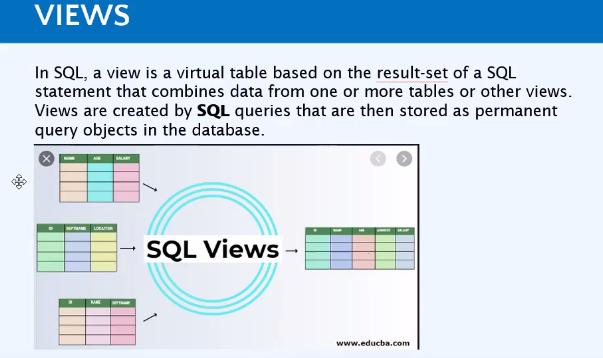
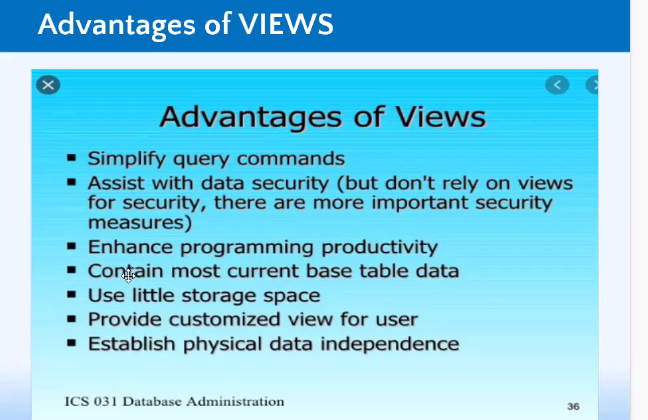
**VIEWS(Oct\_15)-Mr. Ephraim**

-Let’s suppose developers want an info from a table or several tables but you don’t want to give them access to all info, just the columns they need. What you will do is to create a query for a select statement and store it as a view, and grant them access to the view.

-The output of the view is a table, reason why modifications in the table will reflect in the view.

-Though views help in simplifying query commands and security, do not rely on views for security as there are other better security measures.





**TO CREATE A VIEW**

CREATE VIEW NgohOsee (name of view)

AS Select FirstName from Student (table name)

-Any time you want to access the view,

Select \* from NgohOsee (view name)

**-**

**STORE PROCEDURE**

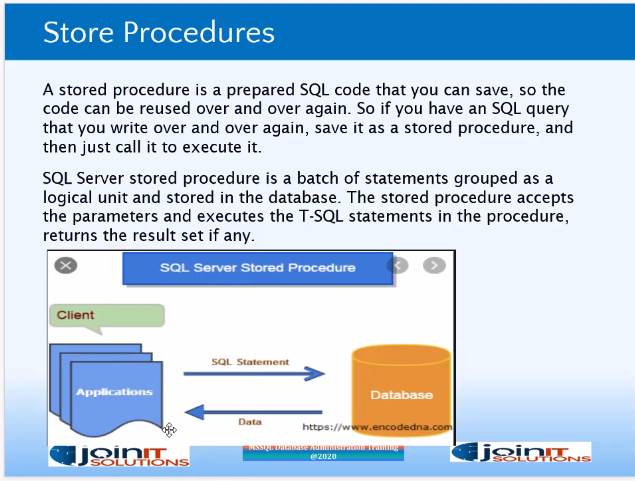
-Information get into the database through applications.

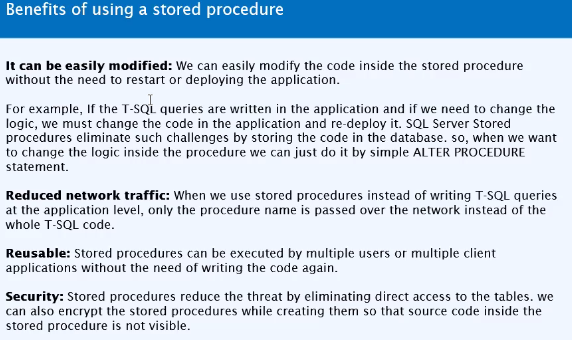
-As a DBA you want input info manually into the db, there is an automated process that does that, via the applications.

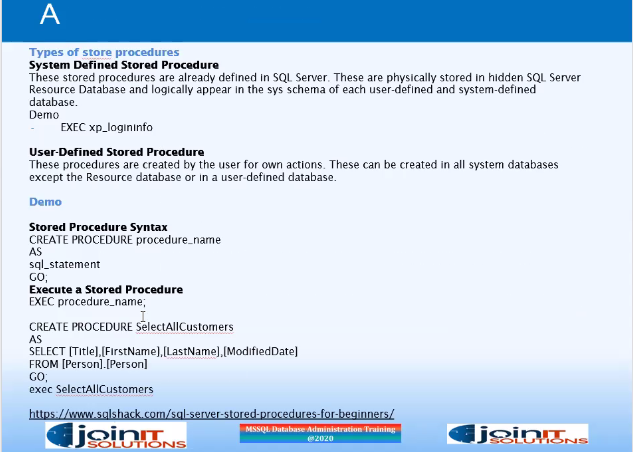
-If you have a sql query that you run over and over, similar to an item (say shoes) people go to search on amazon over and over, the item is saved as an input parameter.

-That is how store procedures run on applications.

-Unlike view when we want to query, we use the SELECT statement, with store procedure we use the **Execute procedure name** to call that store procedure







**Question: If we have a table that has so many names and we want to get any particular name each time we can edit the sp to include a parameter by using say the ID column.**

CREATE PROCEDURE(or Proc) JIT\_Employee @ID INT

AS

SELECT \* FROM Employee

WHERE ID = @ID

GO

- --CREATE STORE PROCEDURE TO ACCEPT ONE PARAMETER

CREATE PROCEDURE JIT\_Employee @ID INT

AS

SELECT \* FROM Employee

WHERE ID = @ID

GO

--Each time you will want to the info of a particular ID, you will not need to write same code over and over, but simply change the ID

EXEC JIT\_Employee @ID =2

----CREATE STORE PROCEDURE TO ACCEPT TWO PARAMETERS

CREATE PROC Employee\_Info

@Name Varchar(20),

@City Varchar(20)

AS

SELECT \* FROM Employee

WHERE Name = @Name

AND City =@City

--TO FETCH INFO, YOU MUST INCLUDE BOTH PARAMETERS

Exec Employee\_Info @Name = 'Milda', @City = 'Austin'

--ALTER STORE PROCEDURE

--Maybe you want to change the sp to give you s

ALTER PROC Employee\_Info

@Name Varchar(20),

@Salary INT

AS

SELECT \* FROM Employee

WHERE Name = @Name

AND Salary =@Salary

Exec Employee\_Info @Name = 'Ngoh', @Salary = '7000'

ALTER PROCEDURE JIT\_Employee

AS

SELECT ID, NAME,GENDER FROM Employee

GO

EXEC JIT\_Employee

**FUNCTIONS**

***-Question what is the difference between a function and a store procedure?***

-sp is what applications use. With sp you build them so you can call them over and over again. Functions are used create an entry for values to be passed inside. Functions work with sp

when we were creating sp, we passed some parameters to return a result set, those parameters are functions.

-Functions can be called from sp but stored procedure can’t be called from functions.

-The function must return a value but in Stored Procedure it is optional

-In SQL server, a Function is a programming language that enables you to pass a value in order to get a set of result. We have system defined systems and user defined functions.

-E.g., of aggregate functions which are inbuilt; If you want to get the Min value in SQL server you can use the MIN function, if you want to get the sum of something you need the COUNT function., SELECT getdate()

-Functions are like store procedures but with sp you need functions in order to accomplish them.

- Create function StudentFirstName --Here we want to be able to rtrieve just firstname on the student table

(

@FirstName Varchar(20)

)

Returns Varchar(20)

AS

Begin Return (Select @FirstName)

SELECT \* FROM [dbo].[Student]

--What if we noticed instead of putting an ID of 6 for Marvin Daniel since its duplicate of 1?

Update Student

set ID = 6 WHERE FirstName='Marvin'

SELECT \* FROM [dbo].[Student]

Create Function StudentFirstName--Here we want to be able to retrieve just firstname on the student table

(

@FirstName Varchar(20)

)

Returns Varchar (20)

AS

Begin Return (Select @FirstName)

END

--CALLING THE FUNCTION

SELECT[dbo].[StudentFirstName] (FirstName) AS [Student Name] From Student

--COMPUTED COLUMNS-Here we want to declare two parameters(firstName and LastName)

Create Function StudentInfo

(

@FirstName Varchar(20),

@LastName varchar(30)

)

Returns Varchar(70)

AS

Begin Return (Select @FirstName)+' '+ @LastName

END

--CALLING THE FUNCTIONSELECT [dbo].[StudentInfo](FirstName, LastName) AS [FullName] From Student

--TABLE FUCNTIONS- This returns a table variable as a result of actions performed by the fucntion

Create Function fnStudent\_Table()

Returns Table

AS

Return (Select \* FROM Student)

--CALLING THE FUNCTION

SELECT \* FROM [dbo].[fnStudent\_Table]()

--BASIC INBUILT FUNCTIONS

--GETDATE-Get the date of the current day

SELECT GetDate()

--LEN-Number of characters

SELECT LEN ('This Kind chop get some fine taste boy') AS Comment

--LEFT- Extracts characters specified starting from the Left

SELECT LEFT ('Cameroonian Youths are brilliant',18) AS Opinion

-- --RIGHT- Extracts characters specified starting from the RIGHT

SELECT RIGHT ('Cameroonian Youths are brilliant',20) AS Opinion

--LOWER-Converts values that are upper case to lower

SELECT LOWER ('CAMEROoniAN YOUths') AS [Lower case]

--UPPER-Converts values that are upper case to UPPER

SELECT UPPER ('CameEROoniAN youths') AS [Upper case]

--TRIM-Removes leading spaces from records

SELECT TRIM (' Cameroonian Youths are brilliant' ) AS Final

--LEFT TRIM-Removes spaces from the leading spaces Left

SELECT TRIM (' Cameroonian Youths are brilliant') AS Final

--RIGHT TRIM-Removes spaces from the leading spaces RIGHT

SELECT TRIM ('Cameroonian Youths are brilliant ') AS Final

--REPLICATE-removes duplicate records

SELECT REPLICATE ( 'Goodness',4) AS Record

--REPLACE-Replaces values

SELECT REPLACE ('Mac-Donald', 'D', 'L')