|  |
| --- |
| **National University of Computer and Emerging Sciences** |
| Lab Manual 9  “**Indexing**” |
|  |
| Data Warehousing and Data Mining |
|  |
|  |
|  |

|  |  |
| --- | --- |
| Course Instructor | Sir Ishaq Raza |
| Lab Instructor(s) | Farhan Azhar  Nouman Ali |
| Section | CS |
| Semester | Fall 2019 |

|  |
| --- |
|  |

Department of Computer Science

FAST-NU, Lahore, Pakistan

**Contents**

*Total (170 minutes)*

* + **Introduction to Index utilization and Execution Plans** *(15 minutes)*
  + **Creating Environment with Scripts and Data loading** *(10 minutes)*
  + **Creating Indexes on all tables** (15 minutes)
  + **Performance Queries for Clustered Index** (20 minutes)
  + **Performance Queries for Non Clustered Index** (20 minutes)
  + **Performance Queries with Clustered Index & Non Clustered Index** (20 minutes)
  + **Practice Exercise** *(70 minutes)*

The purpose of this part of lab is to give you a practical exposure on theoretical concepts like indexing and show you how different indexing techniques are helpful in different scenarios by drawing a comparison between them.

**Database Script:** The SQL script to create tables and indexes is available with name **Indexing.sql**

**Steps for Creating an Environment for Index Utilisation:**

1. **Create table** sx080229sggrp from the script (Provided in Indexing.sql file)
2. **Load** the file sx080229sggrp.txt in the table sx080229sggrp.
3. Create a replica table [sx080229sggrp\_CI] with the same script as given above. (**CI** refers to Clustered Index)
4. Create a replica table [sx080229sggrp\_NCI] with the same script as given above. (**NCI** for non-Clustered Index)
5. **Create Clustered Index** on [sx080229sggrp\_CI] table
6. **Create Non Clustered Index** on [sx080229sggrp\_NCI] table
7. **Load** the data from sx080229sggrp table to these tables

**Clustered Index:**

To create clusterd index, use below mentioned query. Also provided in **Indexing.sql** script file.

ALTER TABLE [dbo].[sx080229sggrp\_CI] ADD CONSTRAINT [PK\_sx080229sggrp\_CI] PRIMARY KEY CLUSTERED

(

[#Date] ASC,

[Instrument Code] ASC

) WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, SORT\_IN\_TEMPDB = OFF, IGNORE\_DUP\_KEY = OFF, ONLINE = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

**Non Clustered Index:**

To create non-clusterd index, use below mentioned query. Also provided in **Indexing.sql** script file.

ALTER TABLE sx080229sggrp\_NCI ADD CONSTRAINT PK\_DateInst\_NCI PRIMARY KEY NONCLUSTERED

(

#Date ASC,

[Instrument code] ASC

)

**Populating tables with data :**

**1. Inserting more records (from existing) for making a huge table**

**INSERT INTO sx080229sggrp**

( #Date, [Instrument Code], [Description], [SEDOL code], Country\_ISO, [Industry Level 1],

[Industry Level 2], Currency, Shares )

SELECT REPLACE(CONVERT(VARCHAR(10), #Date) ,8,7) NewDate ,

REPLACE(CONVERT(VARCHAR(10), [Instrument Code] ) ,'S','M') [New Instrument Code]

, 'Old ' + [Description] Description , [SEDOL code], Country\_ISO, [Industry Level 1]

,[Industry Level 2], Currency, Shares -- #Date, [Instrument Code]

FROM sx080229sggrp

**INSERT INTO sx080229sggrp**

( #Date, [Instrument Code], [Description], [SEDOL code], Country\_ISO, [Industry Level 1],

[Industry Level 2], Currency, Shares )

SELECT REPLACE(CONVERT(VARCHAR(10), #Date) ,8,9) NewDate ,

REPLACE(CONVERT(VARCHAR(10), [Instrument Code] ) ,'M','N') [New Instrument Code]

, 'New ' + [Description] Description , [SEDOL code], Country\_ISO, [Industry Level 1]

,[Industry Level 2], Currency, Shares -- #Date, [Instrument Code]

FROM sx080229sggrp

**INSERT INTO sx080229sggrp**

( #Date, [Instrument Code], [Description], [SEDOL code], Country\_ISO, [Industry Level 1],

[Industry Level 2], Currency, Shares )

SELECT REPLACE(CONVERT(VARCHAR(10), #Date) ,8,5) NewDate ,

REPLACE(CONVERT(VARCHAR(10), [Instrument Code] ) ,'S','R') [New Instrument Code]

, 'Int ' + [Description] Description , [SEDOL code], Country\_ISO, [Industry Level 1]

,[Industry Level 2], Currency, Shares -- #Date, [Instrument Code]

FROM sx080229sggrp

WHERE #Date LIKE '%2012%'

**2. Deleting duplicate records**

**DELETE A**

-- SELECT A.[#Date], A.[Instrument Code], A.Description, B.[#Date], B.[Instrument Code]

FROM sx080229sggrp A

INNER JOIN

( SELECT [#Date], [Instrument Code]

FROM sx080229sggrp

group BY [#Date], [Instrument Code]

HAVING COUNT(#Date) > 1

)B ON A.[#Date]= B.[#Date] AND A.[Instrument Code]= B.[Instrument Code]

**3. Poluating same data from sx080229sggrp to sx080229sggrp\_CI and sx080229sggrp\_NCI**

**INSERT INTO sx080229sggrp\_CI**

( #Date, [Instrument Code], [Description], [SEDOL code], Country\_ISO, [Industry Level 1],

[Industry Level 2], Currency, Shares )

SELECT #Date, [Instrument Code], [Description], [SEDOL code], Country\_ISO, [Industry Level 1],

[Industry Level 2], Currency, Shares

FROM sx080229sggrp

**INSERT INTO sx080229sggrp\_NCI**

( #Date, [Instrument Code], [Description], [SEDOL code], Country\_ISO, [Industry Level 1],

[Industry Level 2], Currency, Shares )

SELECT #Date, [Instrument Code], [Description], [SEDOL code], Country\_ISO, [Industry Level 1],

[Industry Level 2], Currency, Shares

FROM sx080229sggrp

**Queries:**

**Table without Index**

**Query 1:** SELECT \* ***-- Table Scan***

FROM sx080229sggrp

**Query 2:** SELECT \* ***-- Table Scan***

FROM sx080229sggrp

WHERE [#Date]=20120201 AND [Instrument Code] = 'S80323'

**Query 3:** SELECT \* ***-- Table Scan***

FROM sx080229sggrp

WHERE [#Date] BETWEEN 20120201 AND 20120206

AND [Instrument Code] BETWEEN 'S80323' AND 'S90907'

**Clustered Index Table :**

**Query 1:** SELECT \* ***-- Clustered Index Scan***

FROM sx080229sggrp\_CI

**Query 2:** SELECT #Date, [Instrument Code] ***-- Clustered Index Seek***

FROM sx080229sggrp\_CI

WHERE [#Date]=20120201 AND [Instrument Code] = 'S80323'

SELECT #Date, [Instrument Code] ***-- Clustered Index Seek***

FROM sx080229sggrp\_CI

WHERE [#Date] BETWEEN 20120201 AND 20120206

AND [Instrument Code] BETWEEN 'S80323' AND 'S90907'

**Query 3:** SELECT \* ***-- Clustered Index Seek***

FROM sx080229sggrp\_CI

WHERE [#Date]=20120201 AND [Instrument Code] = 'S80323'

**Query 4:** SELECT \* ***-- Clustered Index Seek***

FROM sx080229sggrp\_CI

WHERE [#Date] BETWEEN 20120201 AND 20120206

AND [Instrument Code] BETWEEN 'S80323' AND 'S90907'

**Non Clustered Index Table :**

**Query 2:** SELECT #Date, [Instrument Code] ***-- Index Seek***

FROM sx080229sggrp\_nCI

WHERE [#Date]=20120201 AND [Instrument Code] = 'S80323'

**Query 3:** SELECT #Date, [Instrument Code] ***-- Index Seek***

FROM sx080229sggrp\_NCI

WHERE [#Date] BETWEEN 20120201 AND 20120206

AND [Instrument Code] BETWEEN 'S80323' AND 'S90907'

**Query 4:** SELECT \* ***-- Index Seek + RIDLookup (Nested Loops Inner join)***

FROM sx080229sggrp\_nCI

WHERE [#Date]=20120201 AND [Instrument Code] = 'S80323'

**Query 5:** SELECT \* ***--Table Scan***

FROM sx080229sggrp\_NCI

WHERE [#Date] BETWEEN 20120201 AND 20120206

AND [Instrument Code] BETWEEN 'S80323' AND 'S90907'

**Task:** Please fill following table with time and I/O cost taken by queries

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Simple | Clustered Index | Non-Clustered Index | Reason (Why one takes less time) |
| Query 1 |  |  |  |  |
| Query 2 |  |  |  |  |
| Query 3 |  |  |  |  |
| Query 4 |  |  |  |  |
| Query 5 |  |  |  |  |

**Performance Comparison w.r.t Time:**

**Join based query for verification of time elapsed by query execution**

**1. Single Join (**Time almost same**)**

SELECT \* **Time Taken =**

FROM sx080229sggrp A

INNER JOIN sx080229sggrp B

ON A.#Date= B.#Date AND A.[Instrument Code]= B.[Instrument Code]

SELECT \* **Time Taken =**

FROM sx080229sggrp\_CI A

INNER JOIN sx080229sggrp\_CI B

ON A.#Date= B.#Date AND A.[Instrument Code]= B.[Instrument Code]

**Double Join (**Time taken by queries varies by some frequency**)**

SELECT \* **Time Taken =**

FROM sx080229sggrp A

INNER JOIN sx080229sggrp B

ON A.#Date= B.#Date AND A.[Instrument Code]= B.[Instrument Code]

INNER JOIN sx080229sggrp C

ON A.#Date= C.#Date AND A.[Instrument Code]= C.[Instrument Code]

SELECT \* **Time Taken =**

FROM sx080229sggrp\_CI A

INNER JOIN sx080229sggrp\_CI B

ON A.#Date= B.#Date AND A.[Instrument Code]= B.[Instrument Code]

INNER JOIN sx080229sggrp\_CI C

ON A.#Date= C.#Date AND A.[Instrument Code]= C.[Instrument Code]

**1. Tripple Join (**Time taken by queries varies by almost double frequency**)**

SELECT \* **Time Taken =**

FROM sx080229sggrp A

INNER JOIN sx080229sggrp B

ON A.#Date= B.#Date AND A.[Instrument Code]= B.[Instrument Code]

INNER JOIN sx080229sggrp C

ON A.#Date= C.#Date AND A.[Instrument Code]= C.[Instrument Code]

INNER JOIN sx080229sggrp D

ON A.#Date= D.#Date AND A.[Instrument Code]= D.[Instrument Code]

SELECT \*  **Time Taken =**

FROM sx080229sggrp\_CI A

INNER JOIN sx080229sggrp\_CI B

ON A.#Date= B.#Date AND A.[Instrument Code]= B.[Instrument Code]

INNER JOIN sx080229sggrp\_CI C

ON A.#Date= C.#Date AND A.[Instrument Code]= C.[Instrument Code]

INNER JOIN sx080229sggrp\_CI D

ON A.#Date= D.#Date AND A.[Instrument Code]= D.[Instrument Code]

**Task:** Please fill following table with time and I/O cost taken by queries

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Simple Query | Clustered Index | Non-Clustered Index | Reason (Why one takes less time) |
| Single Join |  |  |  |  |
| Double Join |  |  |  |  |
| Triple Join |  |  |  |  |