

SQL Server Specialist Certificate Program

Maintaining SQL Server 2005

**Week 3 – Creating Tables – Data
Integrity, and XML**

Steve Stedman - Instructor

This Weeks Overview

- Review from Last Week + Homework
 - Class Project
 - Creating Tables
 - Implementing Constraints
 - XML
-
- Review and Homework

Topics from last week

- Design and Normalization
- De-Normalization
- Views

Homework

- Practice
- Assignment
- Group Project
- Reading

Class Project

- Review Plans
- Determine Successful outcome
- Determine Next Steps

Finding more Information

- <http://www.sqlservercentral.com>

- End of this section. Any Questions?

Preparation

- Create a new database called Week3
- Use SSMS
 - Right Click on Database
 - Choose New Database
 - Give it a Name, use all the defaults

1. Creating Tables

- Data Types
- Data Type Sizes
- Nullability
- Identity
- Computed Columns
- Creating a Table

Data Types

- Int
- Decimal / Numeric
- Float / Real
- Money
- Datetime
- Binary / VarBinary
- Others

See Pages 109 to 113

Character Data Types

- Char / Nchar
- Varchar(n), Varchar (max)
- Nvarchar(n), Nvarchar(max)
- Text, Ntext
 - Not allowed with many operations (join)

See Pages 113, and 114

Specialized Data Types

- Bit (0, 1 or null)
- Timestamp
- Uniqueidentifier (16 bit GUID)
- Sql_Variant (can change depending on what is stored)
- XML

See Page 116

Data Type Sizes

- **Tiny**
- **Small**
- **Normal**
- **Big**

Nullability

- Null is not a value and does not consume storage
- Different from 0 or the empty string ""
- ISNULL (check_expression , replacement_value) to modify null values in output
- WHERE column_name IS NULL to test if a column is null

Identity

- Does not allow nulls
- Automatically defaults to the next highest number in ascending sequence
- Cannot insert to an identity column

Accessing the Identity value

```
DECLARE @InsertedRows AS TABLE (Id int)
DECLARE @NewId AS INT
INSERT INTO HumanResources.Employees
    ( /* column names */ )
OUTPUT Inserted.Id INTO @InsertedRows
VALUES ( /* column values */ )
SELECT @NewId = Id FROM @InsertedRows
```

Computed Columns

- For example, in the AdventureWorks sample database, the TotalDue column of the Sales.SalesOrderHeader table has the definition: `TotalDue AS Subtotal + TaxAmt + Freight`
- Calculated every time they are calculated unless PERSISTED
- Cannot be inserted or updated

Creating a Table

- Need to know:
 - Where to create it (database, schema, storage)
 - Table Name
 - Columns, data type, and nullability
- Script it or use SSMS

Temporary Tables

- Similar to standard tables, but the scope and lifetime are different.
- Temporary tables can be global or local
- # pound sign denotes temporary table
- ## double indicates global
- @ or @@ denotes a table variable
 - For functions, trigger or stored proc
- Global tables available to everyone
- Local tables can only be seen by the user who created them.

Lab Project – Creating Tables

- Using the Week 3 Simple Database Example Handout
- Create 2 tables to hold the data on the handout
 - People
 - Phone_Numbers
- Fill in some data
- Run the select statement as defined on the handout.

- End of this section. Any Questions?
- 10 Minute Break

1. Implementing Constraints

- Placed on tables or columns to implement business logic
- Types of constraints
 - Check Constraints
 - Rules
 - Default Constraints
 - Unique Constraints
 - Primary Keys
 - Foreign Key

Check Constraints

- Called on Insert and Update, but not called for Delete
- Used to validate single columns or multiple columns in a table
- Examples
 - Validate Social Security Number format
 - Validate Email Address

Rules

- Similar functionality as a check constraint, but stored separately
- Like a stored or function
- Use sp_bindrule stored proc to bind it to a column

Default Constraints

- Applies to one or more columns in a table
- Fills in a column if it is not specified on Insert
- Does not apply to Select, Update or Delete

Unique Constraints

- Uses an index to enforce uniqueness
- Examples
 - Employee ID
 - Student ID
 - Social Security Number
 - Drivers License Number

Primary Keys

- Columns or combination of columns to allow a rows to be uniquely identified
- Only one primary key per table
- **Clustered Index** automatically created for each primary key
 - Good for integer
 - Bad if uniqueidentifier

Foreign Key

- Links to a primary key in another table
- Value must exist in another table
- Prevents deleting of the referenced row or table

Lab Project –

- Using the tables created in the earlier lab, add one or more of the following constraints.
 - Primary Key
 - Unique Constraint
 - Default Values

- End of this section. Any Questions?

3. XML

- What is XML?
- Why Use Relational Databases for XML Data?
- Formatting Query Output

What is XML?

- Extensible Markup Language
- Aid with Structured Data
- Commonly used to exchange data between systems

XML Sample

```
<recipe name="bread" prep_time="5 mins" cook_time="3 hours">
<title>Basic bread</title>
<ingredient amount="8" unit="dL">Flour</ingredient>
<ingredient amount="10" unit="grams">Yeast</ingredient>
<ingredient amount="4" unit="dL" state="warm">Water</ingredient>
<ingredient amount="1" unit="teaspoon">Salt</ingredient>
<instructions>
<step>Mix all ingredients together.</step>
<step>Knead thoroughly.</step>
<step>Cover with a cloth, and leave for one hour in warm room.</step>
<step>Knead again.</step>
.....
</instructions>
</recipe>
```

Why Use Relational Databases for XML Data?

- XML is not relational
- Can be indexed and queried, but it is slow
- Schema is parsed at run time

Formatting Query Output

```
select LoginID, Title, Gender from Humanresources.employee
```

```
DECLARE @xmlDoc XML
```

```
SET @xmlDoc =
```

```
(
```

```
select LoginID, Title, Gender from Humanresources.employee
```

```
FOR XML AUTO, ELEMENTS
```

```
)
```

```
SELECT @xmlDoc
```

Lab Project

- Build a query to access the tables in the previous labs
- Convert the output from those queries into XML.
- Similar to the example below:

```
DECLARE @xmlDoc XML
SET @xmlDoc =
(
select LoginID, Title, Gender from Humanresources.employee
FOR XML AUTO, ELEMENTS
)
SELECT @xmlDoc
```

- End of this section. Any Questions?

Homework

- Homework for next week
 - Handout

Questions?