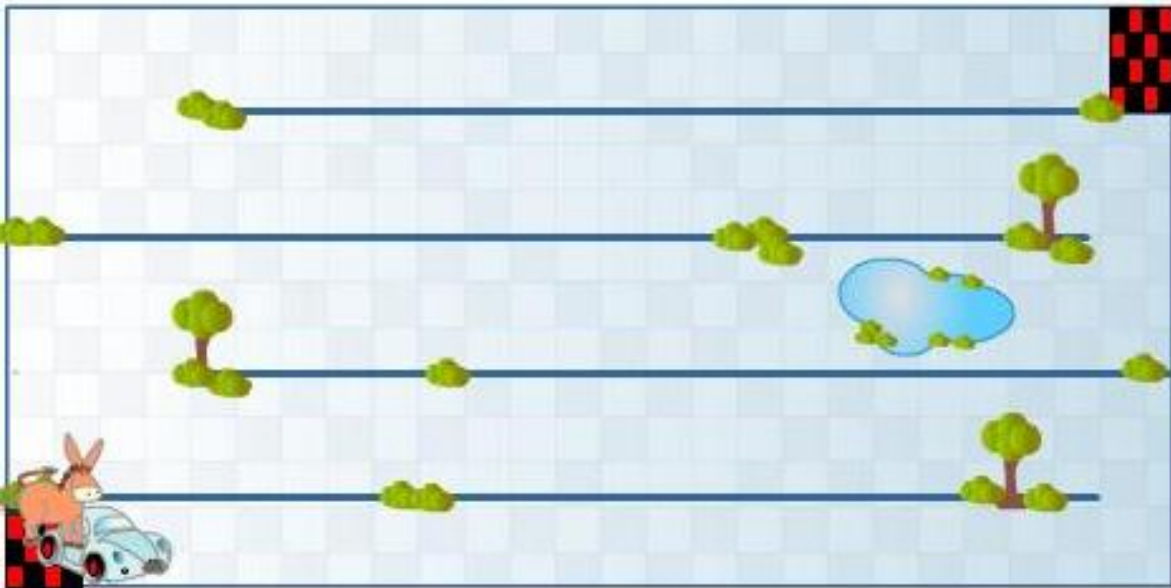


Unidad II. Modelo Relacional



Autoevaluación

Participa en el siguiente Rally. Debes de contestar correctamente para poder avanzar.



1 La 3FN exige que todas las dependencias sean irreducibles.

V () F ()

2 La 4FN exige la eliminación de las dependencias de junta.

V () F ()

3 La regla del acceso garantizado exige que el catálogo de una base de datos sea consultado usando el mismo sublenguaje de datos.

V () F ()

4 La regla de la independencia física de los datos exige que si cambian las tablas o relaciones, los programas de aplicación no se modifiquen.

V () F ()



Unidad II. Modelo Relacional



- 5 La regla del tratamiento sistemático de valores nulos exige que el sistema de bases de datos cuente con un valor distinto al 0, la cadena vacía o el espacio.

V () F ()

- 6 La restricción obtiene un subconjunto de atributos de una relación.

V () F ()

- 7 El producto obtiene aquellos atributos que coinciden en un atributo en común entre las relaciones.

V () F ()

- 8 La normalización tiene por objeto reducir los problemas de redundancia y actualización de datos.

V () F ()

- 9 El cuerpo de una relación es el conjunto de atributos de la misma.

V () F ()

- 10 Una relación tiene 12 propiedades.

V () F ()



Unidad II. Modelo Relacional



Bibliografía básica

Autor	Capítulo	Páginas

Sitios electrónicos

Sitio	Descripción



Unidad II. Modelo Relacional



Lo que aprendí

Elabora un Mapa Conceptual de la Unidad y genera las siguientes tablas empleando comandos del Lenguaje SQL. Introduce dos registros por Tabla.

La Base de Datos se guardará con el Nombre “Biblos”

AUTOR			
ID_AUTOR	NOMB_AUTOR	NACIONALIDAD	ID_LIBRO

LIBRO			
ID_LIBRO	TITULO_LIBRO	ISBN	ID_AUTOR

EDITORIAL			
ID_EDIT	NOMB_EDIT	PAIS_EDIT	ID_LIBRO

Realiza tu actividad en un procesador de textos, guárdala en tu computadora y una vez concluida, presiona el botón **Examinar**. Localiza el archivo, ya seleccionado, presiona **Subir este archivo** para guardarlo en la plataforma.



Unidad II. Modelo Relacional



GLOSARIO

Álgebra relacional

Es el conjunto de operadores que permiten manipular las relaciones (consultar datos). Estos operadores siempre generan nuevas relaciones a partir de otras.

Cálculo relacional

Es posible identificar dos tipos de cálculo relacional: el cálculo de tuplas y el cálculo de dominios

Clave candidata

Es un atributo que identifica de manera única a cada tupla de una relación. Una relación puede tener varias claves candidatas y estas pueden ser simples, formadas por un atributo, o compuestas cuando se forman por varios atributos. A las últimas se les llama superclaves.

Clave principal

Una clave principal es una clave candidata seleccionada para ser la clave principal de la relación. En este caso, a diferencia de las claves candidatas, sólo existe una por relación. También pueden ser superclaves.

Clave foránea

Una clave foránea es un atributo que hace referencia a una clave principal en otra relación. Esta referencia implica que no pueda existir un valor en la clave foránea que no exista primero en la clave principal.

Cuerpo

Es un conjunto de m tuplas en donde cada una es un conjunto de componentes de la forma NombredeAtributo:ValordeAtributo



Unidad II. Modelo Relacional



Dominio

Es el conjunto de valores válidos para un atributo. De esta manera, podríamos decir que cada atributo tendrá asociado siempre un dominio

Modelo jerárquico

Interconecta nodos en una jerarquía estricta de padre e hijos, donde no podía haber relación entre nodos de distintos niveles o entre los del mismo nivel.

Modelo de red

Se trata de una interconexión de nodos mediante apuntadores, sin la restricción del jerárquico, es decir, pueden salir de cada nodo varios arcos apuntando a otros nodos. El modelo distribuido está basado en el modelo relacional o en el orientado a objetos y su principal característica es que la base de datos está fragmentada en una red.

Modelo semiestructurado

Surge ante la necesidad de que cada elemento almacenado cuente con un conjunto de atributos propio, sin necesidad de que este conjunto sea obligatorio como sucede en el modelo relacional. Este tipo de bases de datos se crean con el lenguaje de marcado extensible XML.

Modelo deductivo

Se apoya en un conjunto de datos y reglas de inferencia. Es conocido también como modelo inferencial o modelo basado en la lógica. Las consultas de datos en este modelo son demostraciones de axiomas deductivos.



Unidad II. Modelo Relacional



Modelo relacional

Se fundamenta en una teoría abstracta de datos que toma ciertos aspectos de las matemáticas (teoría de conjuntos y lógica de predicados). De esta manera, podemos definir una relación como un conjunto de tuplas y atributos, y que se compone de dos partes: encabezado y cuerpo.

Normalización

Es el proceso de ajuste de relaciones a ciertas reglas llamadas formas normales. Tiene por objetivo reducir los problemas de redundancia y actualización en las relaciones; su ventaja es que lo hace mediante un procedimiento bien formalizado.

Restricción

Regresa una relación que contiene todas las tuplas de una relación especificada, que satisfacen una condición.



Unidad II. Modelo Relacional



MESOGRAFÍA

Bibliografía básica

Bibliografía complementaria

Sitios electrónicos