

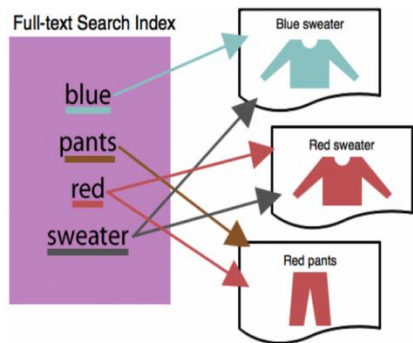


TABLE INDEXES, SEQUENCES, VIEW

Learning Goals

By the end of this lecture students should be able to:

- ✓ Create Indexes to improve query retrieval speed
- ✓ Automatically generate sequence numbers by using a sequence generator
- ✓ Create, maintain, and use View



	EmployeeID	PersonID	FirstName	LastName
1	1	1	Eralper	YILMAZ
2	2	3	Robert	Vieira
3	3	607	Guy	Gilbert
4	4	608	Kevin	Brown
5	5	609	Roberto	Tamburello
6	6	610	Rob	Walters
7	7	611	Thieny	D'Hers
8	8	612	David	Bradley
9	9	613	JoLynn	Dobney
10	10	614	Ruth	Ellerbrock
11	11	615	Gail	Erickson
12	12	616	Barry	Johnson
13	13	617	Jossef	Goldberg
14	14	618	Terri	Duffy
15	15	619	Sidney	Higa

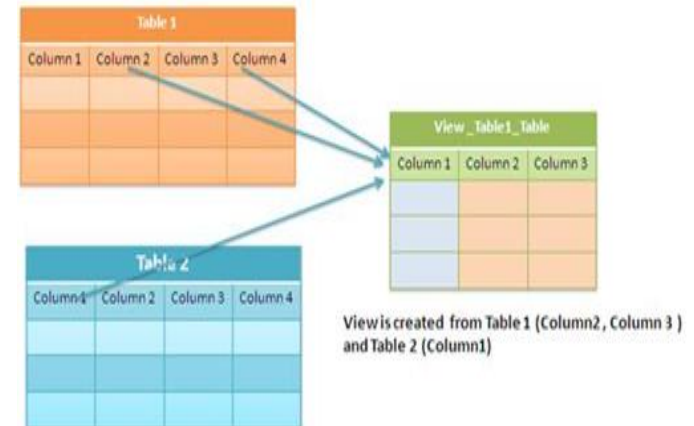


Table of contents



Table Indexes



Sequence



View



Demo



Quiz

Trainee's missions

To complete this course and achieve goals, trainees must:

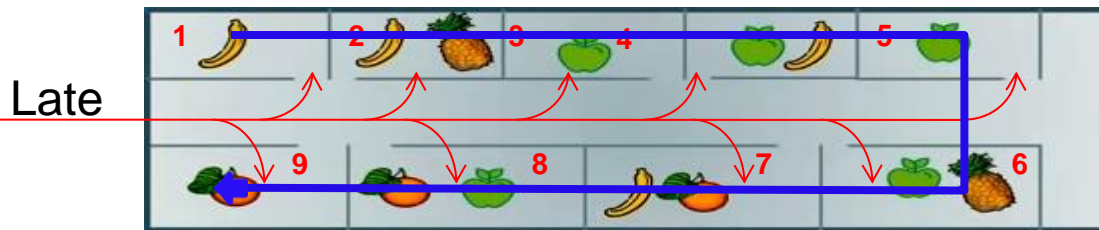
- ✓ **Read Lecture**
- ✓ **Do Exercises**
- ✓ **Take quizz**
- ✓ **Complete final exam**

TABLE INDEXES

Why use indexes?

❁ An **index** in database is similar to an index in a book

❁ **Indexes** in database help speed up search queries. Allow find data in a table without scanning the entire table.



 I want bananas ...

Result:



Quick

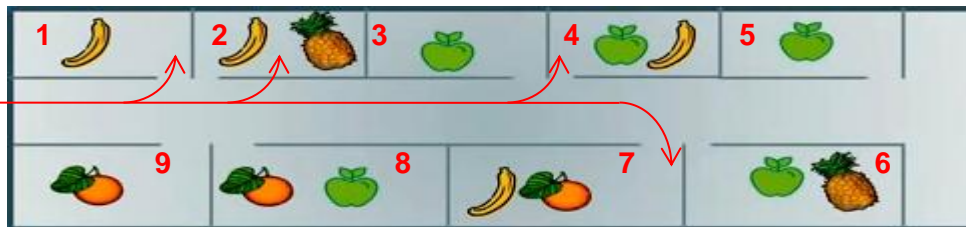


Table Indexes (1/3)

```
CREATE TABLE dbo.PhoneBook
```

```
(
```

```
    LastName          varchar(50) NOT NULL
```

```
    ,    FirstName     varchar(50) NOT NULL
```

```
    ,    PhoneNumber   varchar(50) NOT NULL
```

```
);
```

```
SELECT PhoneNumber
```

```
FROM   dbo.PhoneBook
```

```
WHERE  LastName = 'Logan' AND FirstName = 'Todd';
```

Alexander, Mary
344-555-0133

Kurtz, Jeffrey
452-555-0179

Vessa, Robert
560-555-0171

Thames, Judy
799-555-0118

Martinez, Frank
171-555-0147

Haines, Betty
867-555-0114

Burnett, Linda
121-555-0121

Harris, Keith
170-555-0127

Kitt, Sandra
303-555-0117

Brewer, Alan
494-555-0134

Campbell, Frank
491-555-0132

Logan, Todd
783-555-0110

...

Clayton, Jane
206-555-0195

Johnson, Brian
320-555-0134

Liu, David
440-555-0132

Diaz, Brenda
147-555-0192

Table Indexes (2/3)

```
CREATE TABLE dbo.PhoneBook
```

```
(
```

```
    LastName          varchar(50) NOT NULL
```

```
    ,    FirstName     varchar(50) NOT NULL
```

```
    ,    PhoneNumber   varchar(50) NOT NULL
```

```
);
```

```
SELECT PhoneNumber
```

```
FROM   dbo.PhoneBook
```

```
WHERE  LastName = 'Logan' AND FirstName = 'Todd';
```

Results:

581-555-0172

Alexander, Mary
344-555-0133

Kurtz, Jeffrey
452-555-0179

Vessa, Robert
560-555-0171

Thames, Judy
799-555-0118

Martinez, Frank
171-555-0147

Haines, Betty
867-555-0114

Burnett, Linda
121-555-0121

Harris, Keith
170-555-0127

Kitt, Sandra
303-555-0117

Brewer, Alan
494-555-0134

Campbell, Frank
491-555-0132

Logan, Todd
783-555-0110

...

Clayton, Jane
206-555-0195

Johnson, Brian
320-555-0134

Liu, David
440-555-0132

Diaz, Brenda
147-555-0192

There are 2 types of major Indexes:

✓ Clustered

- Data is stored in the order on the clustered index
- Only 1 clustered index per table
- Usually the Primary Key
- Sort and store the data rows in the table based on their key value

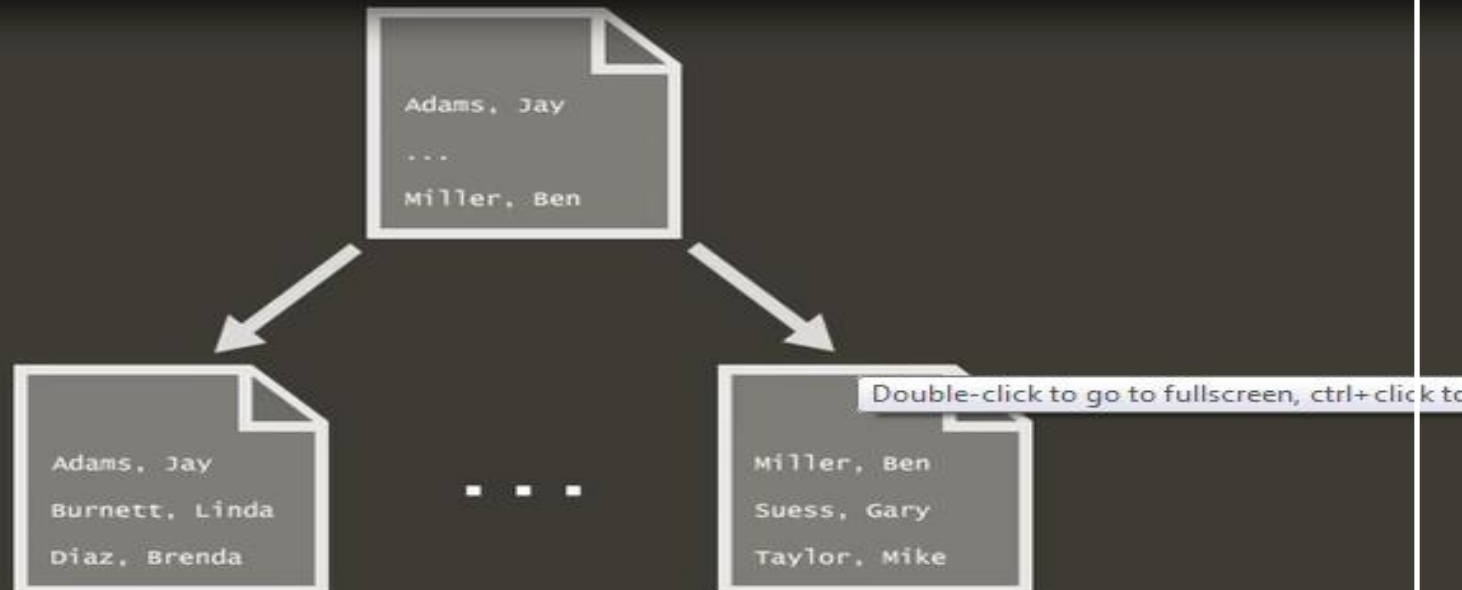
✓ Non-clustered

- Data is not stored in the order on the non clustered index
- Have a structure completely separate from the data rows

Clustered Index

```
CREATE CLUSTERED INDEX IX_PhoneBook_CI  
ON dbo.PhoneBook (LastName, FirstName)
```

Index



Data

```
Adams, Jay  
158-555-0142  
Alexander, Mary  
344-555-0133  
Benson, Edna  
789-555-0189  
Brewer, Alan  
494-555-0134
```

```
Burnett, Linda  
121-555-0121  
Campbell, Frank  
491-555-0132  
Clayton, Jane  
206-555-0195  
Cooper, Scott  
733-555-0182
```

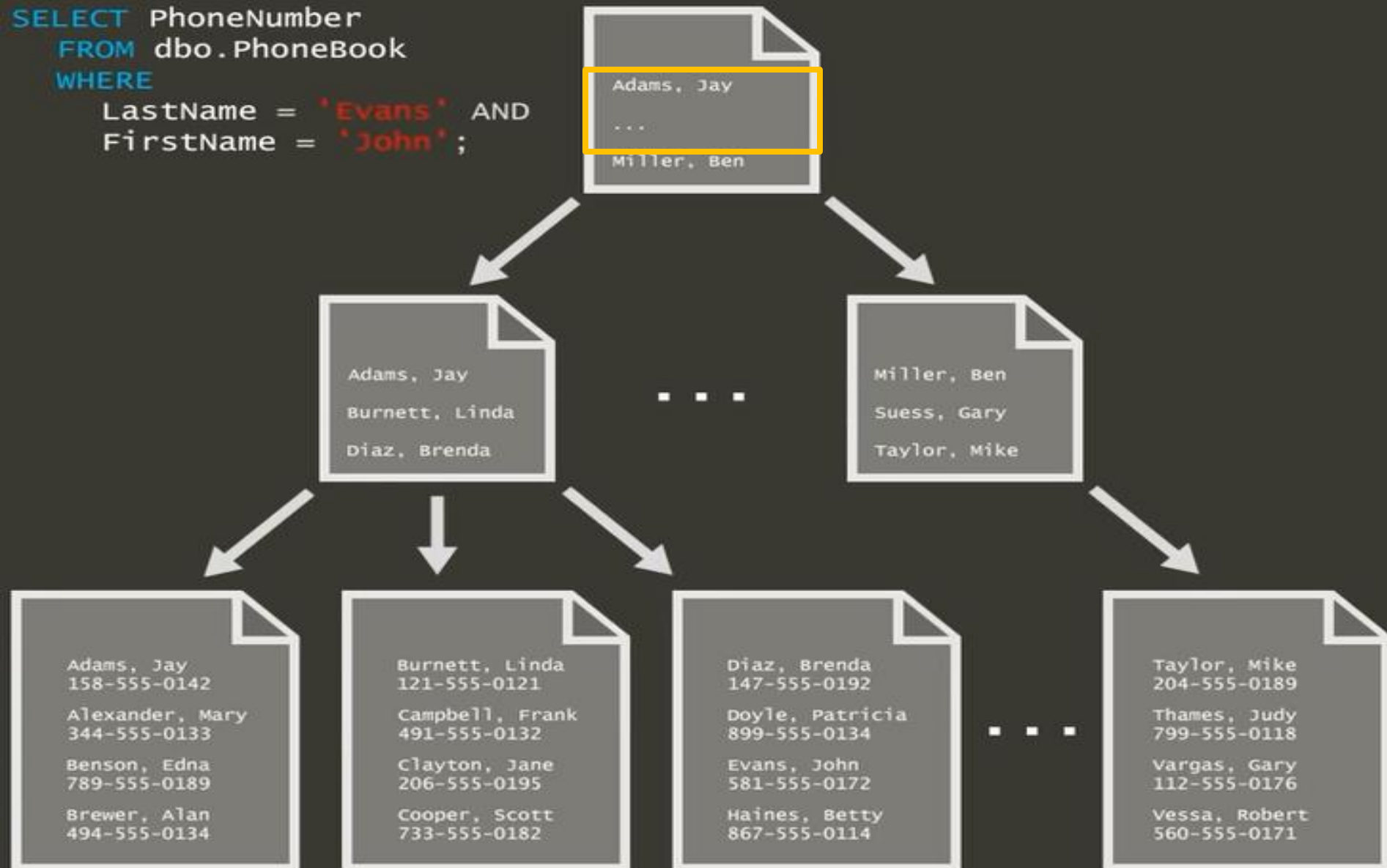
```
Diaz, Brenda  
147-555-0192  
Doyle, Patricia  
899-555-0134  
Evans, John  
581-555-0172  
Haines, Betty  
867-555-0114
```

...

```
Taylor, Mike  
204-555-0189  
Thames, Judy  
799-555-0118  
Vargas, Gary  
112-555-0176  
Vessa, Robert  
560-555-0171
```

Clustered Index

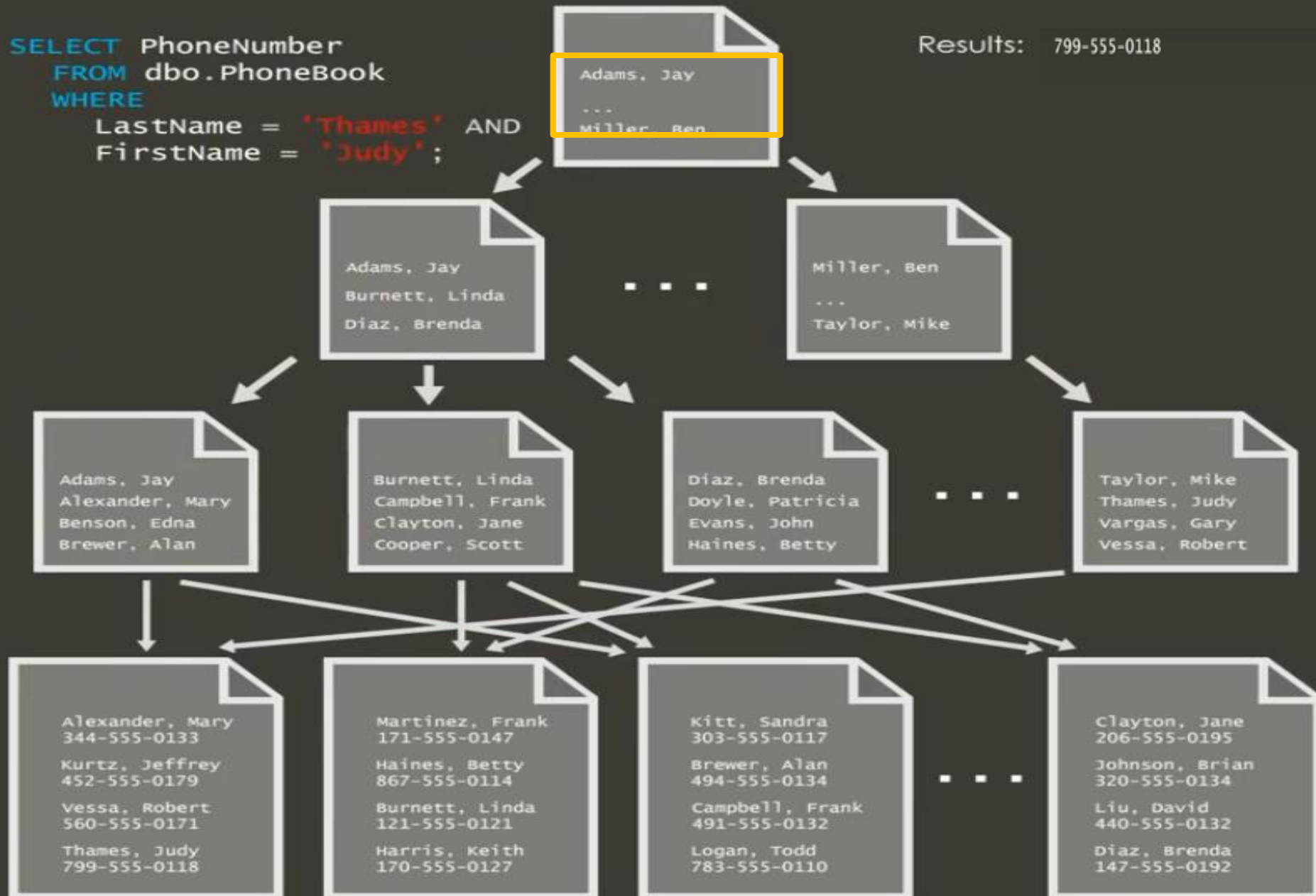
```
SELECT PhoneNumber  
FROM dbo.PhoneBook  
WHERE  
    LastName = 'Evans' AND  
    FirstName = 'John';
```



Non - Clustered Index

```
SELECT PhoneNumber
FROM dbo.PhoneBook
WHERE
    LastName = 'Thames' AND
    FirstName = 'Judy';
```

Results: 799-555-0118



Creating an Index

```
CREATE INDEX index_name  
ON table_name (column1_name, column2_name, ...)
```

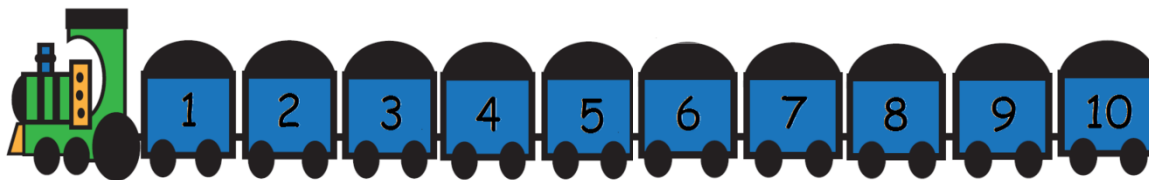
Deleting an Index

```
DROP INDEX table_name.index_name
```

SEQUENCES

What Is a Sequence?

- ✿ This creates an auto increment for a column
- ✿ If a table has a column with sequence or auto increment, the user do not need insert data explicitly for the column
- ✿ The sequence of numeric values is generated in an ascending or descending order at a defined interval and can be configured to restart (cycle) when exhausted. Sequences, unlike identity columns, are not associated with specific tables. Applications refer to a sequence object to retrieve its next value.
- ✿ **Note:** Sequences are new in **SQL Server 2012** and not available in earlier versions



Create Sequence

```
CREATE SEQUENCE [schema_name . ] sequence_name
[AS[built_in_integer_type|user-defined_integer_type]]
[ START WITH <constant> ]
[ INCREMENT BY <constant> ]
[ { MINVALUE [ <constant> ] } | { NO MINVALUE } ]
[ { MAXVALUE [ <constant> ] } | { NO MAXVALUE } ]
[ CYCLE | { NO CYCLE } ]
[ { CACHE [ <constant> ] } | { NO CACHE } ] [ ; ]
```


Example

 Creating a sequence that increases by 1

```
CREATE SEQUENCE Test.CountBy1
  START WITH 1
  INCREMENT BY 1 ;
GO
```

 Creating a sequence that decreases by 1

```
CREATE SEQUENCE Test.CountByNeg1
  START WITH 0
  INCREMENT BY -1 ;
GO
```

 Creating a sequence that increases by 5

```
CREATE SEQUENCE Test.CountBy1
  START WITH 5
  INCREMENT BY 5 ;
GO
```

 Creating a sequence that starts with a designated number

```
CREATE SEQUENCE Test.ID_Seq
  START WITH 24329
  INCREMENT BY 1 ;
```

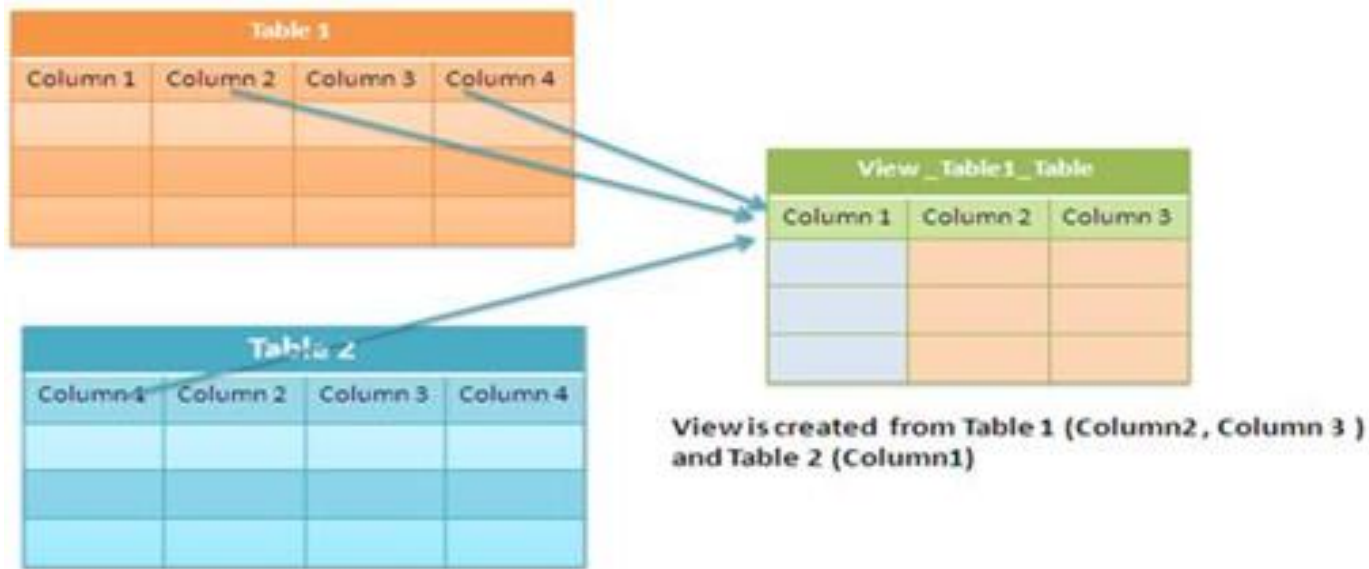
GO



VIEWS

What is a view?

- ❁ A **View** is a logical or virtual table. The fields in a view are fields from one or more real tables in the database.
- ❁ There are **two major reasons** you might want to use views:
 - ✓ Views allow you to limit the data users can access
 - ✓ Views reduce complexity for end users.



Creating a view

```
CREATE VIEW View_Name [list of column names]
AS
SELECT...
```

Example:

```
CREATE VIEW view_EmployeeByDpt
AS
SELECT ID, NAME, AGE, DEPT_NAME
FROM EMP, DEPARTMENT
WHERE EMP.DEPT_ID = DEPARTMENT.DEPT_ID
```

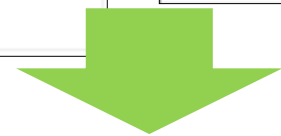
```
SELECT * FROM view_EmployeeByDpt
```

Table: EMP

ID	NAME	AGE	DEP_ID
1	John	25	3
2	Mike	30	2
3	Parm	25	1
4	Todd	23	4
5	Sara	35	1
6	Ben	40	3

Table: DEPARTMENT

DEPT_ID	DEPT_NAME
1	IT
2	Payroll
3	HR
4	Admin



view_EmployeeByDpt


ID	NAME	AGE	DEPT_NAME
1	John	25	HR
2	Mike	30	Payroll
3	Parm	25	IT
4	Todd	23	Admin
5	Sara	35	IT
6	Ben	40	HR

Deleting a view

```
DROP VIEW View_Name
```

Example:

```
DROP VIEW view_EmployeeByDpt
```



ID	NAME	ASSIGNMENT_ID	DEPT_ID	DEPT_NAME
1	John	25	1	HR
2	Nick	NULL	2	Payroll
3	Pam	25	3	IT
4	Todd	4	4	Admin
5	Sara	1	1	IT
6	Ben	3	3	HR

view_EmployeeByDpt



View Demo

 Demo

Quiz!

*Now let's check how you understand
the lecture!*

*There are 6 questions below.
Click **NEXT** button to start!*

Now let's check how you
understand the lecture!

Quiz!

*There are 6 questions below.
Click **NEXT** button to start!*

Summary



Table Indexes

- Why use indexes?
- Create, maintain and use index



Sequences

- Why sequence is used in SQL?
- Automatically generate sequence numbers



View

- Create, maintain and use view



Demo

- View



Quiz



THANK YOU

You have completed "**Lecture 5**" course.

Click EXIT button to exit course and discover
the next Lecture "**Lecture 6**".

EXIT