



Atender el curso de Creación de base de datos relacionales en un lapso de 1 semana

Prácticas Profesionales

Documento de Visión del Proyecto

Versión **1.0**

Fecha: **20/04/2022**

[NOTA: El texto encerrado entre corchetes y color azul, provee guías al autor para el llenado del documento y deben ser eliminadas para su publicación. Las secciones en color Negro no son susceptibles de cambio. El diseño del documento no deberá sufrir modificaciones. Todo el Texto normal del Documento debe estar en letra "Montserrat Light", de Tamaño 9. El título del documento en tamaño 12 y en negrita. Los Títulos de Secciones en tamaño 9 y negrita.

No se permite el uso de letra itálica ni el uso de corchetes.

NOTA SOBRE LAS PROPIEDADES: Las propiedades del documento deben de actualizarse conforme lo establecido en el estándar de documentación. Al terminar el documento se debe eliminar esta nota.]

Smartydreams™
Ideas_Technology Innovation

Atender el curso de Creación de base de datos relacionales en un lapso de 1 semana

Historial de revisión

<i>Fecha</i>	<i>Versión</i>	<i>Descripción</i>	<i>Autor</i>
20/04/2022	1.0	Reporte del curso de diseño de base de datos relacional	Virginia Castañeda Pérez

Atender el curso de Creación de base de datos relacionales en un lapso de 1 semana

Tabla de contenido

Índice

Introducción.....	4
Lo aprendido.....	5
Mi opinión del curso.....	5
Lo que más me gusto.....	5
Mis apuntes del curso.....	5
Diagrama Entidad relación.....	6
Capturas de pantalla de las notas.....	6
Código SQL.....	10

Atender el curso de Creación de base de datos relacionales en un lapso de 1 semana

Introducción

Realizar el curso de Diseño de base de datos proporcionado por el responsable del programa, el cual se complementó con ejercicios que venían el mismo.

El curso abarco desde:

- ✚ ¿Qué es un dato?
- ✚ Historia de los modelos de base datos
- ✚ Introducción a las bases de datos
- ✚ Modelos de BD
- ✚ El modelo relacional
- ✚ Algebra relacional
- ✚ Normalización
- ✚ Diagrama entidad relación
- ✚ Manejadores de bases de datos
- ✚ SQL
- ✚ Disparadores (Triggers)
- ✚ Vistas
- ✚ Store procedures
- ✚ Consultas y creación de sentencias SQL para BD.

En cada punto anterior mencionado, se vio teoría y aplicación practica en ejemplos. Además, por mi parte realicé notas en archivos .md y los scripts para los ejercicios; estos los subí al repositorio de GitHub, para poder visualizar mejor las notas.

El repositorio: https://github.com/vikecp/Practicas_js/tree/main/2_semana

Atender el curso de Creación de base de datos relacionales en un lapso de 1 semana

Lo aprendido

Durante el curso aprendí conceptos nuevos y/o algunos reforcé el conocimiento. Algunos de los puntos que aprendí fue los modelos de datos antes de que llegara el modelo relacional; la normalización hasta la 5NF que en la mayoría de los casos sólo se normaliza hasta la 3NF. Mejore mis conocimientos con el diagrama entidad relación ya que ya había hecho este tipo de diagramas anteriormente. En cuanto a la sintaxis SQL aprendí nuevas instrucciones, los triggers fue totalmente nuevo para mí, pero fue emocionante poder aprenderlo. Algunas otras cosas fueron repaso para mí.

Mi opinión del curso

Fue un curso completo y con un nivel avanzado, pero si consideraría que los videos hubieran sido un poco más cortos en cuanto a la explicación, que los ejemplos con los ejercicios hubieran sido codeados en el mismo, no sólo llegar a la ejecución, ya que así la persona explicaría a detalle cada sentencia y seria de mejor entendimiento para nosotros.

Lo que más me gusto

Fue el ejemplo con la creación de la DB "Academia" además de todas sus tablas, PK, FK, relaciones y restricciones. Fue un ejemplo muy bueno, además de ejecutar los scripts para crear las vistas, triggers, Store procedures, etc.

Mis apuntes del curso

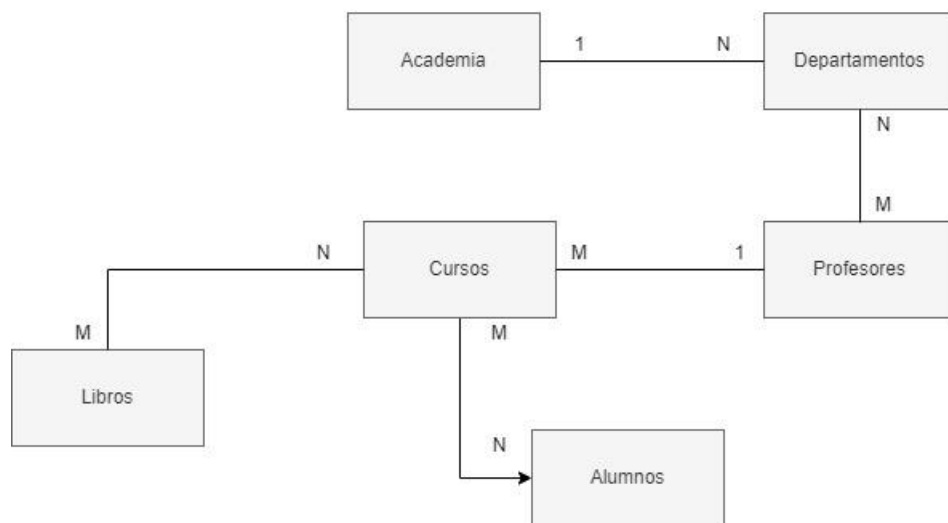
A continuación, adjunto capturas de mis apuntes, porque los hice en archivos .md y su visualización sólo es posible en GitHub o con una herramienta. Todo se puede ver en: El repositorio: https://github.com/vikecp/Practicas_js/tree/main/2_semana

Atender el curso de Creación de base de datos relacionales en un lapso de 1 semana

Diagrama Entidad relación

Diagrama Entidad Relación "Academia"

Virginia Castañeda



Capturas de pantalla de las notas

84 lines (58 sloc) | 1.89 KB
Raw
Blame

Datos

- del latin datum: lo que se dat es la representacion simbolica de un atributo de una variable cuanti o cualitativa.
- Describe una entidad, hecho o momento.
- un dato solo no tiene informacion.
- 3 solo no es nada, debe tener algo 3 gatos, 3 pelotas etc.
- la descripcion que le demos cambia el dato

Metadata: datos que describen al datos

- Se divide en:
- Metadata descriptiva.
- Metada estructural

El dato solo no genera informacion.

Atender el curso de Creación de base de datos relacionales en un lapso de 1 semana

84 líneas (58 sloc) | 1.89 KB

<> [icon] Raw Blame [icon] [icon] [icon]

Que es un BASE DE DATOS

- bd: conjunto, coleccion de informacion organizada, almacenados para su uso, en un mismo contexto
- Datos que estan relacionados entre si que pertenecen a un mismo contexto es una BD.
- Deben estar organizados.
- diferentes tipos de datos y finalidad tamaño

Clasificacion

- Cambio de sus datos:
 - Estaticas: como un libro
 - Dinamicas: E/S de nuevos datos
- Segun el alcance:
 - Personales
 - Locales
 - Mundiales

81 líneas (56 sloc) | 2.5 KB

<> [icon] Raw Blame [icon] [icon] [icon]

Modelos de BD

Historia

- 1960 por parte IBM, manejar todos sus datos sistema para manejo de informacion
- Modelo jerarquico: primera vez se podia relacionar los datos, tenia problemas
- Modelo de redes: modelo complicado dificil de implementar
- 1969: Edgar F. codd ----> le nace la idea de un modelo de bd relacional y genera una revolucion, ven un futuro. soluciona los problemas del modelo jerarquico., Sigue siendo el modelo estandar de la actualidad

Modelo de archivos planos:

- base del almacenamiento.
- no sigue ningun formato
- Separar archivos planos: Formatos
- CSV: separa por una coma
- Lenguajes de marcado: html
- Sin formato.
- Archivos binarios: archivos que no son de texto que se puede leer ni entender

Modelo jerarquico

Atender el curso de Creación de base de datos relacionales en un lapso de 1 semana

81 lines (56 sloc) | 2.5 KB

<> [icon] Raw Blame [icon] [icon] [icon]

Modelo jerarquico

- apoyado por IMB
- datos organizados de forma jerarquica
- Como arbol invertido
- Nodos se conectan por relaciones
- Camino
- Padre e hijo si hay relacion
- Dos nodos del mismo nivel se denomina hermanos
 - Este es el gran problema
 - Todas las relaciones son de 1:1
 - no permite la relacion de N:M
- Formatos como el html y el json usan este modelo

Modelo de redes:

- Busca solucionar l realcion de N:M
- Basado de la estructura de grafos:
- Ser mas flexible que el modelo jerarquico
- se elimina la duplicidad: menor costos, integridad de datos
- Integridad de los datos: mantener y asegurar la exactitud y consistencia durante toda la vidad de los datos.
- hay varias reglas de integridad.
- caida en 1970, por el nuevo modelo relacional.

34 lines (23 sloc) | 894 Bytes

<> [icon] Raw Blame [icon] [icon] [icon]

Modelo Relacional

- Bd: conjunto de relaciones
- relaciones: conjunto de tuplas
- tupla: conjunto de atributos su orden no es importante

Atributos

- Simples:
 - atomicos: edad, ciudad
- Complejos: No-atomicos: se pueden subdividir, depende de las reglas de negocio nombre, direccion: avenida y calle

Atomicidad de los datos

- faciles de: consultar, manejar y asegurar
- Derivabilidad: posible obtener los datos de otra partw mediante un conjunto de operacion

Redundancia

- duplicidad de datos
- Hay dos tipos: Fuerte y debil

consistencia

- calidad de los datos
- Se genera por: errores, reglas poco claras, mala informacion , falta de datos

Atender el curso de Creación de base de datos relacionales en un lapso de 1 semana

54 lines (33 sloc) | 1.58 KB

<> [icon] Raw Blame [icon] [icon] [icon]

Operaciones sobre conjuntos

- **Union:** Dado A y B el conjunto C contiene todos los elementos de A y B = $A \cup B$
- **intersección:** Dado A y B el conjunto C contiene los elementos comunes entre A y B.
- **Complemento:** Elementos que no estan en A
- **Diferencia:** Dado A y B el conjunto C contiene los elementos de A que no esten en B
- **Diferencia simetrica:** Elementos que estan en A o en B, pero no en los dos conjuntos
- **Producto cartesina:** Todos los pares que se puedan formar combinando cada elemento de A con cada elemento de B

Subconjunto

- todo conjunto es subconjunto de si mismo.

Operaciones sobre relacionales

- **Selección:** extraer subconjuntos especificos de una relación

Permutación

- ordenar de forma diferente. Nom, AP = Se puede ordenar de Nom|AP o AP|Nom ----> esto es permutar.
- Permutaciones = Grado!

Union Natural

39 lines (28 sloc) | 1.19 KB

<> [icon] Raw Blame [icon] [icon] [icon]

Normalización

- Poner en orden lo que no estaba
- Ordenar, estandarizar, mejorar nuestras relaciones.
- Que ordenar?----La redundancia.

Proceso secuencial, ciclico y repetitivo a todas y cada una de las entidades del modelo y se aplica un conjunto de reglas bien definidas

Anomalias de insercion

- no se puede insertar datos sin la presencia de otros atributos.
- anomalia de modificación
- anomalia de eliminación

Formas Normales

1. Primera Forma Normal(1NF):

- no debe haber duplad repetidas
- Existencia de una llave primaria
- atributos atomicos

Una relacion esta en (1NF) si, y solo si, no tiene grupos repetitivos

2. Segunda Forma Normal

Una relación esta en (1NF) si, y solo si: 1. Esta en 1NF 2. Todos los atributos dependen de la llave primaria completa - Ningun atributo depende unicamente de una parte de la llave primaria.

Atender el curso de Creación de base de datos relacionales en un lapso de 1 semana

```
41 lines (25 sloc) | 1.19 KB
Manejadores
• herramienta interactiva
• Son programas
• asegurar integridad de la data
• Recuperacion en caso de falla
• control de accesos
• manejo de accesos concurrentes

# componentes importantes en un manejador
1. Data: dueño de la data y el unico que es capaz de manipularla
2. Motor(Database Engine): Son los encargados de acceder y manipular estos datos
3. Meta Data: El esquema de nuestra bd, la representacion de nuestro modelo logico el manejador de bd.

12 Reglas de Codd
• Regla 0 : un DBSM sea debe ser capaz de manejar bd de manera totalmente relacional
• Regla 1: Regla de la informacion: representado a nivel logico: Valores en tablas.
• Regla 2: Acceso garantizado: Combinacion de: nombre de la tabla, nombre PK, nombre de la columna
• Regla 3: Tratameinto de los valores nulos: debe ser soportables.
• Regla 4: Dicionario dinamico en línea.
• Regla 5: Regla del sub-lenguaje de datos amplio:
• Regla 6: Regla de actualizacion de vistas

63 lines (48 sloc) | 1.29 KB
SQL Structured Query Language
• origen 1974 por ibm system R por Donald y Raymond Boyce
• 1986 SQL Standard ANSI
• 1987 SQL Standard ISO

Regla del sublenguaje de datos amplio SQL
• DDL: lenguaje de definicion de datos:
  ◦ Comandos: manejo de la metadata:
    ▪ CREATE: Crear objetos
    ▪ ALTER: modificar
    ▪ DROP: eliminar
• DML: Lenguaje de manipulacion de datos.
  ◦ Comados: para accesar y manipular datos
    ▪ INSERT: agregar
    ▪ UPDATE: modificar
    ▪ DELETE: eliminar
    ▪ TRUNCTE: eliminar todos los datos
    ▪ SELECT: consultar datos
• DCL: lenguaje de control de datos:
  ◦ Comados:
    ▪ GRANT: otorgar permisos
    ▪ REVOKE: quitar permisos
```

Código SQL

Todo se encuentra en el repositorio

Atender el curso de Creación de base de datos relacionales en un lapso de 1 semana

```
19 ---Script para triggers
20
21 USE tempdb;
22
23 --Crear las tablas
24 CREATE TABLE Facturas (Factura INT, Cliente INT, Fecha DATETIME, Total Money Default (0));
25 CREATE TABLE Facturas_Detalles ( Factura INT, Detalle INT, Producto varchar(18), Total money);
26
27 --- Creacion del trigger
28 CREATE TRIGGER Detalles_Modificados
29 ON Facturas_Detalles
30 AFTER UPDATE, INSERT, DELETE
31 AS
32 BEGIN
33     UPDATE Facturas
34     SET      Total = Total + Total_Detalles
35     FROM (
36         SELECT Factura AS Fct, SUM(Total) AS Total_Detalles
37         FROM (
38             SELECT Factura, Total FROM Inserted
39             UNION ALL
40             SELECT Factura , -Total FROM Deleted
41         ) T
42         GROUP BY Factura
43     ) A
44     WHERE Factura = Fct;
45 END;
46
47
48 --- Comprobar si funciona el trigger
49
50 Select * from Facturas;
51 Select * from Facturas_Detalles;
52 --- Crear datos facturas
53 Insert into Facturas (Factura, Cliente, Fecha)
54 Values (1,10,'20171005'),
55        (2,15,'20171005');
56
57 --- Creacion de un detalle
```

Atender el curso de Creación de base de datos relacionales en un lapso de 1 semana

```
319 lines (260 sloc) | 7.23 KB
1  -----
2  -- Creacion de la bd --
3  -----
4  CREATE DATABASE Academia;
5
6  -- Usar la bd --
7
8  USE Academia;
9
10 --- Crear tabla de paises ---
11
12 CREATE TABLE Paises
13 (
14   Cod_Pais char(2) PRIMARY KEY CHECK (LEN(Cod_Pais)=2),
15   Nombre varchar(50) NOT NULL,
16   Cod_ISO3 char(3) NOT NULL UNIQUE CHECK (LEN(Cod_ISO3)=3),
17   Cod_Telefonico smallint
18 );
19
20 -- Crear tabla estados ----
21
22 CREATE TABLE Estados
23 (
24   Cod_Estado char(2) PRIMARY KEY,
25   -- Creamos la restricción como un objeto separado y le damos nombre
26   CONSTRAINT Len_Estado CHECK (LEN(Cod_Estado)=2),
27   Cod_Pais char(2) FOREIGN KEY REFERENCES Paises (Cod_Pais)
28                               ON UPDATE CASCADE
29                               ON DELETE CASCADE,
30   Nombre varchar(50) NOT NULL,
31   Cod_Telefonico smallint
32 );
33
34
35 --- Create table academias ----
36
37 CREATE TABLE Academias
```