Name of the course (Spanish):

Diseño, desarrollo e implementación de bases de datos

Name of the course (English):

Database Design, Development & Implementation

Description of the course (Spanish):

Descubra los secretos de una gestión eficaz de bases de datos con nuestro curso integral sobre diseño, desarrollo e implementación de bases de datos. Sumérgete en el mundo de los datos estructurados, domina el arte de diseñar bases de datos sólidas, desarrollar modelos de datos eficientes e implementar soluciones poderosas. Impartido por expertos de la industria, este curso le otorga las habilidades y conocimientos necesarios para prosperar en el dinámico campo de la tecnología de bases de datos. Eleve sus perspectivas profesionales y únase a nosotros en un viaje transformador hacia el corazón de la excelencia en la gestión de datos.

Description of the course (English):

Unlock the secrets of effective database management with our comprehensive course on Database Design, Development, and Implementation. Dive into the world of structured data, mastering the art of designing robust databases, developing efficient data models, and implementing powerful solutions. Taught by industry experts, this course empowers you with the skills and knowledge needed to thrive in the dynamic field of database technology. Elevate your career prospects and join us on a transformative journey into the heart of data management excellence.

URL of Module: Database Design, Development and Implementation

Cost: $1,295

Mode: Virtual
Duration: 160 (Hours)
Name: Database Design, Development & Implementation (Module)

Module Contents:

COURSE	TOPIC	DESCRIPTION
Data Modeling	Software Development Life Cycle (SDLC)	The Software Development Life Cycle (SDLC) is a systematic process used by software development teams to design, develop, test, and deploy software solutions.
Data Modeling	Requirement Gathering Techniques--JRD (Joint Requirement Definition, JAD(Joint Application Design)	Requirement gathering is a crucial initial phase in data modeling, where stakeholders' needs and objectives are identified and documented. It serves as the foundation for designing an effective data model that aligns with the organization's goals and supports decision-making processes
Data Modeling	Documentation--BRD (Business Requirement Document), FRD (Functional Requirement Document)	Documentation is a vital aspect of data modeling, serving as a comprehensive record of the data model's structure, relationships, and business rules. It facilitates communication among stakeholders, ensuring everyone involved understands the data model's purpose and design decisions
Data Modeling	Software Design Methodologies--Waterfall, Agile, Scrum, Spiral	Software design methodologies provide structured approaches for planning and implementing data modeling projects. These methodologies, such as Agile, Waterfall, or Spiral, offer frameworks for organizing tasks, managing resources, and mitigating risks throughout the data modeling process
Data Modeling	ER-Diagrams	ER diagrams serve as visual representations of the relationships among entities in a database system. They provide a clear and intuitive way to illustrate the structure and dynamics of data, facilitating effective communication between stakeholders, designers, and developers

Data Modeling	Normalization	Normalization is a fundamental concept in data modeling aimed at reducing redundancy and ensuring data integrity within a database. By organizing data into well-structured tables and eliminating duplicate information, normalization minimizes the risk of data anomalies such as insertion, update, and deletion anomalies
Data Modeling	Reverse and Forward Engineering	Reverse and forward engineering facilitate the optimization, evolution, and maintenance of data models, ensuring they remain relevant, efficient, and adaptable over time
Data Modeling	Database Integrity	Database integrity ensures the accuracy, consistency, and reliability of data stored within a database system. It encompasses various aspects such as entity integrity, referential integrity, and domain integrity, which collectively safeguard the quality and reliability of data
SQL Server	Introduction to SQL Server	Provides learners with a foundational understanding of this powerful relational database management system (RDBMS) developed by Microsoft. SQL Server is widely used in various industries for storing, managing, and analyzing data, making it essential for professionals pursuing careers in database administration, development, and business intelligence
SQL Server	Data Definition Language (DDL)----CREATE, ALTER, TRUNCATE,DROP	DDL commands in SQL Server are crucial for defining the structure and organization of database objects such as tables, indexes, views, and schemas. These commands enable database administrators and developers to create, modify, and manage the database schema, ensuring data integrity and consistency
SQL Server	Data Manipulation Language (DML)---INSERT, UPDATE, DELETE	DML commands in SQL Server are essential for manipulating data stored within database tables. These commands, including INSERT, UPDATE, and DELETE, allow users to retrieve, add, modify, and remove data, enabling dynamic interaction with the database
SQL Server	Data Query Language (DQL)---SELECT	DQL commands in SQL Server, primarily the SELECT statement, are fundamental for retrieving data from database tables. These commands enable users to perform queries to extract specific information based on specified criteria, facilitating data analysis, reporting, and decision-making processes

SQL Server	Constraints--Primary Key, Foreign Key, Unique Key, Check, Not Null	Constraints in SQL Server play a vital role in ensuring the integrity and reliability of database data. They define rules and limitations that enforce data accuracy, consistency, and validity within tables
SQL Server	Wild Cards	Wildcards in SQL Server are powerful tools used in query operations to perform pattern matching and search for data with unknown or variable values.By leveraging wildcards in SQL queries, users can create flexible and dynamic search conditions, enabling them to retrieve relevant data efficiently from large datasets.
SQL Server	Joins--INNER JOIN, OUTER JOIN, SELF JOIN, RESTRICTED JOIN, CROSS JOIN	Joins in SQL Server are essential for combining data from multiple tables based on related columns, enabling users to retrieve comprehensive and meaningful information from a database. By using various types of joins such as INNER JOIN, LEFT JOIN, RIGHT JOIN, and FULL JOIN, users can establish relationships between tables and extract relevant data sets that meet specific criteria
SQL Server	Set Operators--UNION, UNION ALL, INTERSECT, EXCEPT	Set operators in SQL Server are powerful tools used to combine and manipulate the results of multiple queries. These operators, such as UNION, INTERSECT, and EXCEPT, enable users to perform set operations like union, intersection, and difference on the results of two or more SELECT statements
SQL Server	Sub Queries---CORRELATED AND NON-CORRELATED SUBQUERES	Subqueries in SQL Server are powerful tools used to nest one query within another, allowing for more complex and dynamic data retrieval and manipulation. They enable users to break down complex problems into smaller, more manageable parts, making queries easier to write, understand, and maintain
SQL Server	CTE (Common Table Expression)	CTEs in SQL Server are valuable tools used to simplify complex queries and improve query readability and maintainability. They allow users to define temporary result sets within a query, which can then be referenced multiple times within the same query
SQL Server	Views (Schemabinding Views, Views with Encryption)	Views in SQL Server are virtual tables that represent a subset of data from one or more tables in the database. They provide a simplified and customized perspective of the underlying data, enabling users to access and manipulate specific data sets without modifying the original tables

SQL Server	Dynamic SQL	Dynamic SQL in SQL Server allows for the generation and execution of SQL statements dynamically at runtime. This flexibility enables developers to construct SQL queries or commands based on varying conditions or parameters, enhancing the adaptability and functionality of applications
SQL Server	Control Flow Statements--If-else, While loop, Case statements	Control Flow statements in SQL Server are essential for implementing conditional logic and iterative processing within SQL scripts and stored procedures. These statements, including IF...ELSE, WHILE, and CASE, allow developers to control the flow of execution based on specified conditions or criteria
SQL Server	Stored Procedures----System Stored Procedures, User Defined Stored Procedures	Stored Procedures in SQL Server are precompiled and stored database objects that encapsulate SQL queries and procedural logic. They offer several benefits, including improved performance, reduced network traffic, and enhanced security by centralizing data access and manipulation tasks
SQL Server	Functions (String Functions, Date Functions, Mathematical Functions, Aggregate Functions, Ranking Functions, Conversion Functions, User Defined Functions)	Functions in SQL Server are reusable blocks of code that accept input parameters, perform calculations, and return a single value or a result set. They offer several advantages, such as code reuse, encapsulation of logic, and improved readability of queries.
SQL Server	Triggers (DML Triggers, DDL Triggers, After/For Triggers)	Triggers in SQL Server are database objects that automatically execute in response to specified events, such as INSERT, UPDATE, or DELETE operations on tables. They enable developers to enforce data integrity constraints, implement complex business logic, and maintain data consistency within the database.
SQL Server	Indexes--Clustered, Non-clustered, Covering Index, Filtered Index, Column Store Index	Indexes in SQL Server are data structures that improve the speed of data retrieval operations by providing quick access to specific rows within a table. They enhance query performance by organizing and optimizing the data storage, enabling faster search and retrieval of information
SQL Server	Error Handling	Error handling in SQL Server is crucial for ensuring the reliability, robustness, and integrity of database operations. It involves the implementation of mechanisms to detect, handle, and recover from errors or exceptions that may occur during database transactions or query execution

SQL Server	Partitions	Partitions in SQL Server enable efficient management and storage of large datasets by dividing tables and indexes into smaller, manageable segments. They enhance query performance, data loading, and maintenance tasks by distributing data across multiple physical or logical storage units
SQL Server	Transactions and Isolation Levels	Transactions and isolation levels in SQL Server are essential for maintaining data consistency, integrity, and reliability within database operations. Transactions allow developers to group multiple SQL statements into a single logical unit, ensuring that all changes are either committed or rolled back together