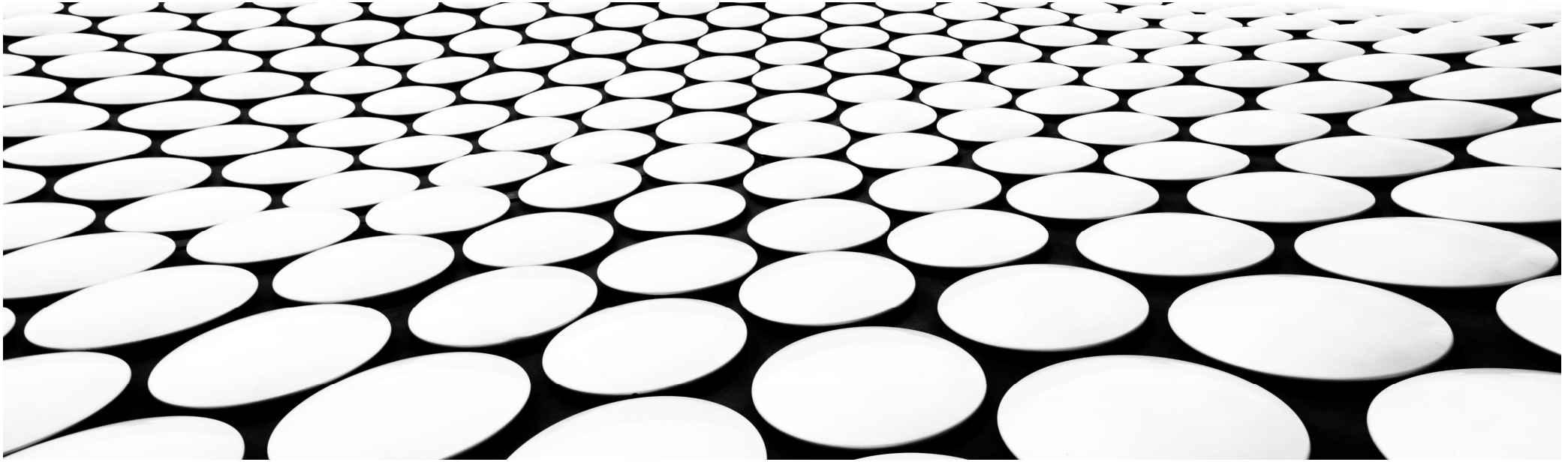
SQL SERVER

ARCTECH INFO

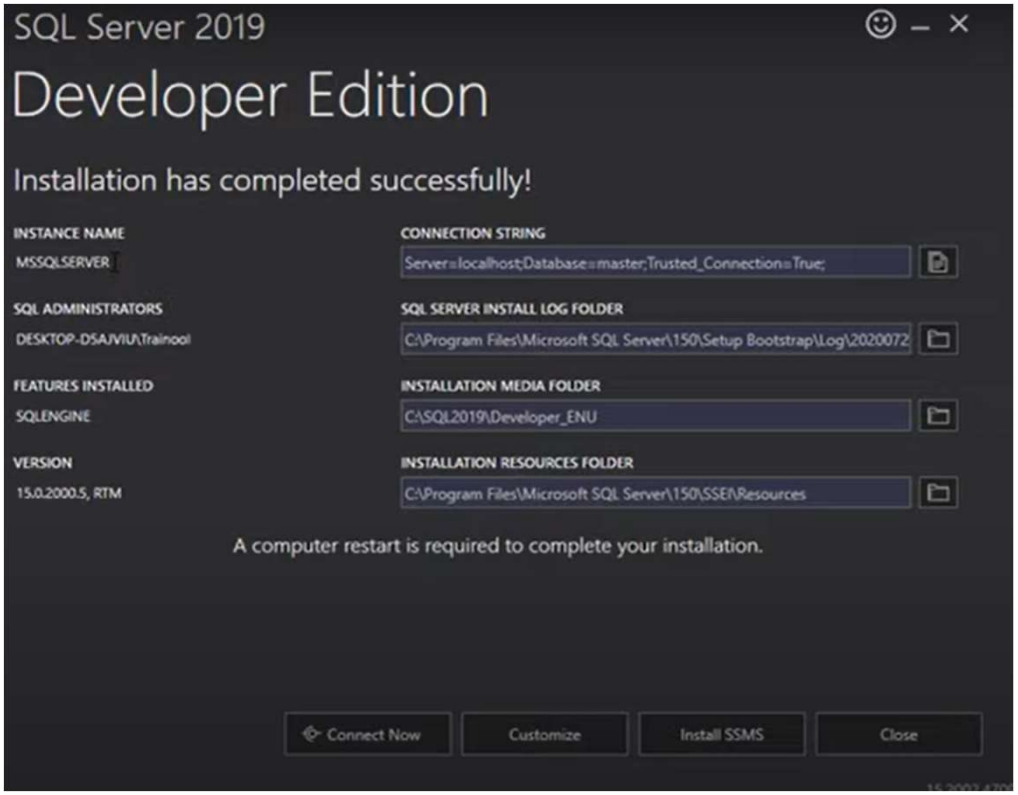


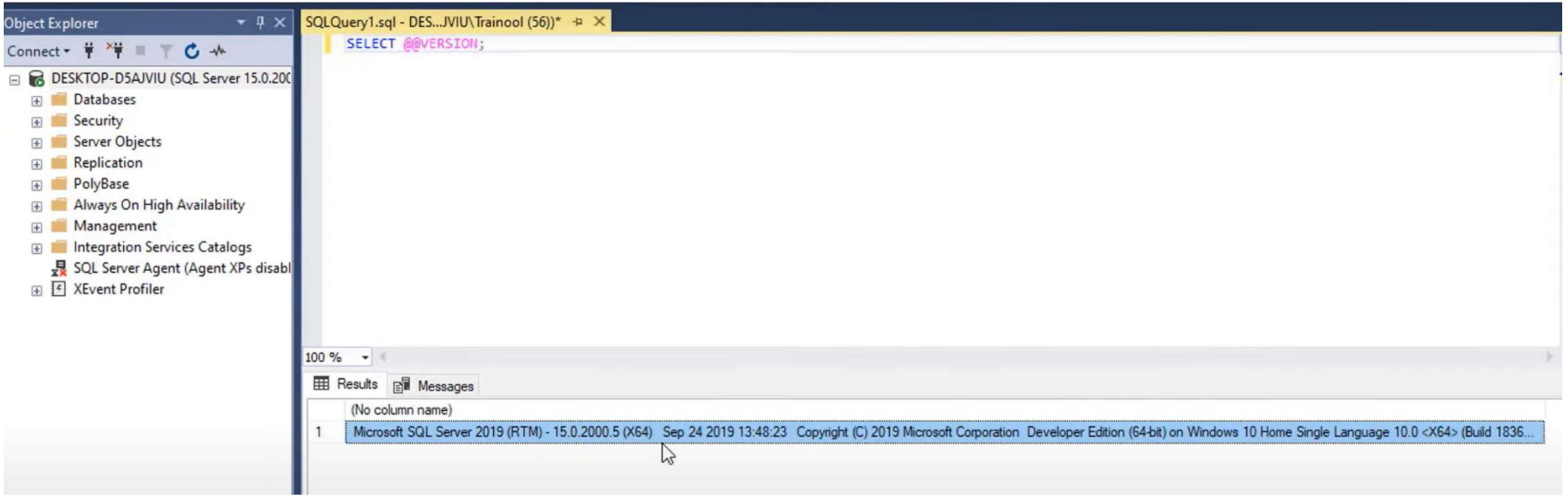
WHAT IS SQL SERVER

- SQL stands for Structure Query Language
- It is a language used by relational databases to SELECT, INSERT, UPDATE and DELETE (DML commands - Data Manipulation Language) data for most of the database platforms like Oracle, SQL Server, MySQL, PostgreSQL, etc. that use the SQL language to query data and manipulate it.
- SQL Server is a relational database management system, or RDBMS, developed and marketed by Microsoft.
- Similar to other RDBMS software, SQL Server is built on top of SQL, a standard programming language for interacting with relational databases.
- SQL Server is tied to Transact-SQL, or T-SQL, the Microsoft's implementation of SQL that adds a set of proprietary programming constructs

INSTALLATION

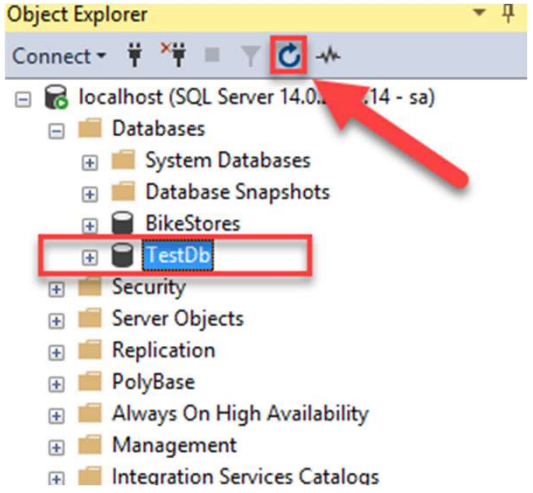
- Download SQL server.
  - https://www.microsoft.com/en-in/sql-server/sql-server-downloads
  - Select developer edition
  - Click on download now
  - Installation type – BASIC
- Download SSMS
  - https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver15
  - After installation restart machine
  - Query to check version – SELECT @@VERSION;



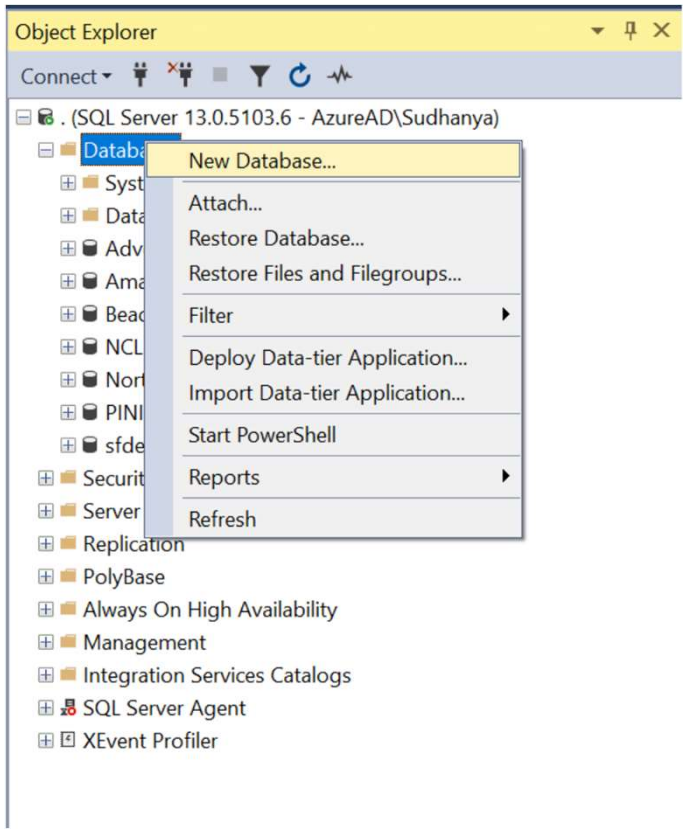

Object Explorer
Connect
DESKTOP-D5AJVIU (SQL Server 15.0.200
Databases
Security
Server Objects
Replication
PolyBase
Always On High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs disabl
XEvent Profiler
SQLQuery1.sql - DES...JVIU\Trainool (56))*
SELECT @@VERSION;
100 %
Results
Messages
(No column name)
1
Microsoft SQL Server 2019 (RTM) - 15.0.2000.5 (X64) Sep 24 2019 13:48:23 Copyright (C) 2019 Microsoft Corporation Developer Edition (64-bit) on Windows 10 Home Single Language 10.0 <X64> (Build 1836...

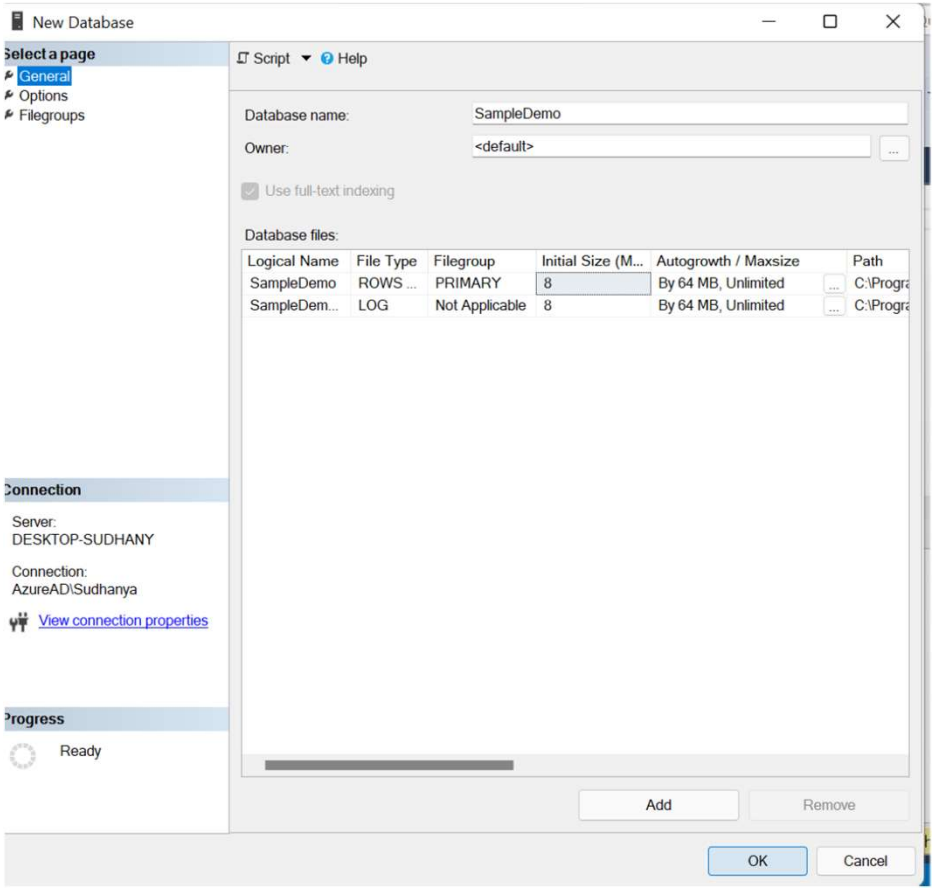
CREATE NEW DATABASE

- With Query
  - CREATE DATABASE database_name;
  - In this syntax, you specify the name of the database after the CREATE DATABASE keyword.
  - The database name must be unique within an instance of SQL Server. It must also comply with the SQL Server identifier's rules. Typically, the database name has a maximum of 128 characters.
  - The following statement creates a new database named TestDb
  - CREATE DATABASE TestDb;



CREATE NEW DATABASE USING SQL SERVER MANAGEMENT STUDIO





LIST DATABASES

- This statement lists all databases in the SQL Server:

```
SELECT
    name
FROM
    master.sys.databases
ORDER BY
    name;
```

SQL SERVER DATA TYPES - STRING DATA TYPES

DATA TYPE	DESCRIPTION	MAX SIZE	STORAGE
char(n)	Fixed width character string	8,000 characters	Defined width
varchar(n)	Variable width character string	8,000 characters	2 bytes + number of chars
varchar(max)	Variable width character string	1,073,741,824 characters	2 bytes + number of chars
text	Variable width character string	2GB of text data	4 bytes + number of chars
nchar	Fixed width Unicode string	4,000 characters	Defined width x 2
nvarchar	Variable width Unicode string	4,000 characters	
nvarchar(max)	Variable width Unicode string	536,870,912 characters	
ntext	Variable width Unicode string	2GB of text data	
binary(n)	Fixed width binary string	8,000 bytes	
varbinary	Variable width binary string	8,000 bytes	
varbinary(max)	Variable width binary string	2GB	
image	Variable width binary string	2GB	

SQL SERVER DATA TYPES - NUMERIC DATA TYPES

Data type	Description	Storage
bit	Integer that can be 0, 1, or NULL	
tinyint	Allows whole numbers from 0 to 255	1 byte
smallint	Allows whole numbers between -32,768 and 32,767	2 bytes
int	Allows whole numbers between -2,147,483,648 and 2,147,483,647	4 bytes
bigint	Allows whole numbers between -9,223,372,036,854,775,808 and 9,223,372,036,854,775,807	8 bytes
decimal(p,s)	Fixed precision and scale numbers.Allows numbers from -10^38 +1 to 10^38 –1. The p parameter indicates the maximum total number of digits that can be stored (both to the left and to the right of the decimal point). p must be a value from 1 to 38. Default is 18. The s parameter indicates the maximum number of digits stored to the right of the decimal point. s must be a value from 0 to p. Default value is 0	5-17 bytes
numeric(p,s)	Fixed precision and scale numbers.Allows numbers from -10^38 +1 to 10^38 –1. The p parameter indicates the maximum total number of digits that can be stored (both to the left and to the right of the decimal point). p must be a value from 1 to 38. Default is 18. The s parameter indicates the maximum number of digits stored to the right of the decimal point. s must be a value from 0 to p. Default value is 0	5-17 bytes
smallmoney	Monetary data from -214,748.3648 to 214,748.3647	4 bytes
money	Monetary data from -922,337,203,685,477.5808 to 922,337,203,685,477.5807	8 bytes
float(n)	Floating precision number data from -1.79E + 308 to 1.79E + 308.The n parameter indicates whether the field should hold 4 or 8 bytes. float(24) holds a 4-byte field and float(53) holds an 8-byte field. Default value of n is 53.	4 or 8 bytes
real	Floating precision number data from -3.40E + 38 to 3.40E + 38	4 bytes

SQL SERVER DATA TYPES - DATE AND TIME DATA TYPES

Data type	Description	Storage
datetime	From January 1, 1753 to December 31, 9999 with an accuracy of 3.33 milliseconds	8 bytes
datetime2	From January 1, 0001 to December 31, 9999 with an accuracy of 100 nanoseconds	6-8 bytes
smalldatetime	From January 1, 1900 to June 6, 2079 with an accuracy of 1 minute	4 bytes
date	Store a date only. From January 1, 0001 to December 31, 9999	3 bytes
time	Store a time only to an accuracy of 100 nanoseconds	3-5 bytes
datetimeoffset	The same as datetime2 with the addition of a time zone offset	8-10 bytes
timestamp	Stores a unique number that gets updated every time a row gets created or modified. The timestamp value is based upon an internal clock and does not correspond to real time. Each table may have only one timestamp variable	

CREATE TABLE

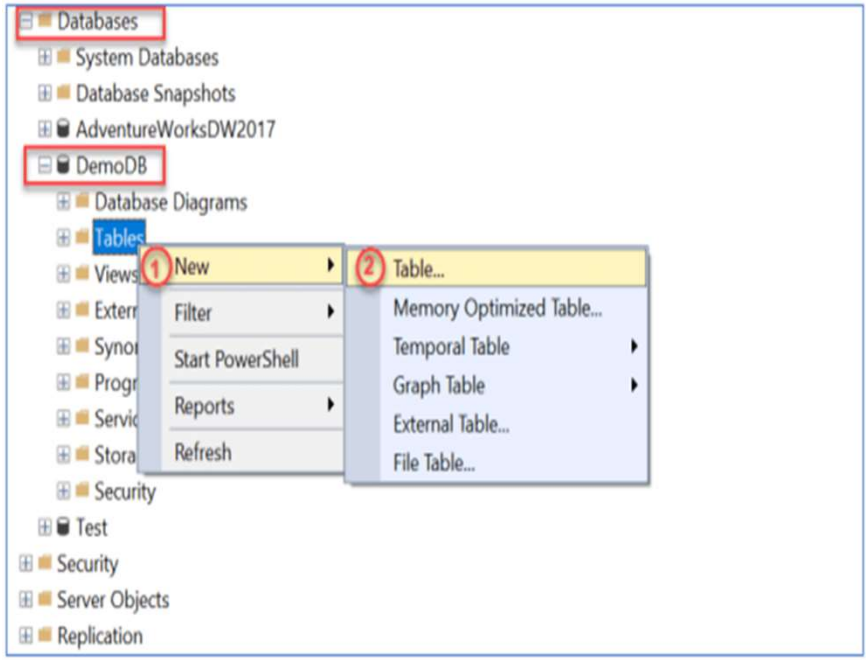
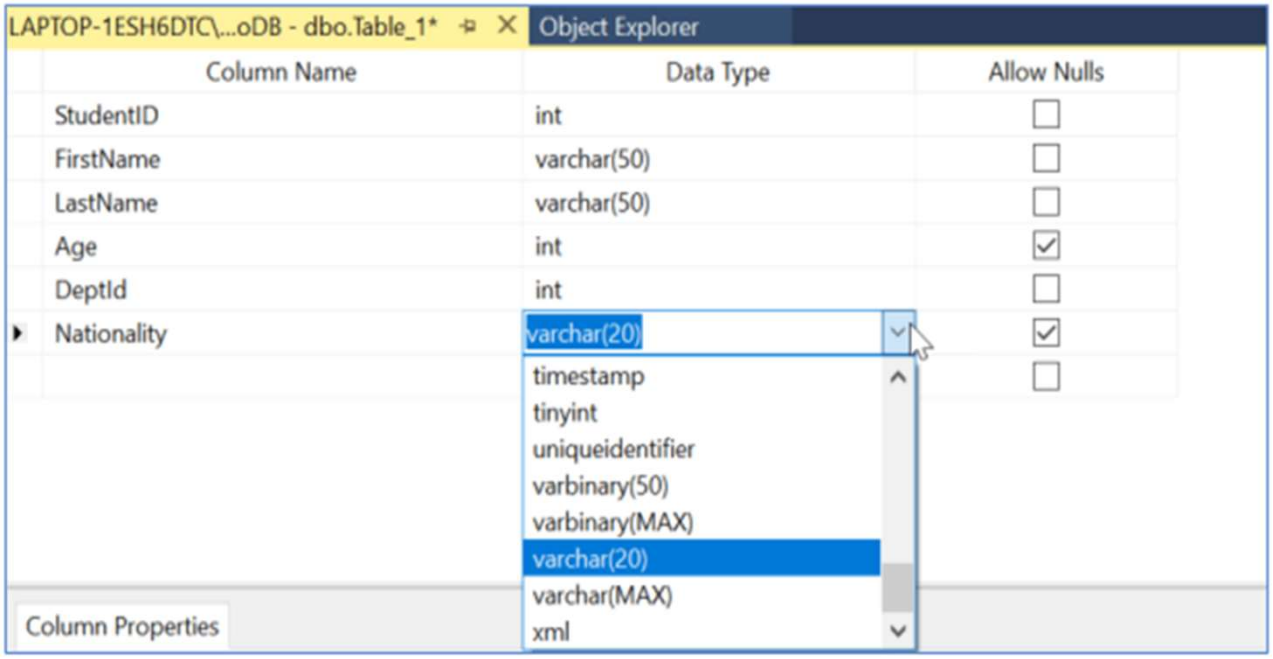
- The CREATE TABLE statement is used to create a new table in a database.
- Syntax

```
CREATE TABLE table_name (
    column1 datatype,
    column2 datatype,
    column3 datatype,
   ....
);
```

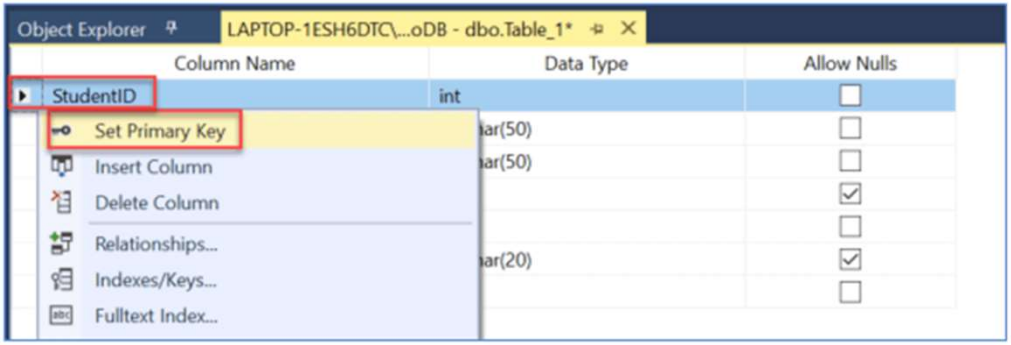
- The column parameters specify the names of the columns of the table.
- The datatype parameter specifies the type of data the column can hold (e.g. varchar, integer, date, etc.).

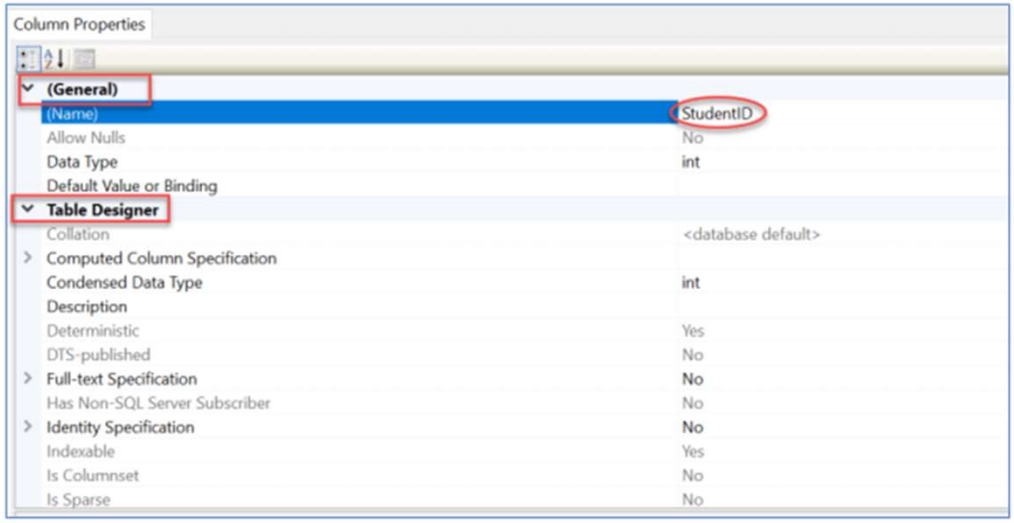
SAMPLE CREATE TABLE - QUERY

- ```
  CREATE TABLE Persons (
      PersonID int,
      LastName varchar(255),
      FirstName varchar(255),
      Address varchar(255),
      City varchar(255)
  );
  ```

- Create table using management studio
  ```

CREATE SQL SERVER TABLE WITH SSMS TABLE DESIGNER





PRIMARY KEY

- The PRIMARY KEY constraint uniquely identifies each record in a table.
- Primary keys must contain UNIQUE values, and cannot contain NULL values.
- A table can have only ONE primary key; and in the table, this primary key can consist of single or multiple columns (fields).
- Primary key can be created for a table while creating the table or table can be modified later to add primary key
- Primary key can be added to table either using query or from SSMS designer

PRIMARY KEY

- Setting primary key while creating table using query –
  - ```
    CREATE TABLE Persons (
        ID int NOT NULL PRIMARY KEY,
        LastName varchar(255) NOT NULL,
        FirstName varchar(255),
        Age int
    );
    ```
- Setting the primary key later in the table
  - ```
    ALTER TABLE Persons
    ADD CONSTRAINT PK_Person PRIMARY KEY (ID,LastName);
    ```
- Drop a primary key constraint
  - ```
    ALTER TABLE Persons
    DROP CONSTRAINT PK_Person;
    ```
    ```

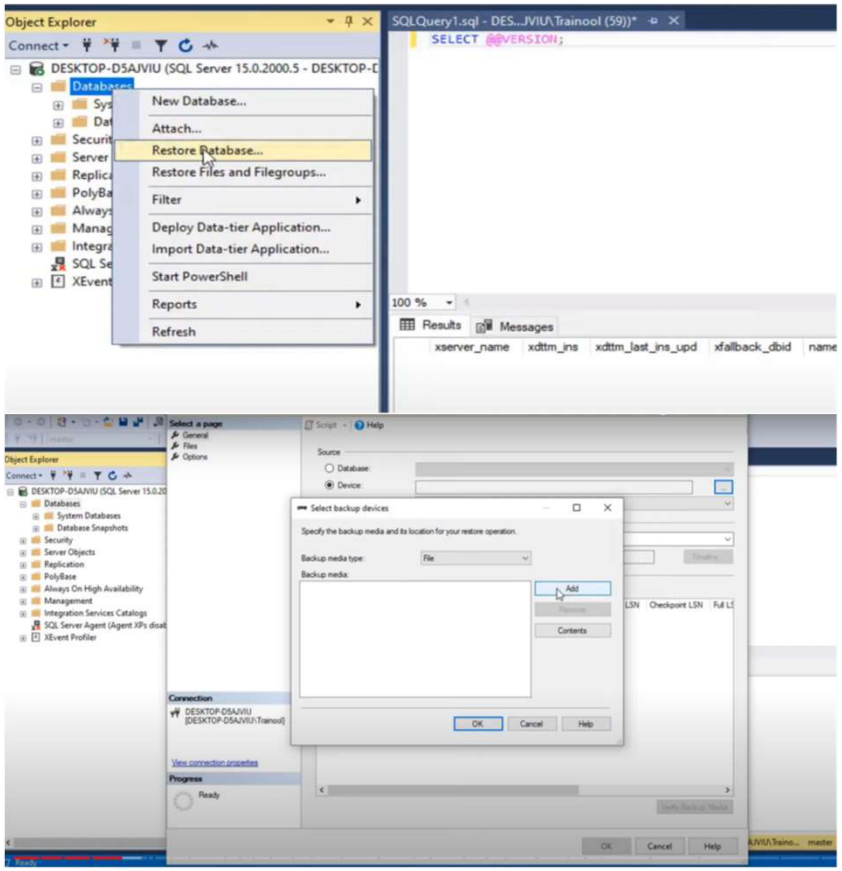
CREATE PRIMARY KEY USING SSMS





ADVENTURE WORKS SAMPLE DATABASE DOWNLOAD AND IMPORT

- Download database
  - https://docs.microsoft.com/en-us/sql/samples/adventureworks-install-configure?view=sql-server-ver15&tabs=ssms
  - **OLTP** data is for most typical online transaction processing workloads
  - AdventureWorks2016.bak
- Restore Database
  - In SSMS right click on databases and select – Restore database
  - Under source – select option "Device"
  - Select the downloaded database location in file system



SELECT QUERY

- Retrieves rows from the database and enables the selection of one or many rows or columns from one or many tables. The full syntax of the SELECT statement is complex, but the main clauses can be summarized as:

  [WITH <common_table_expression>]

  SELECT select_list [INTO new_table]

  [FROM table_source] [WHERE search_condition]

  [GROUP BY group_by_expression]

  [HAVING search_condition]

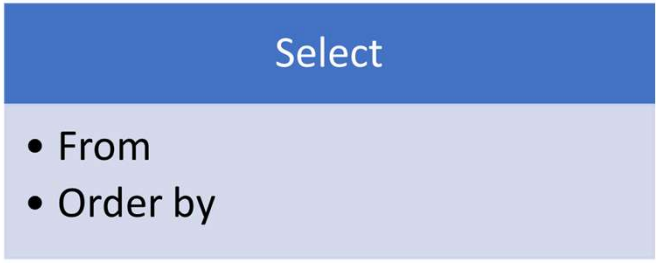
  [ORDER BY order_expression [ASC | DESC]]

  The UNION, EXCEPT and INTERSECT operators can be used between queries to combine or compare their results into one result set.

PROCESSING ORDER OF THE SELECT STATEMENT

- The following steps show the processing order for a SELECT statement.
  1. FROM
  2. ON
  3. JOIN
  4. WHERE
  5. GROUP BY
  6. WITH CUBE or WITH ROLLUP
  7. HAVING
  8. SELECT
  9. DISTINCT
  10. ORDER BY
  11. TOP

SQL QUERIES AND THE SELECT SENTENCE



- The following query will show all the columns from a table:

```
SELECT *
FROM [HumanResources].[Employee]
```

- Try to use the SELECT statement in one line and the FROM statement in a different line. It is easier to re ad that way. Select * means to show all the columns from a table. Another way to do the same is the fol lowing example:

```
SELECT [Employee].*
FROM [HumanResources].[Employee]
```

- The square brackets are optional. They could help if the column names have spaces (which is not reco mmended). You can also select specific column names like this:

SELECT QUERY CONTINUED

- The example below shows the loginid and gender column. As you can see, the data is separated by commas.

```
SELECT [LoginID],[Gender]
FROM [HumanResources].[Employee]
```

- You can also use aliases to have a shorter name like this:

```
SELECT e.Gender
FROM [HumanResources].[Employee] e
```

- **DISTINCT** is a slow command and if the table has several millions of rows, it could take time to execute and co uld bog down performance.

```
SELECT DISTINCT e.Gender g
FROM [HumanResources].[Employee] e
```

SELECT QUERY CONTINUED

- TOP clause

```
SELECT TOP 10 e.[BusinessEntityID], e.Gender g
FROM [HumanResources].[Employee] e
```

- If we want to order data by a column, the order by is very useful. The following example will show how to show the BusinessEntityID sorted in descending order.

```
SELECT [BusinessEntityID]
FROM [HumanResources].[Employee] e
ORDER BY [BusinessEntityID] desc
```

SQL QUERIES TO FILTER DATA USING THE WHERE COMMAND

Select
• From • Where • like/exists/in/not/or/and/any/etc.

- The where command is one of the most common clauses used inside the SELECT command. This clause allows filtering data. The following example shows how to check the BusinessEntityID of the employees whose job title is Design Engineer.

```
select [BusinessEntityID], [JobTitle]
from [HumanResources].[Employee] e
where JobTitle='Design Engineer'
```

- Another powerful operator is the LIKE. Like, can help us in a search. The following example shows the BusinessEntityID and the Job Title of the employees whose titles start with Design:

```
SELECT [BusinessEntityID], [JobTitle]
FROM [HumanResources].[Employee] e
WHERE JobTitle LIKE 'Design%'
```

- The IN operator is a very common operator also, the following example will show all the employees whose JobTitle are equal to Engineering Manager or Senior Tool Designer:

```
SELECT [BusinessEntityID],JobTitle
FROM [HumanResources].[Employee] e
WHERE JobTitle in ('Engineering Manager','Senior Tool Designer')
```

% WILDCARD CHARACTER

- The % wildcard character represents zero, one, or multiple characters
- The query below returns all telephone numbers that have area code “415” in the “PersonPhone” table:

```
SELECT p.FirstName, p.LastName, ph.PhoneNumber
FROM Person.PersonPhone AS ph
WHERE ph.PhoneNumber LIKE '415%'
GO
```

- The Not logical operator reverses the value of any Boolean expression.

```
SELECT p.FirstName, p.LastName, ph.PhoneNumber
FROM Person.PersonPhone AS ph
WHERE ph.PhoneNumber NOT LIKE '415%'
```

- let's say that we want to find all the records where a name contains "ssa" in them. We can use the query below:

```
SELECT p.FirstName, p.LastName
FROM Person.Person AS p
WHERE LastName LIKE '%ssa%'
ORDER BY p.LastName;
GO
```

- Notice that by using '%' before and after "ssa", we are telling the SQL Server to find all records in which "Person .Person" has "ssa" characters and it doesn't matter what other characters are before and after "ssa"

“_” WILDCARD CHARACTER

- “_” wildcard character represents a single character
- The SQL Like underscore character e.g. „ is used when we want to check a single character that can be anything and provide the rest of the characters for our match.
- Let’s say that if we want to return all records wherein the “FirstName” table first character can be anything but rest of them should be “en”. Use the query below:

```
SELECT p.FirstName,
    p.MiddleName
FROM Person.Person p
WHERE p.FirstName LIKE '_en';
GO
```