



**UAEM**

Universidad Autónoma  
Del Estado de México

Centro Universitario  
UAEM Texcoco

**Universidad Autónoma del Estado de México**

**Centro Universitario Texcoco**

**Nombre del Programa Educativo de Licenciatura**

**Informática Administrativa**

***Manual de Prácticas de la Unidad Aprendizaje  
Bases de Datos Relacionales***

**Elaboró: Dr. Adían Trueba Espinosa      Fecha: 27 septiembre de 2019**

**H. Consejo Académico**

**H. Consejo de Gobierno**

**Fecha de  
aprobación**

**30 de septiembre de 2019**



## ÍNDICE

### PRIMERA SECCIÓN

<b>Lineamientos del Laboratorio de la UAEM</b>	<b>4</b>
<b>Lineamientos del laboratorio del Centro Universitario UAEM Texcoco</b>	<b>12</b>
<b>Presentación</b>	<b>14</b>
<b>Sistema de Evaluación</b>	<b>16</b>

### SEGUNDA SECCIÓN

#### Prácticas de laboratorio

<b>Práctica número. 1 Implementación de bases de datos físicas</b>	<b>18</b>
<b>Práctica número. 2 Sentencias CRUD</b>	<b>23</b>
<b>Práctica núm. 3 Condicionantes para crear filtros</b>	<b>29</b>
<b>Práctica núm. 4 Consultas a tablas múltiples</b>	<b>39</b>
<b>Referencias</b>	<b>48</b>



**UAEM**

Universidad Autónoma  
Del Estado de México

Centro Universitario  
UAEM Texcoco

---

## **PRIMERA SECCIÓN**

### **Lineamientos del laboratorio**



## REGLAMENTO DE LAS TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIONES DE LA UAEM

### CAPÍTULO PRIMERO DISPOSICIONES GENERALES

**Artículo 1.** El presente Reglamento tiene por objeto establecer las normas básicas para regular el desarrollo y aplicación de las Tecnologías de la Información y Comunicaciones en la Universidad Autónoma del Estado de México.

**Artículo 2.** Las disposiciones contenidas en el presente ordenamiento serán de observancia obligatoria para los Organismos Académicos, Centros Universitarios, Unidades Académicas Profesionales, Planteles de la Escuela Preparatoria y Dependencias Académicas y Administrativas de la Universidad Autónoma del Estado de México.

**Artículo 3.** Las Tecnologías de la Información y Comunicaciones, son consideradas en la Universidad Autónoma del Estado de México como las tecnologías necesarias para la gestión y transformación de la información, vinculadas al uso del software y dispositivos, herramientas, equipos y componentes electrónicos del campo de las computadoras y las comunicaciones.

**Artículo 4.** Las Tecnologías de la Información y Comunicaciones estarán destinadas prioritariamente a la atención y apoyo de las funciones sustantivas de la Universidad Autónoma del Estado de México, facilitando y elevando la calidad del proceso educativo, vinculándose al mismo tiempo a la actualización y automatización de los servicios de la gestión universitaria.

**Artículo 5.** Para los efectos del presente Reglamento entenderá por:

- I. Ámbitos universitarios. Los Organismos Académicos, Centros Universitarios, Unidades Académicas Profesionales, Planteles de la Escuela Preparatoria y Dependencias Académicas y Administrativas de la UAEM;
- II. DSC. La Dirección de Servicios de Cómputo;
- III. PDATIC. El Programa de Desarrollo y Aplicación de las Tecnologías de la Información y Comunicaciones de la UAEM;
- IV. Responsable(s) de las TIC. El personal universitario encargado del manejo del área de las TIC en el ámbito universitario correspondiente;
- V. Secretaría. La Secretaría de Administración de la Universidad Autónoma del Estado de México;
- VI. TIC. Las Tecnologías de la Información y Comunicaciones; y



VII. UAEM. La Universidad Autónoma del Estado de México.

**Artículo 6.** La Secretaría expedirá las disposiciones jurídicas derivadas de este Reglamento, que regulen el desarrollo y aplicación de las TIC.

Las disposiciones jurídicas deberán ser integradas en forma conjunta con el Abogado General de la UAEM.

**Artículo 7.** La Secretaría, a través de la DSC señalará las condiciones de organización e infraestructura necesarias para la creación, desarrollo y funcionamiento de las áreas de las TIC, en los diferentes ámbitos universitarios.

**Artículo 8.** La Secretaría, a través de la DSC representará a la UAEM ante los organismos vinculados con las TIC, a nivel nacional e internacional.

## CAPÍTULO SEGUNDO DE LA PLANEACIÓN DE LAS TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIONES

**Artículo 9.** El desarrollo y aplicación de las TIC responderá a una planeación estratégica que impulse el cumplimiento de los objetivos institucionales, observando lo establecido en el Plan General de Desarrollo, el Plan Rector de Desarrollo Institucional y demás instrumentos de planeación aprobados para tal efecto.

**Artículo 10.** El PDATIC es el instrumento que define las directrices, estrategias y formas de seguimiento y evaluación del desarrollo y aplicación de las TIC en la Institución, posibilitando actuar con orden y eficacia en la toma de decisiones y acciones que se lleven a cabo.

Su integración, ejecución y seguimiento estará a cargo de la Secretaría, a través de la DSC y será evaluado anualmente en forma coordinada con la Secretaría de Planeación y Desarrollo Institucional de la UAEM.

**Artículo 11.** Los ámbitos universitarios participarán en la integración del PDATIC mediante un plan de trabajo anual elaborado conforme al procedimiento que señale la DSC.

**Artículo 12.** Las acciones y proyectos que no estén comprendidos en los planes de trabajo anuales, deberán ser analizados y autorizados por la Secretaría, a través de la DSC, previa solicitud y justificación por escrito del ámbito universitario, mismo que deberá observar lo establecido en el PDATIC, así como las políticas de racionalización y optimización de recursos de la UAEM.



## CAPÍTULO TERCERO DE LA INFRAESTRUCTURA DE LAS TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIONES

**Artículo 13.** La infraestructura de las TIC se integra por los equipos de cómputo y de comunicaciones, cableado y software necesarios en los ámbitos universitarios.

El equipo de cómputo se integra por los componentes utilizados para manipular información en forma automática, como computadoras, impresoras, scanner, video proyectores y otros.

El equipo de comunicaciones se integra por los componentes utilizados para la interconexión de redes y de telefonía, como switches, conmutadores, convertidores, módems y otros, excepto los relacionados con los servicios de comunicación masiva de radio y televisión.

**Artículo 14.** La DSC determinará el equipo de comunicaciones y cableado necesario en los diferentes ámbitos universitarios y llevará a cabo su instalación y administración.

El equipo de cómputo y software necesarios para el desarrollo y aplicación de las TIC, será propuesto por el ámbito universitario y autorizado por la DSC, la cual efectuará, en su caso, su instalación, supervisión y mantenimiento.

**Artículo 15.** La DSC es la dependencia universitaria encargada del desarrollo y administración de las redes de comunicaciones de la UAEM, que incluyan datos, voz o video, por lo que establecerá los requerimientos necesarios para su configuración, acceso y uso.

**Artículo 16.** Los responsables de las TIC, tendrán las siguientes obligaciones respecto a la infraestructura de las TIC:

- I. Vigilar que la infraestructura sea utilizada en el cumplimiento de los objetivos para los que fue asignada y, en su caso, solicitar a la DSC una reorientación en su uso, para un mejor aprovechamiento;
- II. Supervisar que los bienes porten la etiqueta de control patrimonial;
- III. Elaborar un inventario de los bienes, manteniéndolo actualizado e informando de cualquier cambio a la DSC y a la dependencia universitaria encargada del control patrimonial;
- IV. Supervisar la infraestructura y mantenerla en óptimas condiciones de funcionamiento;
- V. Tomar las medidas de seguridad necesarias para su adecuado resguardo y protección; y
- VI. Cumplir con las instrucciones de su titular, así como las recomendaciones emitidas por la DSC.



**Artículo 17.** La DSC contará con un catálogo del software que estará a disposición de los ámbitos universitarios.

El software que se instale en cualquier equipo propiedad de la UAEM deberá ser autorizado por la DSC y contar con la licencia correspondiente.

En caso de que se instale software sin la licencia respectiva y que con ello se cause daño o perjuicio a la UAEM, la responsabilidad recaerá en aquella persona que tenga bajo su resguardo el equipo.

**Artículo 18.** Los usuarios de la infraestructura de las TIC tienen prohibido copiar, alterar o borrar el software que se encuentre instalado en los equipos de las TIC, así como, desarrollar alguno nuevo sin previa autorización del titular del ámbito universitario y de la DSC.

**Artículo 19.** El equipo y software de las TIC estará bajo resguardo de las personas a quienes se les asigne, siendo además responsables de su uso y mantenimiento.

**Artículo 20.** Los usuarios deberán utilizar la infraestructura de las TIC, exclusivamente para la realización de sus trabajos académicos o administrativos.

**Artículo 21.** Los usuarios de la infraestructura de las TIC que causen daños o perjuicios a la misma, deberán repararlos o reponerlos cuando resulten responsables.

**Artículo 22.** La DSC participará con la Dirección de Obra Universitaria en la elaboración de los proyectos de obra nueva, con el objeto de incluir los requerimientos de infraestructura de las TIC.

**Artículo 23.** Los ámbitos universitarios y la Dirección de Obra Universitaria, deberán hacer del conocimiento de la DSC cualquier cambio que afecte o amplíe la infraestructura de las TIC, derivado de remodelaciones o modificaciones a las instalaciones universitarias, con la finalidad de planear y coordinar las acciones necesarias.

## **CAPÍTULO CUARTO DE LA CAPACITACIÓN Y DIFUSIÓN**

**Artículo 24.** La capacitación para el desarrollo y aplicación de las TIC se llevará a cabo mediante acciones y proyectos señalados en el PDATIC, que serán acordes a las directrices del desarrollo institucional.

Las acciones y proyectos de capacitación serán difundidos oportunamente por la DSC en los ámbitos universitarios.

**Artículo 25.** Los responsables de las TIC deberán ser capacitados permanentemente por la DSC, con la finalidad de que cuenten con los conocimientos y habilidades necesarias para el desarrollo de las actividades a su cargo.



**Artículo 26.** Los titulares de los ámbitos universitarios y los responsables de las TIC, deberán promover a través de los servicios que presta la DSC, la capacitación de aquellas personas que hacen uso de las TIC, con la finalidad de que sean utilizadas correctamente.

**Artículo 27.** La DSC integrará y difundirá la información relacionada con las TIC que considere necesaria, en los ámbitos universitarios.

## **CAPÍTULO QUINTO DE LOS SERVICIOS Y LA DESCONCENTRACIÓN**

**Artículo 28.** Los servicios de las TIC comprenderán: telefonía; redes de comunicaciones; instalación, administración y mantenimiento; asesoría y soporte; capacitación; seguridad informática; desarrollo de software; procesamiento de información; y otros que se establezcan.

**Artículo 29.** La DSC se encargará de recibir, analizar y, en su caso, autorizar las solicitudes de servicios de las TIC, las cuales deberán tramitarse a través del responsable de las TIC, conforme a los procedimientos establecidos.

**Artículo 30.** En caso de que los ámbitos universitarios requieran servicios especiales relacionados con la infraestructura de las TIC, presentarán su solicitud a la DSC, la cual efectuará las gestiones necesarias ante empresas o instituciones externas.

**Artículo 31.** La desconcentración de los servicios de las TIC, deberá promoverse por la DSC conjuntamente con los ámbitos universitarios, mediante la formulación de estrategias que se incluirán en el PDATIC.

**Artículo 32.** La Secretaría podrá crear dependencias de la DSC para una adecuada y oportuna prestación de los servicios de las TIC, observando la regionalización que presenta la UAEM y detectando aquellas zonas en las que se considere propicio establecerlas para cubrir en forma permanente las necesidades de la región.

## **CAPÍTULO SEXTO DE LA SEGURIDAD INFORMÁTICA**

**Artículo 33.** La seguridad informática consiste en salvaguardar la integridad, disponibilidad y confidencialidad de la infraestructura y servicios de las TIC y de la información que en ellas se maneja y transmite.



**Artículo 34.** La DSC señalará las medidas y los mecanismos de seguridad necesarios y, en su caso, podrá implantarlos o llevar a cabo su instalación.

Los usuarios de la infraestructura de las TIC serán responsables del uso y aplicación de las medidas y mecanismos de seguridad.

**Artículo 35.** El usuario no podrá acceder sin permiso a la infraestructura y servicios relacionados con las TIC de la UAEM locales o remotos, en los que se requiera autorización.

**Artículo 36.** La información académica o administrativa de la UAEM, que por su naturaleza sea considerada como confidencial, deberá tener un manejo especial y sólo podrá tener acceso a ella el personal autorizado.

**Artículo 37.** En caso de violar la seguridad de acceso establecida en el artículo 34 o acceder sin autorización a la información señalada en el artículo 35 del presente Reglamento, el usuario incurrirá en faltas a la responsabilidad universitaria.

**Artículo 38.** La UAEM se reserva el derecho de consultar en cualquier momento la información almacenada o transmitida en los bienes de su propiedad, relacionados con las TIC.

El acceso a la información señalada en el párrafo anterior, podrá llevarse a cabo por la DSC, en caso justificado y con la autorización del titular del ámbito universitario, misma que se realizará en presencia de la persona a quien se le haya asignado el equipo.

**Artículo 39.** La DSC podrá supervisar en cualquier momento la infraestructura de las TIC de los ámbitos universitarios, llevando a cabo las medidas correctivas ante cualquier irregularidad o desperfecto.

## ARTÍCULOS TRANSITORIOS

**Primero.** El presente Reglamento entrará en vigor a partir del día de su expedición.

**Segundo.** Este ordenamiento deberá ser publicado en el Órgano Oficial Informativo de la Universidad "Gaceta Universitaria".

**Tercero.** El Programa de Desarrollo y Aplicación de las Tecnologías de la Información y Comunicaciones de la UAEM, deberá quedar integrado conforme a las disposiciones señaladas en el presente Reglamento, en un plazo no mayor de seis meses a partir de la expedición de este ordenamiento.



**PUBLICACIONES EN LA “GACETA  
UNIVERSITARIA”  
EXPEDICIÓN**

**APROBACIÓN:**  
día 28 de

Por el Consejo Universitario en Sesión Ordinaria, celebrada el  
Noviembre de 2007

**PUBLICACIÓN:**  
XXIV

Gaceta Universitaria Núm. 152, Febrero 2008, Época XII, Año

**VIGENCIA:**

28 de Noviembre de 2007



**UAEM**

Universidad Autónoma  
Del Estado de México

Centro Universitario  
UAEM Texcoco

---

## **Lineamientos del laboratorio Particulares del Centro Universitario UAEM Texcoco**



## Universidad Autónoma del Estado de México

## Centro Universitario UAEM Texcoco

REGLAMENTO PARA ACCESO Y USO DE LOS LABORATORIOS Y SALAS DE CÓMPUTO

- 1.- Los servicios de sala de uso general, laboratorios de microcomputadoras y especializados son para uso exclusivo de horas clase y procedimientos institucionales que los requieran.
- 2.- El registro de usuarios de sala de uso general, laboratorios de microcomputadoras y especializados para horas clase y procedimientos institucionales es de carácter obligatorio para el personal docente así como para sus alumnos.
- 3.- Queda estrictamente prohibido introducir bebidas y/o alimentos en los laboratorios de cómputo.
- 4.- Queda estrictamente prohibido fumar dentro de las instalaciones de los Laboratorios de Cómputo.
- 5.- Los objetos personales (mochilas, bolsas, etc.) de los usuarios no deben colocarse sobre la mesa del equipo de cómputo.
- 6.- Si hay necesidad de instalar software específico, deberá solicitarse a los RTIC's (Responsable de Tecnologías de la Información y Comunicaciones) con 5 días hábiles de anticipación, proporcionando lo requerido (software, licencia en caso de no ser software libre y método de instalación en caso de requerir alguna configuración especial).
- 7.- Toda información deberá ser almacenada en alguna unidad de almacenamiento externo propio del usuario, ya que los equipos cuentan con un freeze que al reiniciar el equipo restaura su configuración anterior borrando cambios o información guardada en dicho equipo.
- 8.- Por ningún motivo se deshabilitará el freeze de los laboratorios.
- 9.- Cada equipo solo podrá ser utilizado por dos personas como máximo.
- 10.- Por ningún motivo se permitirá el acceso a los alumnos al Laboratorio sin la presencia de su profesor.
- 11.- El profesor deberá permanecer dentro del espacio durante su estancia hasta que el último alumno se retire o finalice la sesión de laboratorio.
- 12.- Queda prohibido cambiar la configuración de las computadoras (tapices, colores de pantalla, punteros, password, sistema operativo, etc.) y/o desconectar el equipo y sus componentes.



13.-Toda persona que sea sorprendida sustrayendo accesorios, equipamiento, mobiliario, cambiando configuraciones y/o desconectando equipo de cómputo, será presentada ante la autoridad competente.

14.-El uso de internet en el laboratorio deberá ser requerido y justificado al iniciar la sesión por el profesor y este será de uso estrictamente académico por lo tanto no se podrá acceder a ningún tipo de chat, páginas de contenido pornográfico, snuff, o páginas de descarga, música, videos, redes sociales, deportes y/o juegos a menos que el programa de estudios lo especifique.

15.-La tolerancia de acceso a los laboratorios es de 15 minutos después de la hora marcada, en caso de que haya alguna solicitud de uso de laboratorio, este será asignado por ausencia de grupo o profesor.

16.-En caso de existir algún desperfecto, daño o faltante con respecto a equipo de cómputo, mobiliario o infraestructura de sala de uso general y Laboratorios, deberá de ser notificado al RTIC y/o personal a cargo por parte del profesor para levantar el reporte y dar el seguimiento necesario a este.

17.-Es responsabilidad propia del usuario todo objeto olvidado dentro del

laboratorio. 18.-Es responsabilidad del profesor mantener el orden dentro del

laboratorio.

19.-El acceso a los laboratorios será proporcionado única y exclusivamente por el RTIC y/o el personal a cargo.

20.-Es responsabilidad del profesor solicitar y entregar la(s) listas de registro debidamente llenadas (una lista por laboratorio), así como notificar de inmediato cualquier faltante, anomalía o desperfecto encontrado o suscitado durante su sesión, de lo contrario se hará responsable al grupo anterior al momento del reporte.

21.- El usuario deberá apagar el equipo de cómputo al terminar la sesión de clases.

22.- El profesor deberá cerrar la(s) puertas del Laboratorio colocando seguro y notificar a la persona a cargo de su salida la cual deberá ser con al menos cinco minutos de anticipación de la hora límite de su clase para que el Laboratorio sea revisado por el RTIC y/o el personal a cargo.

NOTA: En caso de incumplir el presente reglamento, se girará el reporte a las autoridades universitarias correspondientes.



## Presentación

Una de las principales actividades del Licenciado en Informática Administrativa es la elaboración de bases de datos relacionales desde su diseño hasta la administración por lo que, cuyas bases deben ser adquiridas en su formación. La administración de bases de datos relacionadas como una parte de la informática, evoluciona continuamente, sin embargo la administración de las bases de datos es uno de los conocimientos base en la construcción de aplicaciones de mediana y alta complejidad. Esta unidad de aprendizaje pretende introducir al alumno en el manejo y almacenamiento de datos por medios electrónicos y con el uso de manejadores de bases de datos relacionales de actualidad.

Las prácticas están organizadas para aprender el lenguaje SQL de forma paulatina, considerando el contenido temático de la unidad de aprendizaje **BASES DE DATOS RELACIONALES**.

La práctica uno aborda el lenguaje de definición de datos (LDD) que es donde se aprende a crear las bases de datos, tablas, definición de datos con tipo y longitud y las relaciones entre las tablas. Esta actividad está asociada a la unidad de competencia IV. Donde se requiere implementar una base de datos en un medio electrónico.

Las practicas 2,3,4 están asociadas a la unidad de competencia V, donde se deben adquirir las competencias para manipulación de datos a partir del Lenguaje de Manipulación de Datos (LMD).

Particularmente la práctica dos considera las sentencias **Select, Insert, Delete y Update**, que tiene el objetivo de insertar modificar y eliminar datos en una base de datos.



La práctica tres tiene el objetivo de estimular el aprendizaje para la creación de filtros para obtener consultas específicas considerando los datos de una tabla y que después se pueden generalizar en otras consultas más complejas

La práctica cuatro permitirá fortalecer el conocimiento para la consulta con dos más tablas, usando las relaciones que hay entre atributos llave y llaves foráneas y después con el mismo fin el uso de la sentencia JOIN.



## Sistema de Evaluación

Todas las prácticas serán implementadas con el sistema manejador de bases de datos MYSQL, mediante consola o una interface gráfica, a fin de tener un dominio total del manejador de bases de datos.

La evaluación será a partir de planteamiento de sentencias del Lenguaje de Definición de Datos (LDD), así como, del Lenguaje de Manipulación de Datos (LMD) gravadas en un archivo de texto, para que al ser evaluado se pueda ejecutar en la consola del manejador de bases de datos y demostrar que las consultas se ejecuten adecuadamente.

La escala de evaluación se pone en el siguiente rubrica:

10-9	8-7	6	5 o menos
Presentar un impreso sin faltas de ortografía.  Caratula de presentación con los datos que identifique la actividad y participantes.	Presentar un impreso con algunas faltas de ortografía.  Caratula de presentación con los datos incompletos que identifique la actividad y participantes.	Presentar un impreso con faltas de ortografía en exceso.  Caratula de presentación con los datos incompletos que identifique la actividad y participantes.	Presentar un impreso con faltas de ortografía en exceso.  Sin presentar caratula de presentación con los datos que identifiquen la actividad y participantes.
Planteamiento de la sentencias SQL con un semántica correcta, distinguiendo todas las palabras reservadas en mayúsculas	Planteamiento de la sentencias SQL con un semántica correcta sin distinguir algunas palabras reservadas con mayúsculas	Planteamiento de la sentencia SQL con un semántica correcta sin distinguir las palabras reservadas con mayúsculas	Planteamiento de la sentencia SQL sin un semántica correcta y no se distinguen las palabras reservadas con mayúsculas
Que la consulta se ejecute y se obtengan los resultados esperados	Que la consulta se ejecute y se obtengan los resultados esperados	Que la consulta se ejecute y no se obtengan todos resultados esperados	Que la consulta no se ejecute y no obtengan resultados



**UAEM**

Universidad Autónoma  
Del Estado de México

Centro Universitario  
UAEM Texcoco

---

## **SEGUNDA SECCIÓN**

### **Prácticas de Laboratorio**



# Práctica número. 1

## Nombre de la Práctica

### Implementación de bases de datos físicas (Lenguaje de Definición de Datos)

## Introducción

Aprender la implementación y funcionamiento de una base de datos relacional sirve para desarrollar las capacidades del uso de los elementos teóricos adquiridos del álgebra y cálculo relacional.

La implementación de una base de datos se hace por medio del Lenguaje de Definición de datos (LDD). La herramienta que se utiliza es un Sistema Gestor de Bases de Datos (SGBD) relacional, donde se puedan implementar bases de datos físicas, usando el Lenguaje de Estructurado de Consultas (SQL)

SQL (Structured Query Language) es un lenguaje diseñado para almacenar, manipular y recuperar datos almacenados en bases de datos relacionales. La primera encarnación de SQL apareció en 1974, cuando un grupo de IBM desarrolló el primer prototipo de una base de datos relacional. Relational Software (luego se convirtió en Oracle) lanzó la primera base de datos relacional comercial.

Existen estándares para SQL. Sin embargo, el SQL que puede utilizarse en cada uno de los principales SGBD actuales viene en distintas formas. Esto se debe a dos razones: 1) el estándar SQL es bastante complejo, y no es práctico implementar el estándar completo, y 2) cada proveedor de base de datos necesita una forma de diferenciar su producto de otros.

Existen tres tipos de comandos SQL:

Los DDL (Data Definition Language) que permiten crear y definir nuevas bases de datos, atributos e índices.



Los DML(Data Manipulation Language) que permiten generar consultas para ordenar, filtrar y extraer datos de la base de datos.

Los DCL(Data Control Language) que se encargan de definir los permisos sobre los datos

El lenguaje de definición de datos (en inglés Data Definition Language, o DDL), es el que se encarga de la modificación de la estructura de los objetos de la base de datos. Incluye órdenes para modificar, borrar o definir las tablas en las que se almacenan los datos de la base de datos. Existen cuatro operaciones básicas: CREATE, ALTER, DROP y TRUNCATE. Ya vistas en clase

## Objetivo

Aprender a la implementación física e una base de datos a partir del lenguaje de definición de datos con SQL

## Unidad de Competencia a ser fortalecida IV

### Desempeño

Identificar la implementación de una base de datos relacional

Realizar la implementación de base de datos mediante aun SGBD

### Producto a obtener

Implementación de base de datos en un medio electrónico

### Materiales y software



Laboratorio de cómputo  
Equipo de cómputo  
Sistema operativo Windows o Linux  
MySQL  
Manual de MySQL

## Ejemplos

### CREATE

Ejemplo (crear una tabla):

```
CREATE TABLE Empleado (  
id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,  
Nombre VARCHAR(50),  
Apellido VARCHAR(50),  
Direccion VARCHAR(255),  
Ciudad VARCHAR(60),  
Telefono VARCHAR(15),  
Peso VARCHAR (5),  
Edad (2),  
Actividad Específica (100),  
idCargo INT);
```

### ALTER

Ejemplo (agregar columna a una tabla):

```
# ALTER TABLE 'NOMBRE_TABLA' ADD NUEVO_CAMPO INT;  
# ALTER TABLE 'NOMBRE_TABLA' DROP COLUMN NOMBRE_COLUMNNA;
```

### DROP

Ejemplo:

```
# DROP TABLE 'NOMBRE_TABLA';  
# DROP SCHEMA 'ESQUEMA';
```



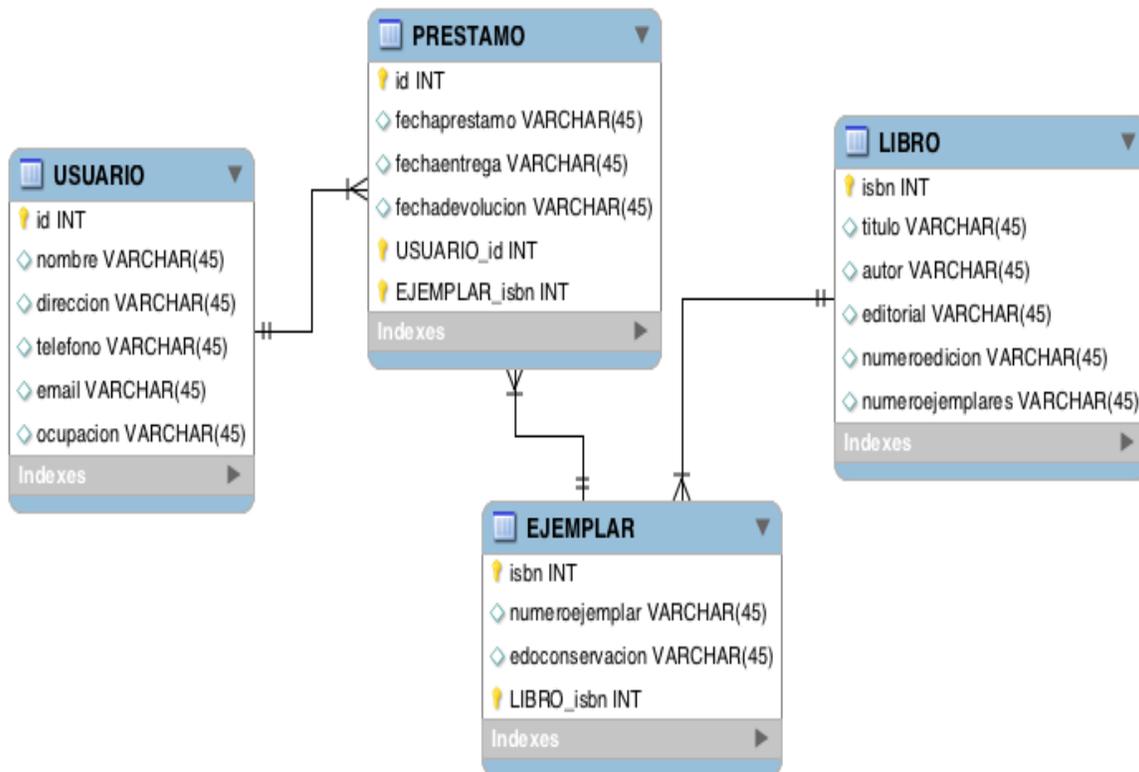
```
# DROP DATABASE 'BASEDATOS';
```

### TRUNCATE

```
# TRUNCATE TABLE 'NOMBRE_TABLA';
```

## Metodología y/o procedimiento

A partir del siguiente esquema de bases de datos



- a) Implementar la base de datos física en el sistema gestor de bases de datos MySQL.

Use las sentencias del LMD

**CRÉATE DATABASE  
CREATE TABLES**

Considere que las llaves primarias deben ser auto incrementables.

Deben de tener las restricciones de borrar y actualizar en cascada.

b) Haga una copia de la base de datos y emplee la sentencia ALTER para:

Borrar el atributo ocupación de la tabla usuario.

Agregar los atributos Apellido paterno y materno de la tabla usuario.

Renombrar el atributo edoconservacion de la tabla ejemplar por Estado\_De\_Conservacion.

Cambiar la llave primaria de la tabla Libro isbn por ISBN con el tipo de datos Varchar con una longitud de 40 caracteres

Agregar un índice a la tabla usuario del atributo teléfono, requiere cambiar el tipo de datos a numérico.

Borrar la llave primaria de la tabla ejemplar

Borrar la tabla libro

## Resultados esperados

Scrip de la base de datos original

Scrip de la base de datos modificada

Sentencias planteadas y utilizadas en el block de notas

Copia de las pantallas que demuestren el resultado de cada una de las sentencias usadas

Describa el uso de cada sentencia considerando la semántica del SQL



# Práctica número. 2

## Nombre de la Práctica

### Sentencias CRUD

### Lenguaje de manipulación de datos DML (Data Manipulation Language)

### PARTE I

## Introducción

Lenguaje de Manipulación de Datos (Data Manipulation Language, DML) es un lenguaje proporcionado por los sistemas gestores de bases de datos, que permite a los usuarios, llevar a cabo las tareas de consulta o modificación de los datos contenidos en las Bases de Datos del Sistema Gestor de Bases de Datos.

En general a las operaciones básicas de manipulación de datos que se realizan con SQL se les denomina **operaciones CRUD** (de **C**reate, **R**ead, **U**ppdate and **D**elete, o sea, *Crear, Leer, Actualizar y Borrar*, en español sería CLAB, pero no se usa). Lo verás utilizado de esta manera en muchos sitios, así que hay que aprenderse ese acrónimo.

El lenguaje de manipulación de datos más popular hoy día es SQL, usado para recuperar y manipular datos en una base de datos relacional. Otros ejemplos de DML son los usados por bases de datos IMS/DL1, CODASYL.

## Elementos del lenguaje de manipulación de datos

### Select, Insert, Delete y Update

#### Clasificación de los DML

Se clasifican en dos grandes grupos:

*Lenguajes procedimentales.* En este tipo de lenguaje el usuario da instrucciones al sistema para que realice una serie de procedimientos u operaciones en la base de datos para calcular un resultado final.



*Lenguajes de consulta* no procedimentales En los lenguajes no procedimentales el usuario describe la información deseada sin un procedimiento específico para obtener esa información.

**SELECT** Es utilizado para consultar registros de la base de datos que satisfagan un criterio determinado

**INSERT** Se utiliza para cargar lotes de datos en la base de datos en una única operación

**UPDATE** Sirve para modificar los valores de los campos y registros especificados Utilizado para modificar las tablas agregando campos o cambiando la definición de los campos

**DELETE** Es Usado para eliminar registros de una tabla

## Objetivo

Aprender a insertar, modificar y borrar datos en con el lenguaje de manipulación de datos con SQL, en un gestor de bases de datos

## Unidad de Competencia a ser fortalecida V

### Desempeño

Realizar la implementación de base de datos mediante aun SGBD

Insertar datos en una base de datos

Actualizar datos en una base de datos

Borrara datos en una base de datos

### Producto a obtener

Implementación de base de datos en un medio electrónico e insertar, borrar y actualizar datos en una base de datos



## Materiales y software

Laboratorio de cómputo  
Equipo de cómputo  
Sistema operativo Windows o Linux  
MySQL  
Manual de MySQL

## Ejemplos

### *INSERT*

Una sentencia INSERT de SQL agrega uno o más registros a una (y sólo una) tabla en una base de datos relacional.

Forma básica:

```
# INSERT INTO 'tabla' ('columna1', ['columna2,... ']) VALUES ('valor1', ['valor2,... '])
```

Las cantidades de columnas y valores deben ser iguales. Si una columna no se especifica, le será asignado el valor por omisión. Los valores especificados (o implícitos).

Si ocurre un error de sintaxis o si alguna de las restricciones es violada, no se agrega la fila y se devuelve un error.

Ejemplo:

```
# INSERT INTO usuario  
(nombre, dirección, teléfono, email, ocupacion)  
VALUES ('Roberto Jeldrez', 'Aldama 1100', '4886850', 'rj@gmail.com', 'Carpintero');
```

Cuando se especifican todos los valores de una tabla, se puede utilizar la sentencia acortada:



```
# INSERT INTO ''VALUES (''valor1'', [ ''valor2,...'' ])
```

Ejemplo:

(Asumiendo que 'nombre' y 'teléfono' son las únicas columnas de la tabla 'agenda'):

```
# INSERT INTO agenda VALUES ('Alma Altamirano', '595473968');
```

Ejemplo:

(Inserto valores alumno pepe en la materia matemáticas a la tabla cursos):

```
INSERT INTO ''cursos'' (''alumno'', ''materia'') VALUES (''pepe'', ''matemáticas');
```

## UPDATE

Una sentencia UPDATE de SQL es utilizada para modificar los valores de un conjunto de registros existentes en una tabla.

Ejemplo:

```
# UPDATE mi_tabla SET campo1 = 'nuevo valor campo1' WHERE campo2 = 'N';
```

Ejemplo (modifico la materia donde el alumno sea Roberto):

```
# UPDATE ''curso'' SET ''materia''= ''matemáticas'' WHERE ''alumno''= ''Roberto';
```

## DELETE



Una sentencia DELETE de SQL borra uno o más registros existentes en una tabla.

Forma básica:

```
# DELETE FROM 'tabla' WHERE 'columna1' = 'valor1'
```

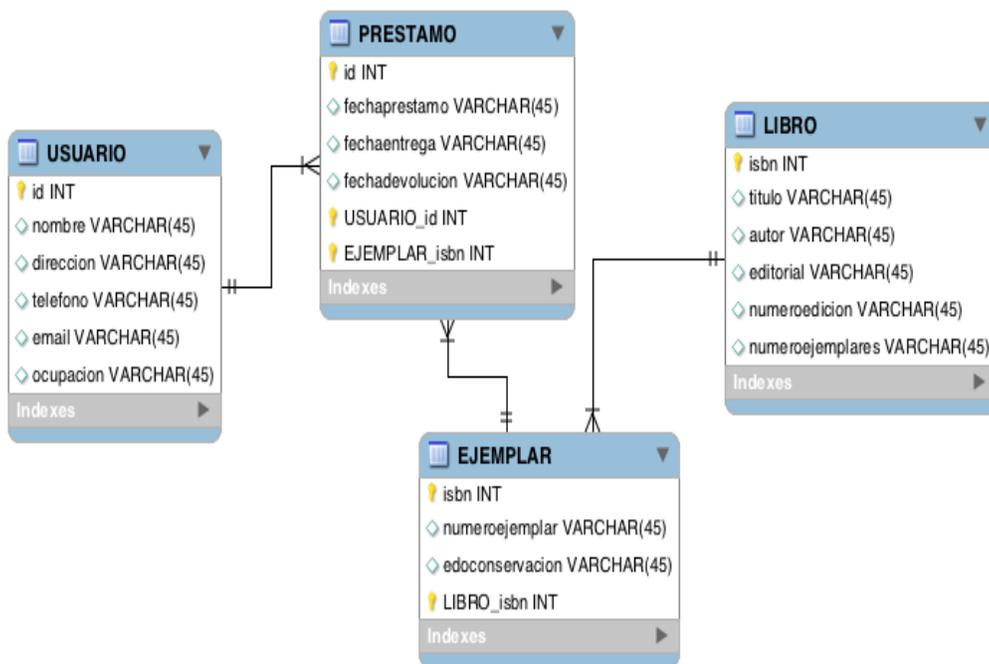
Ejemplo:

Ejemplo (borro todos los valores de las columnas alumno y materia donde la materia sea matemáticas, de la tabla materia):

```
# DELETE FROM cursada WHERE "materia" = 'matemáticas';
```

## Metodología y/o procedimiento

A partir del siguiente esquema de bases de datos (si ya realizó la práctica uno utilice la base de datos física ya creada)





- a) Implementar la base de datos física en el sistema gestor de bases de datos MySQL.

Use las sentencias del LMD

SELECT  
INSERT  
UPDATE  
DELETE

Considere que las llaves primarias son auto incrementables.

- b) Inserte 20 datos en cada una de las tablas a fin de poder hacer la segunda parte de esta practica
- c) Después de tener los 20 registros en cada tabla, actualice o cambie datos en cada una de ellas a fin de adquirir la competencia de modificar datos
- d) Agregué otras 5 tuplas en la tabla usuarios y después bórrelos a fin de adquirir las competencias de borrado de datos

## e) Resultados esperados

Scrip que contenga la creación de la base de datos, creación de las tablas y la inserción de los datos contenidos en cada tabla de la base de datos

Sentencias de inserción, actualización y eliminación de datos de datos en el block de notas

Capturar las pantallas que demuestren el resultado de cada una de las sentencias usadas

Describe el uso de cada sentencia considerando la semántica del SQL



# Práctica número. 3

## Nombre de la Práctica

### Condicionantes para crear filtros

Lenguaje de manipulación de datos DML  
(Data Manipulation Language)¶

## PARTE II

### Introducción

### CONSULTAS

Para realizar consultas sobre las tablas de las bases de datos disponemos de la instrucción SELECT. Con ella podemos consultar una o varias tablas. Este comando es el más versátil del lenguaje SQL.

Existen muchas cláusulas asociadas a la sentencia SELECT (GROUP BY, ORDER, HAVING, UNION). También es una de las instrucciones en la que con más frecuencia los motores de bases de datos incorporan cláusulas adicionales al estándar, que es el que veremos aquí.

Vamos a empezar viendo las consultas simples, basadas en una sola tabla. Veremos cómo obtener tuplas y atributos de una tabla en el orden en que nos haga falta.

SELECT Es utilizado para consultar registros de la base de datos que satisfagan un criterio determinado

El resultado de una consulta SELECT, devuelve **una tabla lógica o una tabla en memoria**. Es decir, los resultados son una relación de datos, que tiene registros, con una serie de atributos. Igual que cualquier tabla de la base de datos. Sin embargo esta tabla está en memoria, mientras la utilizamos, y luego se descarta. Cada vez que ejecutamos la consulta se vuelve a calcular el resultado.



La sintaxis básica de una consulta SELECT es la siguiente (los valores opcionales van entre corchetes):

```
SELECT [ ALL / DISTINCT ] [ * ] / [Lista de atributos] AS [Expresion]
FROM Nombre_Tabla_Vista
WHERE Condiciones
ORDER BY ListaColumnas [ ASC / DESC ]
```

## SELECT

Permite seleccionar las columnas que se van a mostrar y en el orden en que lo van a hacer. Simplemente es la instrucción que la base de datos interpreta como que vamos a solicitar información.

## ALL / DISTINCT

ALL es el valor predeterminado, especifica que el conjunto de resultados puede incluir filas duplicadas. Por regla general nunca se utiliza.

DISTINCT especifica que el conjunto de resultados sólo puede incluir filas únicas. Es decir, si al realizar una consulta hay registros exactamente iguales que aparecen más de una vez, éstos se eliminan. Muy útil en muchas ocasiones.

## Nombres de atributos

Se debe especificar una lista de nombres de atributos de la tabla o tablas que nos interesan consultar y que por tanto, queremos devolver. Normalmente habrá más de uno, en cuyo caso separamos cada nombre de los demás mediante comas.

Se puede anteponer el nombre de la tabla al nombre del atributo, utilizando el formato *Tabla.atributo*. Además de nombres de atributos,



en la lista se pueden poner constantes, expresiones aritméticas, y funciones, para obtener campos calculados de manera dinámica.

Si queremos que devuelva todos los atributos de la tabla se utiliza el comodín "\*" (asterisco).

Los nombres indicados deben coincidir exactamente con los nombre de los campos de la tabla, pero si queremos que en nuestra tabla lógica de resultados tengan un nombre diferente podemos utilizar:

## AS

Permite renombrar los atributos si lo utilizamos en la cláusula SELECT, o renombrar tablas si lo utilizamos en la cláusula FROM. Es opcional. Con ello podremos crear diversos alias de atributos y tablas.

## FROM

Esta cláusula permite indicar las tablas o vistas de las cuales vamos a obtener la información.

## WHERE

Especifica la **condición de filtro** de las filas devueltas. Se utiliza cuando no se desea que se devuelvan todas las filas de una tabla, sino sólo las que cumplen ciertas condiciones. Lo habitual es utilizar esta cláusula en la mayoría de las consultas.

## Condiciones

Son **expresiones lógicas** a comprobar para ser usadas como un segundo filtro, que tras su aplicación devuelven para cada atributo TRUE o FALSE, en función de que se cumplan o no la condición.

Se puede utilizar cualquier expresión lógica y en ella utilizar **diversos operadores** como:

> (Mayor)



>= (Mayor o igual)

< (Menor)

<= (Menor o igual)

= (Igual)

<> o != (Distinto)

IS [NOT] NULL (para comprobar si el valor del atributo es o no es nula, es decir, si contiene o no contiene algún valor)

LIKE

Para la comparación de un modelo se utiliza los caracteres comodines especiales: “%” y “\_”.

Con “%” se indica que en su lugar puede ir cualquier cadena de caracteres, y con “\_” se puede ir cualquier carácter individual (un solo carácter). Con la combinación de estos caracteres se pueden obtener múltiples patrones de búsqueda.

Por ejemplo:

- El nombre empieza por A: Nombre LIKE ‘A%’
- El nombre acaba por A: Nombre LIKE ‘%A’
- El nombre contiene la letra A: Nombre LIKE ‘%A%’
- El nombre empieza por A y después contiene un solo carácter cualquiera: Nombre LIKE ‘A\_’
- El nombre empieza una A, después cualquier carácter, luego una E y al final cualquier cadena de caracteres: Nombre LIKE ‘A\_E%’

BETWEEN: para un intervalo de valores.

Por ejemplo:



Usuarios entre el 30 y el 100: Usuario BETWEEN 30 AND 100

Usuarios nacidos entre 1970 y 1979: FechaNac BETWEEN '1970-01-01' AND '1979-12-31'

IN( ): para especificar una relación de valores concretos. Por ejemplo: Préstamo de usuarios 10, 15, 30 y 75: Usuario IN(10, 15, 30, 75)

Es posible combinar varias condiciones simples de los operadores anteriores utilizando los operadores lógicos OR, AND y NOT, así como el uso de paréntesis para controlar la prioridad de los operadores (como en matemáticas).

## ORDER BY

Define el orden de las tuplas del conjunto de resultados. Se especifica el atributo o atributos (separados por comas) por los cuales queremos ordenar los resultados.

## ASC / DESC

ASC es el valor predeterminado, especifica que el atributo indicado en la cláusula ORDER BY se ordenará de menor a mayor. Si por el contrario se especifica DESC se ordenará de mayor a menor.

## Algunos ejemplos

Mostrar todos los datos de los usuarios de nuestra empresa:

```
SELECT * FROM Usuarios;
```



Mostrar apellido, ciudad y región (LastName, city, region) de los empleados de USA (nótese el uso de AS para darle el nombre en español a los campos devueltos):

```
SELECT E.LastName AS Apellido, City AS Ciudad, Region  
FROM Employees AS E  
WHERE Country = 'USA';
```

Mostrar los clientes que no sabemos a qué región pertenecen (o sea, que no tienen asociada ninguna región):

```
SELECT * FROM Clientes WHERE Region IS NULL;
```

Mostrar las distintas regiones de las que tenemos algún cliente, accediendo sólo a la tabla de clientes:

```
SELECT DISTINCT Region FROM Clientes WHERE Region IS NOT NULL;
```

Mostrar los clientes que pertenecen a las regiones CA, MT o WA, ordenados por región ascendentemente y por nombre descendentemente.

```
CODE SELECT * FROM Clientes WHERE Region IN('CA', 'MT', 'WA')  
ORDER BY Region, CompañiaNom DESC;
```

Mostrar los clientes cuyo nombre empieza por la letra "W":

```
SELECT * FROM Clientes WHERE CompañiaNom LIKE 'W%';
```

Mostrar los empleados cuyo código está entre el 2 y el 9:

```
SELECT * FROM Usuario WHERE codigo BETWEEN 2 AND 9;
```



Mostrar los clientes cuya dirección contenga “ki”:

```
SELECT * FROM Clientes WHERE Direccion LIKE '%ki%';
```

Mostrar las Ventas del producto 65 con cantidades entre 5 y 10, o que no tengan descuento:

```
SELECT * FROM [Ventas] WHERE (ProductoID = 65 AND Cantidad BETWEEN 5 AND 10) OR SinDescuento = 0;
```

## Objetivo

Aprender a realizar consultas de bases de datos usando filtros con los comandos: LIKE, BETWEEN, IN(), OPERADORES LOGICOS OR AND Y NOT, ORDER BY, FUNCIONES DE AGREGACIÓN

## Unidad de Competencia a ser fortalecida V

### Desempeño

Realizar la implementación de base de datos mediante aun SGBD

Insertar datos en una base de datos

Hacer consultas a una tabla

Hacer consultas a dos tablas

Hacer consultas a tres tablas

### Producto a obtener

Implementación de base de datos en un medio electrónico e insertar,  
Consultas de múltiples tablas al mismo tiempo

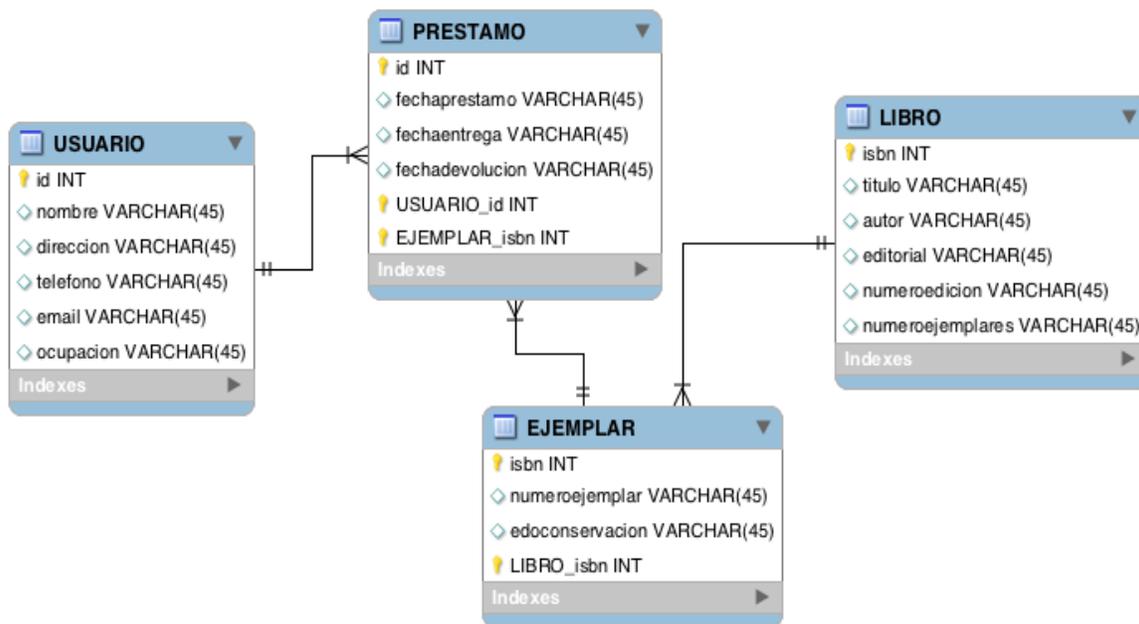


## Materiales y software

Laboratorio de cómputo  
Equipo de cómputo  
Sistema operativo Windows o Linux  
MySQL  
Manual de MySQL

## Metodología y/o procedimiento

A partir del siguiente esquema de bases de datos (si ya realizó la práctica uno utilice la base de datos física ya creada)



- Implementar la base de datos física en el sistema gestor de bases de datos MySQL.
- Realice las siguientes consultas



Obtener todos los usuarios que tengan el correo electrónico con el dominio Gmail.

Obtenga a todos los usuarios que tengan de ocupación estudiante

Obtenga todos los ejemplares que su estado de conservación sea nuevo.

Obtenga todos los libros que tiene la editorial MC GRILL

Cuente cuantos libros tiene la editorial DIANA

Se desea saber cuántos ejemplares tiene cada libro

Identifique que id de usuario pidió libros prestados del 2018-01-01 al 2019-01 01.

Identifique que usuarios tienen el teléfono numérico donde los números iniciales sean 595

Obtenga a todos los libros cuyo autor sea GARCIA MARQUEZ

Obtenga todos los libros con título matemáticas de la editorial RAMSA

Obtenga el número de ISBN de los libros que tengan más de 2 ejemplares

Obtenga todos los libros cuyo número de edición este entre la 5 y 8

Cuente cuantos libros hay de cada editorial

Obtenga los nombres de usuario de los que tienen correo electrónico en Hotmail y Yahoo



Obtenga a todos los usuarios con apellido Juárez y García

## Resultados esperados

Scrip que contenga la creación de la base de datos, creación de las tablas y los datos contenidos en cada tabla de la base de datos

Sentencias de las consultas planteadas en el block de notas

Capturar las pantallas que demuestren el resultado de cada una de las sentencias usadas

Describa el uso de cada sentencia considerando la semántica del SQL



# **Práctica número. 4**

## **Nombre de la Práctica**

### **Consultas a tablas múltiples**

#### **Lenguaje de manipulación de datos DML (Data Manipulation Language)¶**

### **PARTE III**

## **Introducción**

### **CONSULTAS A MULTIPLES TABLAS**

La consulta a múltiples tablas al mismo tiempo, en las bases de datos a veces se puede volver un poco complicada, sobre todo con el tema de las tablas relacionales. Este tipo de consultas son las más usadas por las empresas ya que se requiere cruzar datos para poder obtener información.

Las consultas pueden ser entre dos tablas, tres tablas o más. En estos casos es recomendable usar las relaciones entre tablas y las sentencias JOIN de SQL.

#### **Con Relaciones**

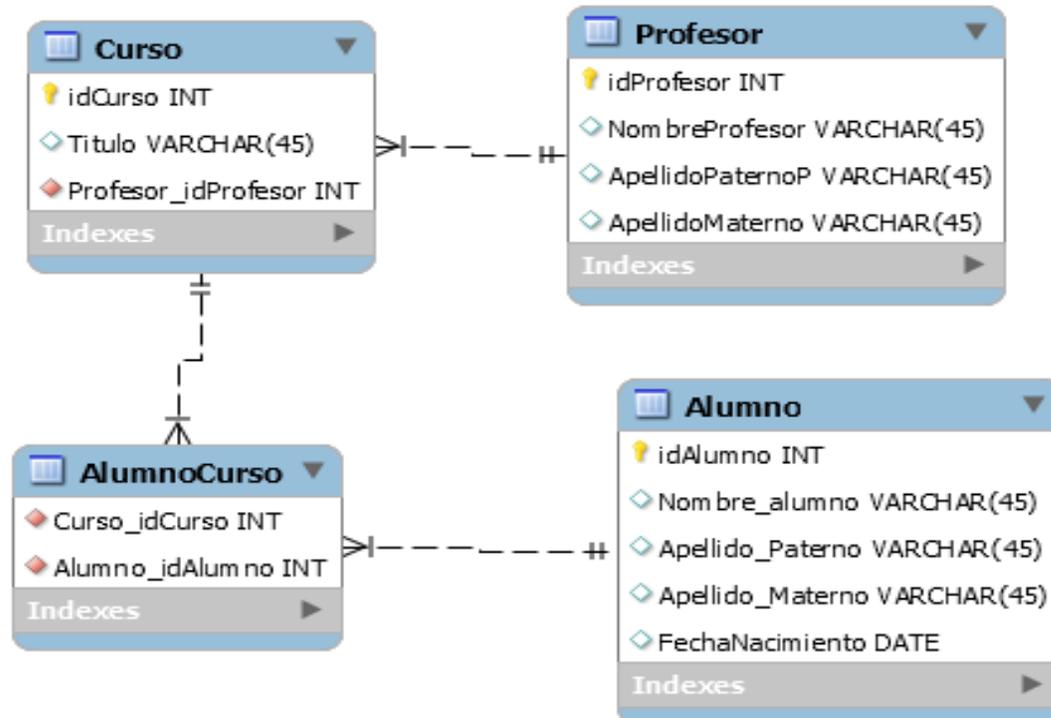
El modo de relacionar registros entre tablas es por tanto mediante referencias, para lo cual se usan los identificadores definidos como claves primarias y foráneas.

Supongamos una academia donde se imparten clases, habrá cursos, profesores y alumnos.

En una base de datos se diseña una tabla para cada entidad, es decir, para alumnos, profesores y cursos. Veamos cómo se relacionan entre



si estas tres tablas y como se establecen estas relaciones en la base de datos.



## Ejemplos

Para obtener la consulta con **dos tablas** se puede plantear lo siguiente que cursos imparte el profesor se usa la siguiente sentencia

```
SELECT
Curso.Titulo, Profesor.NombreProfesor, Profesor.ApellidoPaternoP ,
Profesor.ApellidoMaterno
FROM
Curso, Profesor
WHERE
Curso.Profesor_idProfesor = Profesor.IdProfesor;
```



Para **tres tablas** se puede plantear la siguiente consulta

Se desea saber qué curso ha cursado cada alumno.

Se Plantea la siguiente sentencia

```
SELECT
Curso.Titulo, Alumno.Nombre_alumno, Alumno.Apellido_Paterno,
Alumno.Apellido_Materno
FROM
Curso,Alumno,AlumnoCurso
WHERE
Curso.Profesor_idProfesor = Profesor.IdProfesor
AND Curso.idCurso = AlumnoCurso.Curso_idCurso
AND AlumnoCurso.Alumno_idAlumno = Alumno.idAlumno;
```

Con JOIN

El INNER JOIN cláusula MySQL

El INNER JOIN sirve para juntar dos o más tablas

Tiene la siguiente semántica

```
SELECT
    Selecciona lista de tributos
FROM t1 (tabla 1)
INNER JOIN t2 (tabla 2) ON join_condition1
INNER JOIN t3 (tabla 3) ON join_condition2
...;
```



En esta sintaxis:

- Primero, especifique la tabla principal que aparece en la FROM cláusula (t1).
- En segundo lugar, especificar la tabla que se combina con la tabla principal, que aparece en la INNER JOIN cláusula (t2, t3, ...).
- Tercero, especifique una condición de unión después de la ON palabra clave de la INNER JOIN cláusula. La condición de unión especifica la regla para hacer coincidir filas entre la tabla principal y la tabla que aparece en la INNER JOIN cláusula.

Suponiendo que desea unir dos tablas t1y t2. La siguiente declaración ilustra cómo unir dos tablas t1y t2 al usar la INNER JOIN:

```
SELECT
    select_list
FROM
    t1
INNER JOIN t2 ON join_condition;
```

Considerando la base de datos de alumno cursos el planteamiento sería el siguiente

```
SELECT
Curso.Titulo, Profesor.NombreProfesor, Profesor.ApellidoPaternoP ,
Profesor.ApellidoMAterno
```



```
FROM Curso
```

```
INNER JOIN Profesor ON Curso.Profesor_idProfesor=Profesor.IdProfesor;
```

## Con tres tablas seria

```
SELECT
```

```
Curso.Titulo, Alumno.Nombre_alumno, Alumno.Apellido_Paterno,  
Alumno.Apellido_Materno  
FROM Curso
```

```
INNER JOIN AlumnoCurso ON AlumnoCurso.Alumno_idAlumno = Alumno.idAlum  
no
```

```
INNER JOIN Profesor ON Curso.Profesor_idProfesor = Profesor.IdProfeso  
r
```

A estas consulta se pueden agregar los filtros que se vieron en la practica 3 para hacer consultas más específicas.

Esto es intuitivo y se puede plantear para más de tres tablas basta con hacer la igualdad de las relaciones que hay entre las tablas.

Para practicar plantee los siguientes ejercicios

Ejercicio 1 Construya una consulta que devuelva los cursos en que se ha matriculado el alumno con identificador 1

Modifique la anterior consulta para que devuelva los nombres y apellidos de los alumnos, y los cursos en que se han matriculado, tales que el nombre de pila del alumno contenga un E.

### Ejercicio 2

¿Cuántos cursos imparte cada profesor? Construya una consulta que responda a esta cuestión de modo que el resultado muestre el nombre completo del profesor acompañado del número de cursos que imparte.



### Ejercicio 3

¿Cuántos alumnos hay matriculados en cada uno de los cursos? Construya una consulta que responda a esta cuestión de modo que el resultado muestre el título del curso acompañado del número de alumnos matriculados.

Modifique la anterior consulta de modo que muestre aquellos cursos que el número de alumnos matriculados sea exactamente de dos alumnos.

### Ejercicio 4

Si ahora a usted le pidiesen que adaptara la BD, que consta de las tres tablas presentadas en esta lección, a la siguiente necesidad: A todo alumno se le asignara un profesor que lo tutele. ¿Que cambios realizaría en la BD?

## Objetivo

Aprender a realizar consultas de bases de datos con dos y tres tablas usando las relaciones y el JOIN

## Unidad de Competencia a ser fortalecida V

### Desempeño

Realizar consultas a más de una tabla al mismo tiempo

Consultas de dos tablas a partir de las relaciones y los atributos llave primaria y foránea

Consultas de tres tablas a partir de las relaciones y los atributos llave primaria y foránea



Consultas de **dos** tablas usando el JOIN de SQL  
Consultas de **tres** tablas usando el JOIN de SQL

## Producto a obtener

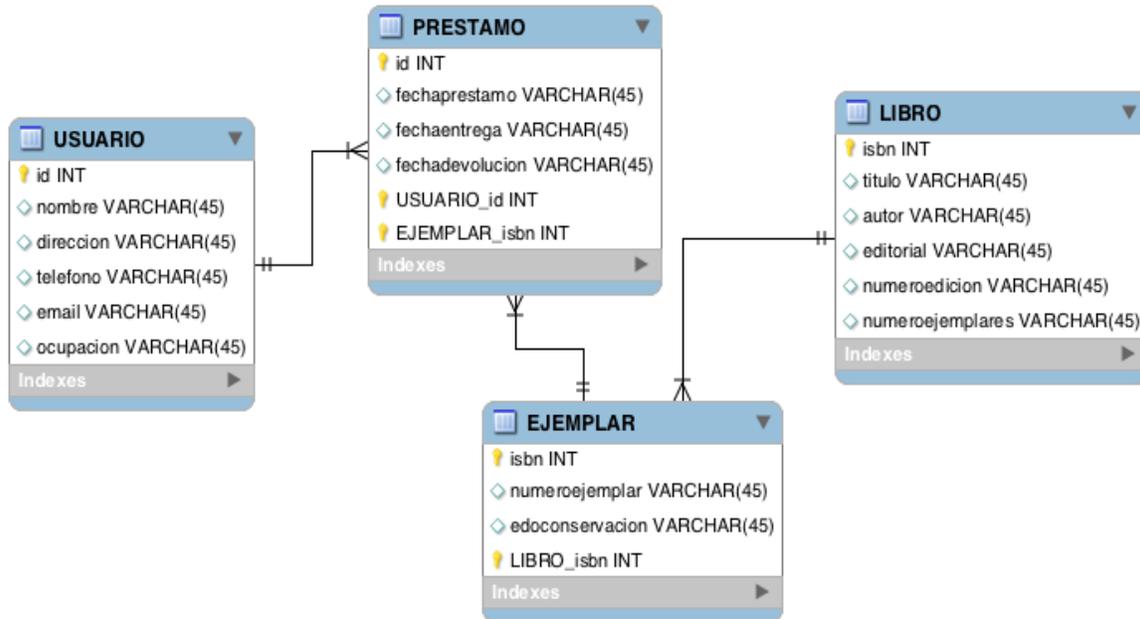
Consultas de múltiples tablas al mismo tiempo

## Materiales y software

Laboratorio de cómputo  
Equipo de cómputo  
Sistema operativo Windows o Linux  
MySQL  
Manual de MySQL

## Metodología y/o procedimiento

A partir del siguiente esquema de bases de datos (si ya realizó la práctica uno, utilice la base de datos física ya creada)



Obtenga el nombre de usuario, ocupación, número de ejemplar prestado y fecha de entrega

Obtenga el nombre de usuario, ocupación, número de ejemplar prestado y fecha de entrega y ordénelos por fecha de entrega

Obtenga el título de libro, autor, editorial, el número de ejemplares de todos los libros y ordénelos por editorial

Obtenga el título de libro, autor, editorial, el número de ejemplares de todos los libros de la editorial McGraw.

Obtenga el título de libro, autor, editorial, el número de ejemplares de todos los libros de la editorial McGraw y DIANA

Obtenga el título de libro, autor, editorial, el número de ejemplares de todos los libros de la editorial McGraw y ordénelos por editorial





Obtenga el nombre del usuario, teléfono, fecha de devolución, número de ejemplar y título.

Obtenga el nombre del usuario, teléfono, fecha de devolución, número de ejemplar y título que la fecha de devolución sea del día 2018-03-01.

Obtenga el nombre del usuario, teléfono, fecha de devolución, número de ejemplar y título que la fecha de devolución sea del periodo 2018-03-01.al 2019-03-01

Obtenga el nombre del usuario, teléfono, fecha de devolución, número de ejemplar y título que la fecha de devolución sea del periodo 2018-03-01.al 2019-03-01 y ordénelos por fecha de devolución descendientemente.

## Resultados esperados

Sentencias de las consultas planteadas en el block de notas

Capturar las pantallas que demuestren el resultado de cada una de las sentencias usadas

Describa el uso de cada sentencia considerando la semántica del SQL



## Referencias

Andy Opper Robert Sheldon (2010) Fundamentos de SQL. Tercera edición. McGRAW-HILL INTERAMERICANA EDITORES. ISBN: 978-607-15-0251-3

Manual de referencia de MySQL 8.0.

<https://dev.mysql.com/doc/refman/8.0/en/>

P. Rob y C. Coronel (2004) Sistemas de bases de datos, Diseño, implementación y administración. Ed. Thomson, 2004. ISBN 9706862862.

Perez Lopez Cesar (2017) MySQL para Windows y Linux. Segunda edición. Editorial RA-MAS. ISBN 978-84-7897-790-1

Saddle River Upper (2013). Data administration : the complete guide to DBA practices and procedures 2 Edicion. AddisonWesley.

Silberschatz, H. F. Horth y S. Sudarshan (2006) Fundamentos de Bases de Datos (4ª Edición). McGraw-Hill. ISBN: 0-07-295886-3