



Datos Generales de la asignatura

Nombre de la asignatura:	Programación de Base de Datos
Clave de la asignatura:	GIB-2501
SATCA ¹ :	1 - 4- 5
Carrera:	Ingeniería en Sistemas Computacionales

2. Presentación

Caracterización de la asignatura

Programación de base de datos, es una asignatura de especialidad perteneciente al 7º semestre, dicha asignatura presenta nuevos conceptos, pero se asienta en otros ya impartidos, relacionados tanto con: Fundamentos de Bases de Datos, Taller de Bases de Datos y Administración de Bases de Datos, así como también, asignaturas orientadas a la programación (Fundamentos de programación y Programación Orientada a Objetos).

La asignatura Programación de Base de Datos tiene como propósito desarrollar en el estudiante las habilidades necesarias para diseñar, implementar y optimizar la programación del lado del servidor en bases de datos relacionales. Los estudiantes aprenderán a utilizar SQL avanzado, procedimientos almacenados, triggers y manejo de transacciones para garantizar la integridad y eficiencia en la gestión de datos. Se trabajará con diferentes sistemas gestores de bases de datos (SGBD), tanto propietarios como de código abierto.

Esta asignatura es clave en el perfil de un ingeniero en sistemas computacionales, ya que la mayoría de las aplicaciones empresariales dependen de bases de datos relacionales para almacenar y procesar información de manera eficiente y segura.

Esta asignatura aporta al perfil del egresado la capacidad de aplicar, analizar y sintetizar procesos de diseño de ingeniería que resulten en proyectos que cumplen las necesidades especificadas empleando base de datos, comunicando de manera asertiva y efectiva, oral y/o escrita, los resultados

¹ Sistema de Asignación y Transferencia de Créditos Académicos





obtenidos en su investigación o propuesta de solución a problemas específicos, utilizando un lenguaje técnico y aplicando eficientemente las TIC; participando de forma colaborativa en un equipo para la construcción de soluciones computacionales, asumiendo un rol específico en la planeación, cumplimiento de las actividades y gestión de riesgos, para el logro de los objetivos comunes.

Intención didáctica

La asignatura de Programación de Base de Datos permite que el alumno desarrolle las habilidades necesarias para implementar programación avanzada en bases de datos relacionales, garantizando la integridad, seguridad y eficiencia en la manipulación de datos. A través de la programación del lado del servidor, los estudiantes aprenderán a utilizar procedimientos almacenados, disparadores, transacciones y técnicas de optimización para gestionar grandes volúmenes de información en sistemas empresariales.

El curso está estructurado en cuatro unidades de aprendizaje, que siguen una progresión lógica desde los fundamentos de la programación en bases de datos hasta la implementación de estrategias avanzadas para el manejo y optimización del rendimiento.

En la primera unidad, se introducen los conceptos fundamentales de la programación en bases de datos relacionales, abordando el uso de estructuras de control, cursores y técnicas de manejo de errores en SQL. En la segunda unidad, los alumnos desarrollan procedimientos almacenados y funciones definidas por el usuario, aplicando parámetros de entrada y salida para optimizar el acceso y procesamiento de datos.

En la tercera unidad, se trabaja con disparadores (triggers) para la automatización y validación de datos dentro de una base de datos, asegurando la integridad de la información en los procesos transaccionales. Finalmente, en la cuarta unidad, se abordan las transacciones, el control de concurrencia y la optimización del rendimiento, incluyendo la implementación de índices, planes de ejecución y estrategias de seguridad para la protección de los datos.

A lo largo del curso, los estudiantes trabajarán con SGBD propietarios y de código abierto como SQL Server, MySQL y PostgreSQL, realizando ejercicios prácticos que les permitirán consolidar su aprendizaje mediante la implementación de soluciones en bases de datos reales. Además, el enfoque didáctico incluye el desarrollo de un proyecto final, donde los alumnos deberán diseñar y optimizar una base de datos relacional con las técnicas aprendidas en la asignatura.





3. Participantes en la actualización, el diseño, consolidación y/o seguimiento curricular del programa

Lugar y fecha de elaboración o revisión	Participantes	Observaciones
Tecnológico Nacional de México campus Lerdo. Ciudad Lerdo, Durango a Febrero de 2025	Instituto Tecnológico Superior de Lerdo. Academia de Sistemas Computacionales en coordinación con la Dra. Karla Verónica Rodríguez Lozano y MA. Edna Velia Josefina Solorio Vega	Reunión Académica de la División de Ingeniería en Sistemas Computacionales del Instituto Tecnológico Superior de Lerdo, para el análisis y diseño de la especialidad "Gestión de Información y Soluciones Multiplataforma".

4. Competencia(s) a desarrollar

Competencia(s) específica(s) de la asignatura
Diseñar e implementar programación del lado del servidor en bases de datos relacionales mediante el uso de procedimientos almacenados, triggers, funciones y transacciones, garantizando integridad, seguridad y eficiencia en la manipulación de los datos.

5. Competencias previas

• Diseño de bases de datos relacionales. • Modelo entidad-relación y normalización. • Conocimiento del lenguaje SQL. • Administración básica de bases de datos

6. Temario

No.	Temas	Subtemas
	1. Programación del Lado del Servidor en SQL	1.1 Introducción a la programación en bases de datos relacionales. 1.2 Programación con Transact-SQ, SQL/PSM y PL/pgSQL 1.3 Uso de variables y estructuras de control (IF, CASE, WHILE, LOOP). 1.3 Cursores: Definición, uso y optimización. 1.4 Manejo de errores y depuración en SQL.





2. Procedimientos Almacenados y Funciones	2.1 Procedimientos almacenados: Creación, uso y optimización. 2.2 Parámetros de entrada y salida en procedimientos almacenados. 2.3 Funciones definidas por el usuario (UDF). 2.4 Diferencias entre funciones y procedimientos almacenados. 2.5. Creación de procedimientos y funciones en Transact-SQ, SQL/PSM y PL/pgSQL 2.6 Optimización y mejores prácticas en procedimientos almacenados.
3. Disparadores (Triggers) y Control de Integridad de Datos	3.1 Introducción a los disparadores en bases de datos relacionales. 3.2 Tipos de disparadores: BEFORE, AFTER, INSTEAD OF. 3.3 Creación y administración de triggers en Transact-SQ, SQL/PSM y PL/pgSQL. 3.4 Uso de disparadores para validación e integridad de datos. 3.5 Casos de uso de triggers en sistemas empresariales.
4. Manejo de Transacciones y Control de Concurrencia	4.1 Concepto de transacción y propiedades ACID. 4.2 Implementación de transacciones en SQL Server y PostgreSQL. 4.3 Control de concurrencia y niveles de aislamiento. 4.4 Bloqueos y estrategias para evitar el acceso concurrente conflictivo. 4.5 Optimización del rendimiento en transacciones de bases de datos.





7. Actividades de aprendizaje de los temas

Fundamentos de Programación en Bases de Datos Relacionales	
Competencias	Actividades de aprendizaje
Específica(s): - Comprender y aplicar estructuras de control en programación SQL. - Implementar cursores para el procesamiento de datos en bases de datos relacionales. - Manejar adecuadamente errores en consultas y procedimientos almacenados. Genéricas: - Capacidades metodológicas, para organizar el tiempo y estrategias de aprendizaje. - Habilidades interpersonales. - Capacidad crítica y autocrítica. - Habilidad para trabajar en forma autónoma. - Trabajo en Equipo. - Uso de la Tecnología	- Escritura de scripts SQL con estructuras de control. - Desarrollo de consultas dinámicas en bases de datos relacionales. - Implementación de cursores y control de errores en procedimientos almacenados.
Procedimientos Almacenados y Funciones	
Competencias	Actividades de aprendizaje
Específica(s): - Diseñar y desarrollar procedimientos almacenados para mejorar la eficiencia en bases de datos relacionales. - Implementar funciones definidas por el usuario en SQL Server, MySQL y PostgreSQL. - Aplicar técnicas de optimización en procedimientos almacenados. Genéricas:	- Creación de procedimientos almacenados con diferentes tipos de parámetros. - Implementación de funciones personalizadas en bases de datos. - Comparación de rendimiento entre consultas directas y procedimientos almacenados. - Diseña un sistema de base de datos real en el que aplique SQL, procedimientos almacenados y desencadenadores, cursores y reportes.





• Capacidades metodológicas, para organizar el tiempo y estrategias de aprendizaje. • Habilidades interpersonales. • Capacidad crítica y autocrítica. • Habilidad para trabajar en forma autónoma. • Trabajo en Equipo. • Uso de la Tecnología	
Disparadores (Triggers) y Control de Integridad de Datos	
Competencias	Actividades de aprendizaje
Específica(s): • Diseñar y programar disparadores para la automatización y validación de datos. • Implementar estrategias de control de integridad en bases de datos relacionales. Genéricas: • Capacidades metodológicas, para organizar el tiempo y estrategias de aprendizaje. • Habilidades interpersonales. • Capacidad crítica y autocrítica. • Habilidad para trabajar en forma autónoma. • Trabajo en Equipo. • Uso de la Tecnología	• Creación de triggers para control de reglas de negocio y auditoría. • Implementación de triggers para mantener consistencia en bases de datos. • Análisis del impacto de los disparadores en el rendimiento de la base de datos.
Manejo de Transacciones y Control de Concurrencia	
Competencias	Actividades de aprendizaje
Específica(s): • Implementar transacciones para garantizar la consistencia y seguridad de los datos.	• Desarrollar scripts SQL con transacciones y manejo de errores. • Evaluar diferentes niveles de aislamiento en bases de datos. • Implementar pruebas de concurrencia para detectar problemas de bloqueo.





• Aplicar estrategias de control de concurrencia en entornos multiusuario • Genéricas: • Capacidades metodológicas, para organizar el tiempo y estrategias de aprendizaje. • Habilidades interpersonales. • Capacidad crítica y autocrítica. • Habilidad para trabajar en forma autónoma. • Trabajo en Equipo. • Uso de la Tecnología	
--	--

8. Práctica(s)

1. Consultas SQL Avanzadas 2. Operaciones Básicas de una Base de Datos 3. Desarrollo de procedimientos almacenados para operaciones CRUD avanzadas. 4. Creación de triggers para auditoría y control de cambios en datos. 5. Implementación de triggers para control de reglas de negocio. 6. Implementación de transacciones y pruebas de concurrencia. 7. Optimización de rendimiento en consultas SQL mediante índices. 8. Configuración de seguridad y roles en bases de datos relacionales. 9. Configuración de seguridad y pruebas de inyección SQL.
--

9. Proyecto de asignatura

Desarrollo de un sistema de base de datos relacional para una aplicación empresarial, el cual deberá incluir: • Procedimientos almacenados y funciones definidas por el usuario. • Disparadores para automatización y validación de datos. • Implementación de transacciones y control de concurrencia. • Estrategias de optimización y seguridad en bases de datos. El objetivo del proyecto que plantee el docente que imparta esta asignatura, es demostrar el desarrollo y alcance de la(s) competencia(s) de la asignatura, considerando las siguientes fases:



- **Fundamentación:** marco referencial (teórico, conceptual, contextual, legal) en el cual se fundamenta el proyecto de acuerdo con un diagnóstico realizado, mismo que permite a los estudiantes lograr la comprensión de la realidad o situación objeto de estudio para definir un proceso de intervención o hacer el diseño de un modelo.
- **Planeación:** con base en el diagnóstico en esta fase se realiza el diseño del proyecto por parte de los estudiantes con asesoría del docente; implica planificar un proceso: de intervención empresarial, social o comunitario, el diseño de un modelo, entre otros, según el tipo de proyecto, las actividades a realizar los recursos requeridos y el cronograma de trabajo.
- **Ejecución:** consiste en el desarrollo de la planeación del proyecto realizada por parte de los estudiantes con asesoría del docente, es decir en la intervención (social, empresarial), o construcción del modelo propuesto según el tipo de proyecto, es la fase de mayor duración que implica el desempeño de las competencias genéricas y específicas a desarrollar.
- **Evaluación:** es la fase final que aplica un juicio de valor en el contexto laboral-profesión, social e investigativo, ésta se debe realizar a través del reconocimiento de logros y aspectos a mejorar se estará promoviendo el concepto de “evaluación para la mejora continua”, la metacognición, el desarrollo del pensamiento crítico y reflexivo en los estudiantes.

10. Evaluación por competencias

- Exámenes teóricos y prácticos sobre SQL avanzado, procedimientos almacenados y transacciones.
- Desarrollo de prácticas y casos de estudio que involucren programación avanzada en bases de datos.
- Proyecto final que integre los conocimientos adquiridos durante la asignatura.

Para evaluar las actividades de aprendizaje se recomienda solicitar: mapas conceptuales o mentales, cuadros comparativos, reportes de prácticas, portafolio de evidencias, entre otros. Para verificar el nivel del logro de las competencias del estudiante se recomienda utilizar: listas de cotejo, listas de verificación, matrices de valoración, guías de observación, rúbricas, Desarrollo de un proyecto final que integre todas las unidades de aprendizaje, entre otros.



11. Fuentes de información

- 1) Silberschatz, A., Korth, H., Sudarshan, S. (2020). *Fundamentos de bases de datos*. McGraw-Hill.
- 2) Date, C. J. (2019). *An Introduction to Database Systems*. Addison-Wesley.
- 3) Ben-Gan, I. (2019). *T-SQL Fundamentals*. Microsoft Press.
- 4) Elmasri, R., & Navathe, S. B. (2015). *Fundamentos de sistemas de bases de datos*. Pearson Educación.
- 5) Coronel, C., & Morris, S. (2012). *Bases de datos: diseño, implementación y administración*. Cengage Learning.
- 6) PostgreSQL Global Development Group (2023). *PostgreSQL Documentation*.
- 7) Redgate Software (2023). *SQL Server Execution Plans*.
- 8) Microsoft Docs (2023). *SQL Server Documentation*.

