Section 6: Jumping Back to Inserting Data (section 4)

- *Updating Data*
- *Deleting Data*
- *Truncate Tables*
- *Alter Table*
- *Select Into*
    - *With conditions*
- *Temporary Tables*

Section 6: Updating Data

Note be careful when updating data. Overwriting data cannot be reversed.

```
Update TableName
Set Column1 = value
Where condition

-> Update TestTable
Set PatientState = 'CA'
Where PatientID = '12354'
```

-> Example Statement

Section 6: Updating (Swapping) Values

Note be careful when updating data. Overwriting data cannot be reversed.

```
Update TableName
Set Column1 = Column2,
Column2 = Column1

-> Update TestTable
Set PatientState = Gender,
Gender = PatientState
```

-> Example Statement

Section 6: Deleting Data

Note be careful when deleting data. Once data is deleted it cannot be reversed.

```
Delete From TableName
Where condition

-> Delete From TestTable
Where PatientID = '12354'
```

-> Example Statement

Section 6: Truncating Tables

Truncating Tables is like deleting data and dropping a table. However, you delete all the data without dropping the table.

```
   Truncate Table TableName

-> Truncate Table TestTable
```

-> Example Statement

Section 6: Alter Table

Three ways you can alter a table

- Alter a column
- Add a column
- Drop a column

```
Alter Table TableName
Alter Column ColumnName datatype

Alter Table TableName
Add ColumnName datatype

Alter Table TableName
Drop Column ColumnName
```

-> Example Statement

Section 6: Alter Table Examples

```
-> Alter Table TestTable
   Alter Column Visits float

-> Alter Table TestTable
   Add PatientAge int
```

```
-> Update TestTable
   Set PatientAge = '25’
   Where PatientID = '12345'

-> Alter Table TestTable
   Drop Column PatientAge
```

-> Example Statement

Section 6: Select Into

```
-> Select
   Into
   From
   Where
   Group by
   Having
   Order by
```

```
-> Select
    PatientID
   ,PatientState
   ,Gender
   ,Visits
   ,Charges
   INTO TestTable2
   From TestTable
   Where Charges > 10000
```

-> Example Statement

Section 6: Select Into

```
-> Select
   Into
   From
   Where
   Group by
   Having
   Order by
```

```
-> Create Database Backup_SQLCourse_DB

   Select *
   INTO Backup_SQLCourse_DB.dbo.TestTable
   From TestTable
```

-> Example Statement

Section 6: Select Into with conditions

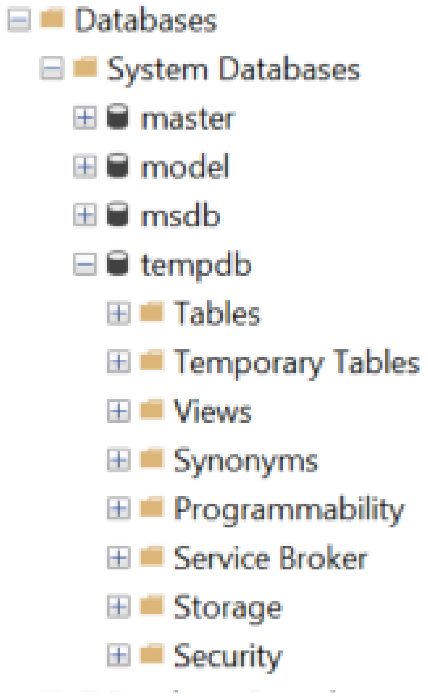
```
-> Select
   Into
   From
   Where
   Group by
   Having
   Order by
```

```
-> Select
Case When Gender = 'm' then 'Male'
 When Gender = 'f' then 'Female'
 Else null End as 'Gender'
,Sum(Visits) as Visits
,Sum(charges) as Charges
,Count(PatientID) as Number_of_Patients
Into TestTable3
From TestTable
Where PatientState in
('GA','FL','WY','UT','CA')
Group by Gender
Having Sum(Charges) > 10000
```

-> Example Statement

Section 6: Temporary Tables

Temporary Tables are useful when working with a very large datasets. You can create a subset of the tables that are needed to improve run times.



- Databases
  - System Databases
    - master
    - model
    - msdb
    - tempdb
      - Tables
      - Temporary Tables
      - Views
      - Synonyms
      - Programmability
      - Service Broker
      - Storage
      - Security

```
->Select
Case When Gender = 'm' then 'Male'
 When Gender = 'f' then 'Female'
 Else null End as 'Gender'
,Sum(Visits) as Visits
,Sum(charges) as Charges
,Count(PatientID) as Number_of_Patients
Into #TestTable
From TestTable
Where PatientState in
('GA','FL','WY','UT','CA')
Group by Gender
Having Sum(Charges) > 10000
```

-> Example Statement

Section 6: Self Evaluation

Step 1: Connect to SQLCourse_DB (or create this database if you haven’t already)
Step 2: Upate all of the values in the Gender Column to 'Male' and 'Female'
Step 3: Upate the Patients name to 'Bobby' where PatientID = '12348'
Step 4: Delete patients from 'FL'
Step 5: Alter the PatientState datatype to varchar(50)
Step 6: Put all the PatientNames and PatientIDs into another table called 'PatientTable'
Step 7: Add a few columns to 'PatientTable' - Weight, Height, Age
*You decide the datatype