

DBM032: SQL Server 2019 Programming and Implementation

หลักการและเหตุผล:

This course provides the knowledge and skills to develop and implement a Microsoft SQL Server 2019 database

วัตถุประสงค์:

- Create databases and database files.
- Determine appropriate data types when designing table, covert data between data types, and create alias data types.
- Be aware of good design practices regarding SQL Server tables and be able to create tables using T-SQL.
- Determine appropriate single column and composite indexes strategies.
- Create tables as heaps and tables with clustered indexes. Also consider the design of a table and suggest an appropriate structure.
- Design effective non-clustered indexes.
- Implement PRIMARY KEY, FOREIGN KEY, DEFAULT, CHECK and UNIQUE constraints, and investigate cascading FOREIGN KEY constraints.
- Design and implement DML triggers
- Design and implement views
- Design and implement stored procedures.
- Use both traditional T-SQL error handling code and structured exception handling.
- Design and implement functions, both scalar and table-valued. (Also describe where they can lead to performance issues).Implement managed code in the database.
- Learn appropriate uses for SQL CLR integration and implement an existing .NET assembly within SQL Server.
- Perform basic investigation of a deadlock situation and learn how transaction isolation levels affect application concurrency.

หลักสูตรนี้เหมาะสำหรับ:

The primary audience for this course is individuals who develop and implement SQL Server databases.

ความรู้พื้นฐาน:

- Basic knowledge of the Microsoft Windows operating system and its core functionality.
- Basic knowledge of Transact-SQL.
- Basic knowledge of Relational Databases.

เนื้อหาหลักสูตร:

Module1: Creating Databases and Database Files

- Creating Databases
- Creating Filegroups
- Creating Schemas
- Creating Database Snapshots

Module 2: Working with Data Types and Tables

- Using Data Types
- Understanding Collations
- Converting Data Types
- Work with Specialized Data Types
- Creating and Altering Tables
- Surrogate Keys vs. Sequence
- Temporary Tables
- Compute Columns
- Creating Partitioned Tables for heavily table
- Using Switch, Merge and Split

Module3: Implementing Indexes

- The Need for Indexes
- Table Scan vs. Index Scan
- About Heap
- Implementing Clustered Indexes
- Primary key and Clustering key
- How to choose Clustering key
- Implementing NonClustered Indexes
- Consideration Data Types and Indexes
- About Execution Plan
- Implementing Columnstore Indexes
- Maintenance Index with Rebuild and Reorganize
- Fill factor and Pad Index
- Index Statistics concept
- Why need to update Index Statistics
- Optimizing Indexes

Module4: Ensuring Data Integrity through Constraints

- Enforcing Data Integrity
- Implementing Rules
- Implementing Default
- Implementing CHECK
- Implementing FOREIGN KEY
- Implementing NULL
- Implementing DDL Triggers
- Implementing DML Triggers

Module5: Implementing Views

- Introduction to Views
- Advantages of Views
- Types of Views
- Implementing Standard Views
- About Updatable Views
- Obfuscating View definitions
- Nested Views consideration
- Implementing Partition Views
- Implementing Indexes Views

Module6: Implementing Stored Procedures

- Introduction to Stored Procedures
- Benefit of Store Procedures
- Working with Store Procedures
- Guideline for Creating Store Procedures
- How to encrypt Store Procedures
- Implementing Parameterized Store Procedures
- About Parameter Sniffing and Performance impact
- Why we need recompile Store Procedures
- Controlling Execution Context
- How to raising error in Store Procedures
- Implementing Exception Handling in Store Procedures

Module7: Implementing User-Defined Functions

- Overview of Functions
- About System Functions
- Implementing Scalar Functions
- Deterministic vs. Non-deterministic Functions
- Implementing Table-valued Functions (TVF)
- Inline Table vs. Multi-statement Table-valued Functions
- Controlling execute context
- Guideline for Creating Functions
- How to choose TVF vs. Stored Procedures
- How to choose TVF vs. Views

Module8: Implementing Managed Code in SQL Server 2019

- Introduction to the SQL Server CLR Integration
- About .NET Framework Common Language Runtime
- Why use Managed Code in SQL Server
- T-SQL vs. Managed Code
- Importing and Configuring Assembly
- Implementing SQL CLR Integration

Module9: Creating Highly Concurrent SQL Server 2019 Applications

- What are Transactions
- Designing for Concurrency
- Controlling Transactions
 - Auto-Commit Transactions
 - Explicit Transactions
 - Implicit Transactions
- About XACT_ABORT
- Consideration for Using Transactions
- Management of Locking
- Method to View Locking Information
- Types of Locks
- About Concurrency problems
- Transaction Isolation Levels concept
- Implementing Read Uncommitted
- Implementing Read Committed
- Implementing Repeatable Read
- Implementing Serializable
- Implementing Snapshot

Career for the Future Academy: CFA

วิทยากร :



อาจารย์สุรัตน์ เกษมบุญศิริ

- วิทยากรรับเชิญประจำสถาบันพัฒนาบุคลากรแห่งอนาคต
- Managing Director (Born2Learn Co., Ltd)
- Microsoft Certified Trainer (MCT)
- Microsoft Certified System Engineer (MCSE)
- Microsoft Certified Solution Associate -Database
- Microsoft Certified Professional (MCP)
- Certified ITIL Foundation v.3
- Certified CompTIA Security +
- Cisco Certified Network Associate (CCNA)
- Certified AOTS of Japan Members

จำนวนชั่วโมงในการฝึกอบรม: 5 วัน (30 ชั่วโมง)

กำหนดการอบรม : ตามตารางปฏิทินอบรมประจำปี <https://www.career4future.com/trainingprogram>

ช่วงเวลาฝึกอบรม: 9.00 - 16.00 น.

ค่าลงทะเบียนอบรม : ท่านละ 13,500 บาท (รวมภาษีมูลค่าเพิ่มแล้ว)

** สถาบันฯ เป็นหน่วยงานราชการ จึงไม่อยู่ในเกณฑ์ที่ต้องถูกหักภาษี ณ ที่จ่าย

สถานที่ฝึกอบรม :

สถาบันพัฒนาบุคลากรแห่งอนาคต

เลขที่ 73/1 อาคารสำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ (สวทช.) ชั้น 6

ถนนพระรามที่ 6 แขวงทุ่งพญาไท เขตราชเทวี กรุงเทพฯ 10400

วิธีการสำรองที่นั่ง :

ติดต่อสำรองที่นั่งล่วงหน้า ในวัน-เวลาราชการ

โทรศัพท์: 0 2644 8150 ต่อ 81887

โทรสาร: 0 2644 8110

Website: www.career4future.com

E-mail: training@nstda.or.th