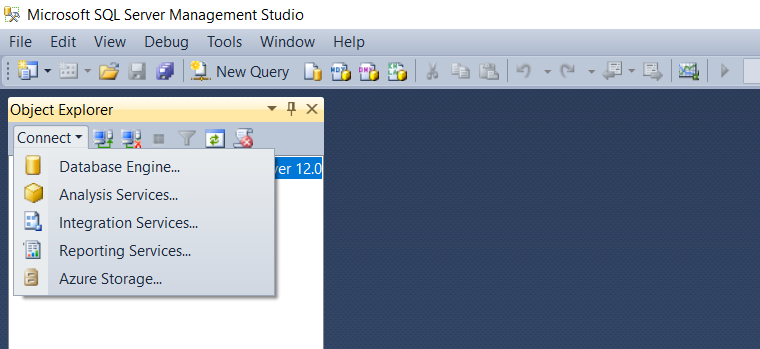
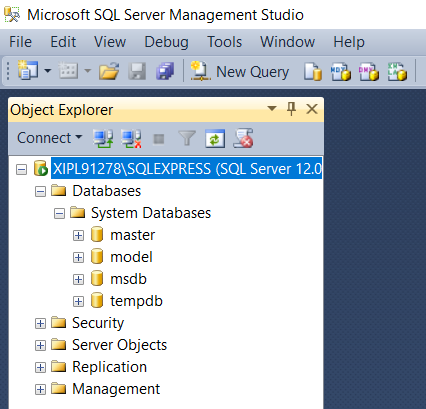
SQL 101

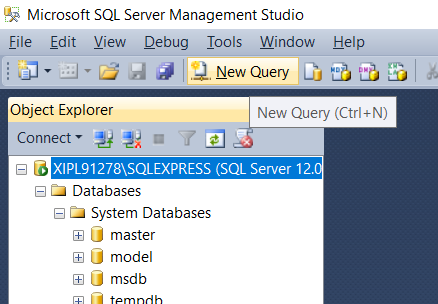
1. Select Database Engine and type server name to connect.



1. To check database, select the Databases



1. For querying, select “New Query”



1. To Create Database:

***create database mytestdb***

press F5 to run query after selecting

1. Select Database to work with:

**Use mytestdb**

1. To Create Table:

**Create table mytesttable (rollno int, firstname varchar(50), lastname varchar (50))**

1. Entering Data in Table

**Insert into mytesttable(rollno, firstname, lastname)**

**Values (1, “Shounak”, “Deshpande”)**

**Values (2, “Sharvari”, “Ballal”)**

1. Retrieving data from database (querying):

**Select rollno, firstname from mytesttable** – this selects only rollno, first name and lastname from mytesttable

1. Select everything from the table

**Select \* from mytesttable**

Working with AdventureWords database

1. **Select name from HumanResources.Department** – gives only the name column from the table
2. To get unique values:

**Select distinct name from HumanResources.Department**

1. To filter values, use “WHERE” clause

Example1 TEXT: show all rows where GroupName == “Manufacturing”

**Select \* from HumanResources.Department where GroupName like “Manufacturing”**

Example2 INTEGER: show all employees where OrganizationLevel == 2

**Select \* from HumanResources.Employee where OrganizationLevel = 2**

Example3 INTEGER: show all employees where OrganizationLevel == 2 or 3

**Select \* from HumanResources.Employee where OrganizationLevel in (2,3)**

Example4 TEXT: show all rows with jobtitle has the word “manager” in it

Select \* from show all employees where OrganizationLevel == 2

**Select \* from HumanResources.Employee where jobtitle like “%Manager%”**

If you want the cases where the “Manager” word is at the end:

**Select \* from HumanResources.Employee where jobtitle like “%Manager”**

If you want the cases where the “Manager” word is at the start:

**Select \* from HumanResources.Employee where jobtitle like “Manager%”**

Example5 DATE: show all rows where date of birth is after 31 Dec 1980

**Select \* from HumanResources.Employee where birthdate > ‘12/31/1980’**

show all rows where date of birth is before 31 Dec 1980

**Select \* from HumanResources.Employee where birthdate < ‘ 12/31/1980’**

show all rows where date of birth is between 01 Jan 1975 and 31 Dec 1980

**Select \* from HumanResources.Employee where birthdate between ‘01/01/1975’ and ‘12/31/1980’**

1. Calculated Tables: if you have 2 columns and you want to create 3rd column which is based upon mathematical operations of these 2 columns, you can do so by giving Alias Name. Alias Name will be the name of the new column that gets added

**Select \*, listprice+10 as new\_column from Production.Product**

1. Adding new table based on query operations (new table name = New\_Table)

**Select Name, ListPrice, ListPrice + 10 as new\_column into Production.New\_Table** **from Production.Product**

This creates a permanent table which gets reflected in the database.

If you want to create a temporary table (which disappears after the server is closed), just add # before the table name i.e. instead of New\_Table, use #New\_Table

1. Delete Data in Table:

**Delete from Production.Product where name like “Bearing Ball”**

1. Update Data in Table:

**Update** **Production.Product set name = “blade\_new” where name = “blade”**

1. Joining Tables

There are 3 types of join: Inner, Outer and Cross

Sample Tables:

Create table Table1 (ID INT, FirstName VARCHAR(20), LastName VARCHAR(20))

Insert into Table1 Values(1, “Michael”,”Scott”)

Insert into Table1 Values(2, “Jim”,”Halpert”)

Insert into Table1 Values(3, “Dwight”,”Schrute”)

Create table Table2 (ID INT, Salary INT)

Insert into Table2 Values(1, 10000)

Insert into Table2 Values(2, 8000)

Insert into Table2 Values(3, 7000)

Create table Table3 (ID INT, Number INT)

Insert into Table3 Values(1, 1211211121)

Insert into Table3 Values(2, 6565656565)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Table1 | | |  | Table2 | |  | Table3 | |
| **ID** | **FirstName** | **LastName** |  | **ID** | **Salary** |  | **ID** | **Number** |
| 1 | Michael | Scott |  | 1 | 10000 |  | 1 | 1211211121 |
| 2 | Jim | Halpert |  | 2 | 8000 |  | 2 | 6565656565 |
| 3 | Dwight | Schrute |  | 3 | 7000 |  | 4 | 9999999999 |

***Inner Join:*** *There should be at least one common column between the tables*

**Select A.FirstName, A.LastName, B.Salary from Table1 A INNER JOIN Table2 B on A.ID = B.ID**

Result:

|  |  |  |
| --- | --- | --- |
| **First Name** | **Last Name** | **Salary** |
| Michael | Scott | 10000 |
| Jim | Halpert | 8000 |
| Dwight | Schrute | 7000 |

***Outer Join:*** *Left Outer Join, Right Outer Join, Full Outer Join*

*Left Outer Join*

Select A.FirstName, A.LastName, B.Number from Table1 A LEFT JOIN Table3 B on A.ID = B.ID

Result:

|  |  |  |
| --- | --- | --- |
| **First Name** | **Last Name** | **Number** |
| Michael | Scott | 1211211121 |
| Jim | Halpert | 6565656565 |
| Dwight | Schrute | null |

*Right Outer Join*

Select A.FirstName, A.LastName, B.Number from Table1 A RIGHT JOIN Table3 B on A.ID = B.ID

Result:

|  |  |  |
| --- | --- | --- |
| **First Name** | **Last Name** | **Number** |
| Michael | Scott | 1211211121 |
| Jim | Halpert | 6565656565 |
| null | null | 9999999999 |

*Full Outer Join*

Creates full table and replaces missing values with null

**Select A.FirstName, A.LastName, B.Number from Table1 A FULL OUTER JOIN Table3 B on A.ID = B.ID**

|  |  |  |
| --- | --- | --- |
| **First Name** | **Last Name** | **Number** |
| Michael | Scott | 1211211121 |
| Jim | Halpert | 6565656565 |
| Dwight | Schrute | null |
| null | null | 9999999999 |

***Cross Join:*** *Each row of Table1 connects with every row of Table2*

**Select \* from Table1 cross join Table3**

|  |  |  |
| --- | --- | --- |
| **First Name** | **Last Name** | **Number** |
| Michael | Scott | 1211211121 |
| Michael | Scott | 6565656565 |
| Jim | Halpert | 1211211121 |
| Jim | Halpert | 6565656565 |
| Dwight | Schrute | 1211211121 |
| Dwight | Schrute | 6565656565 |

1. Dealing with Dates:

**Select GETDATE()** – gives today’s date

**Select GETDATE() - n** (where n is integer) – Gives date ‘n’ days before today’s date

**Select GETDATE() + n** (where n is integer) – Gives date ‘n’ days after today’s date

# Get only the year, month, day from a date

**Select DATEPART (yyyy, GETDATE()) as Year** – returns the year in today’s date in “Year” column

**Select DATEPART (mm, GETDATE()) as Month** – returns the month in today’s date in “Month” column

**Select DATEPART (dd, GETDATE()) as Day** – returns the day in today’s date in “Day” column

Add Year/Month/Day from the date

**Select DATEADD (day, 4, GETDATE())** – adds 4 days to today’s date

**Select DATEADD (day, 4, ‘10/31/2020’)** – adds 4 days to 31 October 2020

**Select DATEADD (month, 4, ‘10/31/2020’)** – adds 4 months to 31 October 2020

**Select DATEADD (year, 4, ‘10/31/2020’)** – adds 4 years to 31 October 2020

Get difference between the dates using DATEDIFF(day/month/year, START\_DATE, END\_DATE)

**Select \*, DATEDIFF(day, StartDate, EndDate) from Production.WorkOrder**

1. Aggregate Functions:

|  |  |
| --- | --- |
| Table2 | |
| **ID** | **Salary** |
| 1 | 10000 |
| 2 | 8000 |
| 3 | 7000 |

To get average of salaries from Salary column:

**Select AVG(SALARY) from Table2**

To get count of row entries from Salary column:

**Select COUNT(SALARY) from Table2**

To get sum of salaries form Salary column:

**Select SUM(SALARY) from Table2**

To get minimum of salaries form Salary column:

**Select MIN(SALARY) from Table2**

To get maximum of salaries form Salary column:

**Select MAX(SALARY) from Table2**

1. String Functions:

Concatenate String:

**Select ordernumber, ordername, concat(ordernumber, “ ”, ordername) as concatedtext\_column from myorder**

(table name == myorder, columns in table == ordernumber, ordername)

This concatenates the elements in columns ordernumbe & ordername with space “ “ and is shown in column concatedtext\_column.

**RAND()** – returns a random float between 0 and 1

Select only the first “n” or last “n” letters from the string using LEFT & RIGHT resp.

**Select \*, LEFT(ordername, 5) from myorder** – selects 5 letters from left of element

**Select \*, RIGHT(ordername, 5) from myorder** – selects 5 letters from right of element

**Select \*, SUBSTRING(ordername, 2, 5) from myorder** – gives letters from 2nd position till 5th letter to right from 2nd position.

Convert to lower case

**Select ordernumber, ordername, upper(ordername) from myorder**

Convert to upper case

**Select ordernumber, ordername, upper(ordername) from myorder**

Get the length of element (using LEN)

**Select ordernumber, ordername, LEN(ordername) from myorder** – gives length of element in ordername column

If you want to make first letter capital and rest lower

**Select \*, concat(upper(left(ordername, 1)), lower(substring(ordername,2,len(ordername)))) from order**

Trim function – removes whitespaces

Remove whitespaces from left side

Select \*, LTRIM(odername) from myorder

Remove whitespaces from right side

Select \*, RTRIM(odername) from myorder