National University of Computer and Emerging Sciences

Lab Manual 3

"Data Retrieval"

Database Systems

Spring 2021

Department of Computer Science
FAST-NU, Lahore, Pakistan

Table of Contents

1. Objective

The purpose of this manual is to get stared with data retrieval queries, starting from simple Select-From-Where, Order by clause, arithmetic operations and finally covering set operations.

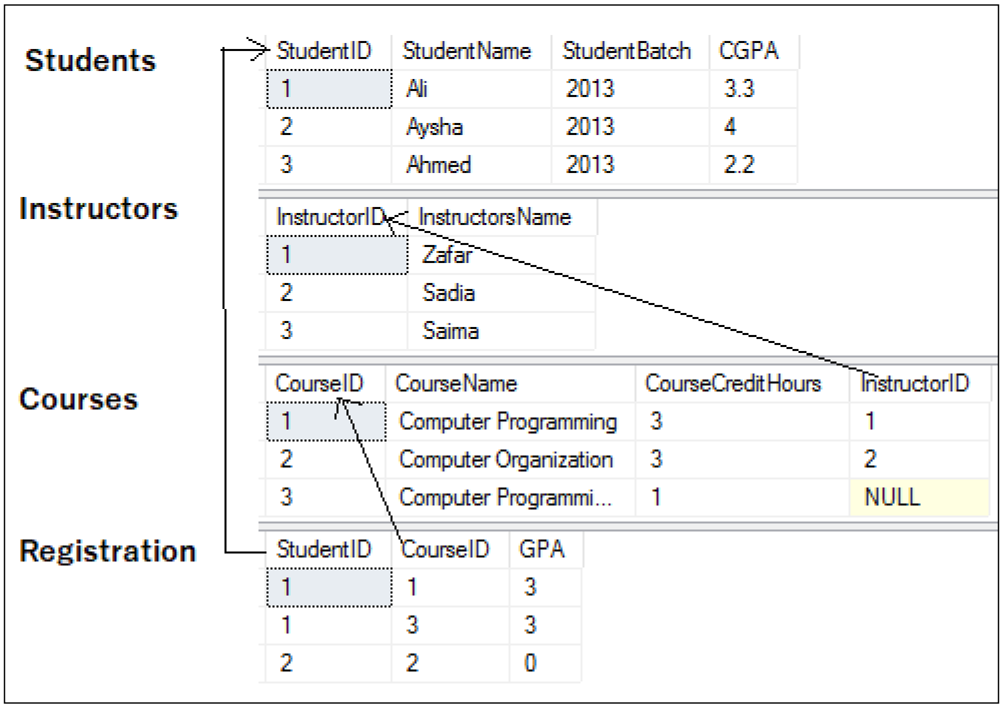
2. Pre-requisites

- Lab 2 manual, on how to get started with MS-SQL server
- How Select-From-Where clause works
- How Order by clause works
- How arithmetic operations like +, -, *, /, % works
- How set Operations like Union, Intersect, Except work

3.

SELECT-FROM-WHERE

Select from where is equivalent to projection and selection in Relational Algebra, it will give output in form of a table.
The most basic select statement includes Select and from clause, and it will retrieve all columns and rows from the table.
We will use the following schema and database for the examples. Find the queries for this database in InLab3Practice.sql and start practicing.



Students

StudentID	StudentName	StudentBatch	CGPA
1	Ali	2013	3.3
2	Aysha	2013	4
3	Ahmed	2013	2.2

Instructors

InstructorID	InstructorsName
1	Zafar
2	Sadia
3	Saima

Courses

CourseID	CourseName	CourseCreditHours	InstructorID
1	Computer Programming	3	1
2	Computer Organization	3	2
3	Computer Programmi...	1	NULL

Registration

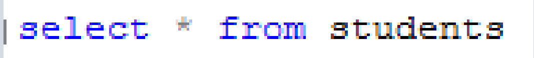
StudentID	CourseID	GPA
1	1	3
1	3	3
2	2	0

Most Basic Select:

Retrieve data from table. Operator * after select means that all columns will be retrieved.

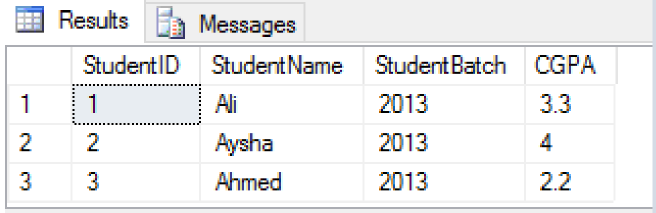
Syntax:
SELECT *
FROM <tableName>

Try this



```
select * from students
```

Results



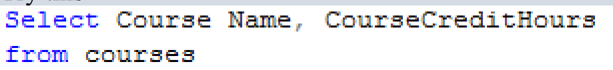
	StudentID	StudentName	StudentBatch	CGPA
1	1	Ali	2013	3.3
2	2	Aysha	2013	4
3	3	Ahmed	2013	2.2

Retrieving Certain Columns from Select

To retrieve only certain columns give a comma separated list of those columns after Select keyword

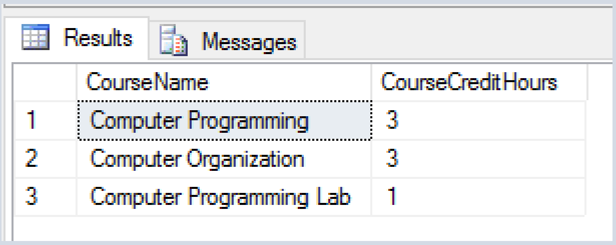
Syntax:
SELECT ColumnX, ColumnY, ColumnZ
FROM <tableName>

Try this



```
Select Course Name, CourseCreditHours
from courses
```

Results



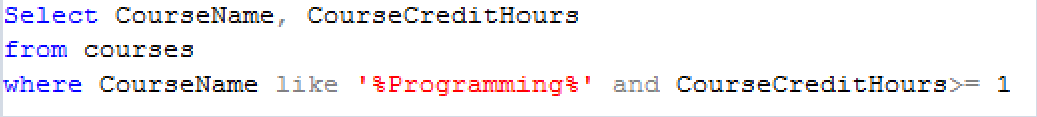
	CourseName	CourseCreditHours
1	Computer Programming	3
2	Computer Organization	3
3	Computer Programming Lab	1

Retrieving Certain Rows from SELECT - WHERE CLAUSE

Rows can be filtered in SQL using WHERE clause. Rows that fulfill where clause conditions will be projected in result. Where clause can put condition on original columns of tables mentioned in from clause. Also, observe the use of Like operator in where clause.

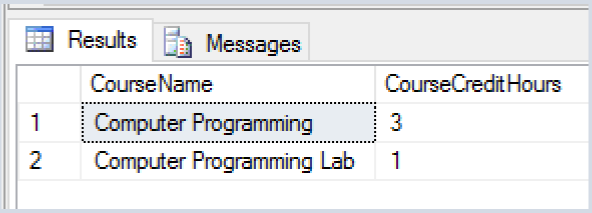
Syntax:
SELECT *
FROM <tableName>
where <conditions>

Try this



```
Select CourseName, CourseCreditHours
from courses
where CourseName like '%Programming%' and CourseCreditHours>= 1
```

Results



	CourseName	CourseCreditHours
1	Computer Programming	3
2	Computer Programming Lab	1

Like Operator Scenarios

WHERE CourseName LIKE 'C%'	Finds any values that start with "C"
WHERE CourseName LIKE '%C'	Finds any values that end with "C"
WHERE CourseName LIKE '%Co%'	Finds any values that have "Co" in any position
WHERE CourseName LIKE '_r%'	Finds any values that have "r" in the second position
WHERE CourseName LIKE 'C_%'	Finds any values that start with "C" and are at least 2 characters in length
WHERE CourseName LIKE 'C__%	Finds any values that start with "C" and are at least 3 characters in length
WHERE CourseName LIKE 'C%r'	Finds any values that start with "C" and ends with "r"

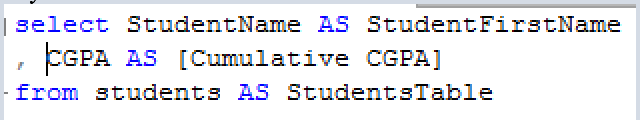
NOTE: % is referred to as **wildcard**.

Renaming Resulting Column

You can rename a column in result by using AS keyword also called Alias. The scope of this renaming is only to that select query, this is useful in joining where more than one table have same column names.

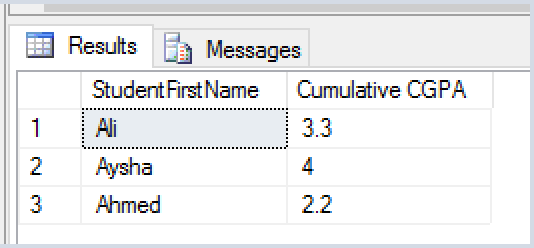
Syntax:
SELECT ColumnX as X , ColumnY as Y, ColumnZ
FROM <tableName> as Table1

Try this



```
select StudentName AS StudentFirstName
, CGPA AS [Cumulative CGPA]
from students AS StudentsTable
```

Results



	StudentFirstName	Cumulative CGPA
1	Ali	3.3
2	Aysha	4
3	Ahmed	2.2

SQL Server Built-in Functions

Sql Server has many built-in functions which can be used for different purposes.
For example:
1) GETDATE Returns the current database system date and time
2) CURRENT_TIMESTAMP Returns the current date and time
3) SUBSTRING Extracts some characters from a string

Syntax:
1) SELECT GETDATE();
3) SELECT CURRENT_TIMESTAMP;
2) SELECT SUBSTRING(columnName, startposition, substringlength) AS alias FROM <tableName>;

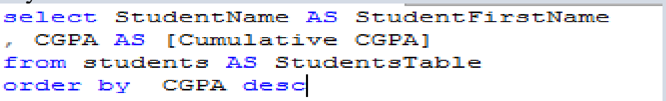
Try to explore as many string and data functions through this link:
https://www.w3schools.com/sql/sql_ref_sqlserver.asp

4. Order by Clause

Order by clause is used to arrange the rows in ascending or descending order of one or more columns

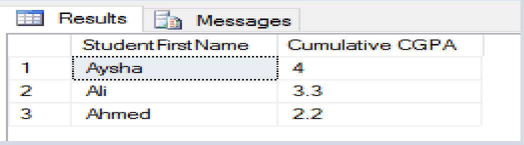
Syntax:
SELECT ColumnX as X, ColumnY as Y, ColumnZ
FROM <tableName> as Table1
ORDER BY ColumnX asc/desc, ColumnZ asc/desc

Try this



```
select StudentName AS StudentFirstName
, CGPA AS [Cumulative CGPA]
from students AS StudentsTable
order by  CGPA desc
```

Results



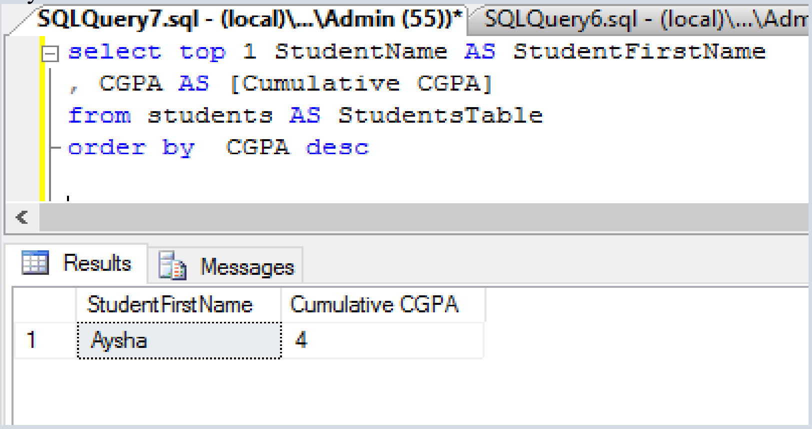
	StudentFirstName	Cumulative CGPA
1	Aysha	4
2	Ali	3.3
3	Ahmed	2.2

TOP Clause

Top n clause will give you first n rows from result instead of all the rows.

Syntax:
SELECT TOP <n> *
FROM <tableName>
WHERE <conditions>
ORDER BY <column Name> asc/desc

Try this



SQLQuery7.sql - (local)\...\Admin (55))*
SQLQuery6.sql - (local)\...\Adm
Results
Messages

```
select top 1 StudentName AS StudentFirstName
, CGPA AS [Cumulative CGPA]
from students AS StudentsTable
order by  CGPA desc
```

	StudentFirstName	Cumulative CGPA
1	Aysha	4

5. Arithmetic Operations

Sql arithmetic operators are:

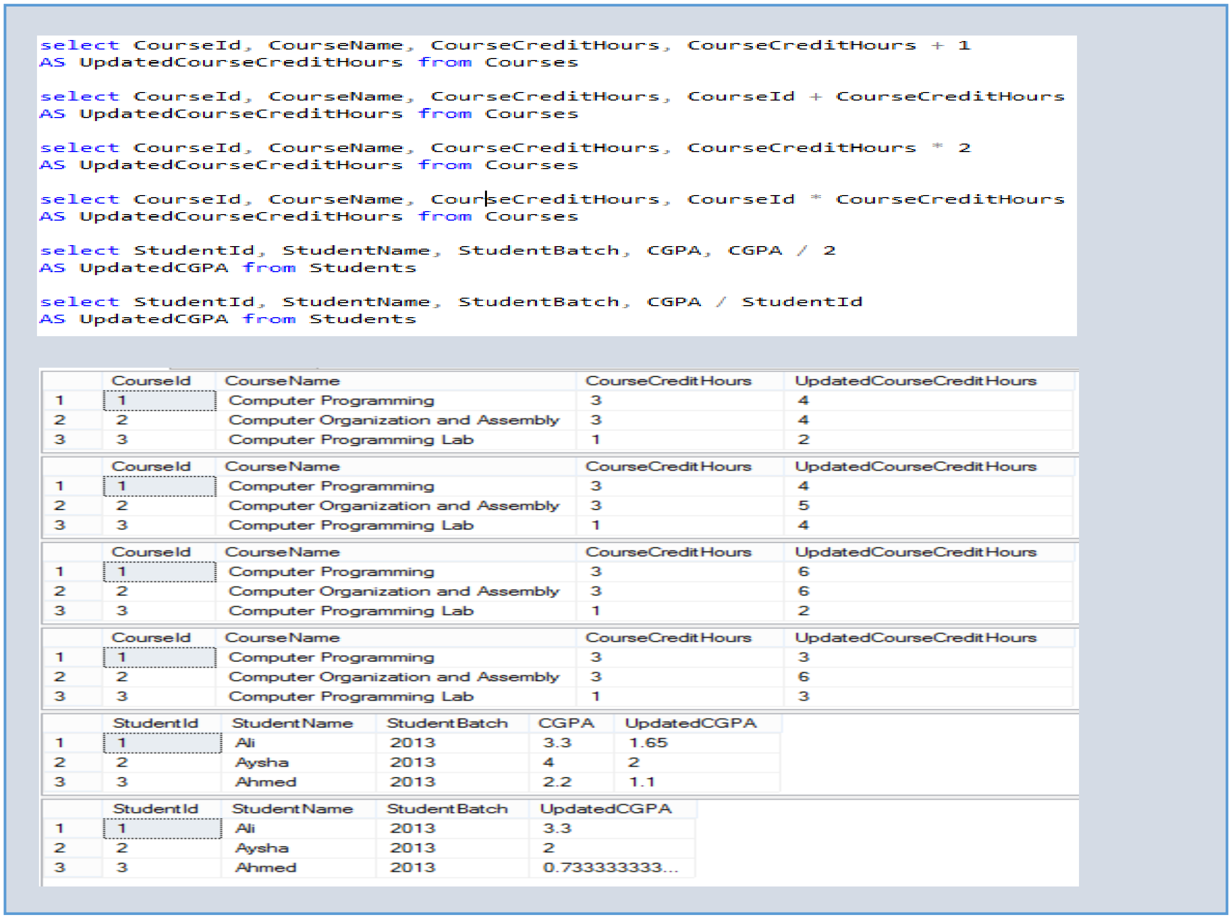
- + Addition
- - Subtraction
- / Division
- * Multiplication
- % Modulus

All operations can be performed on either single column or multiple columns

Syntax:
1. Apply operation on single columns
SELECT ColumnX, ColumnY + 100
FROM <tableName>

2. Apply operation on multiple columns
SELECT ColumnX, ColumnY + ColumnZ
FROM <tableName>

Replace + with other operators and try them out yourself.



```
select CourseId, CourseName, CourseCreditHours, CourseCreditHours + 1
AS UpdatedCourseCreditHours from Courses

select CourseId, CourseName, CourseCreditHours, CourseId + CourseCreditHours
AS UpdatedCourseCreditHours from Courses

select CourseId, CourseName, CourseCreditHours, CourseCreditHours * 2
AS UpdatedCourseCreditHours from Courses

select CourseId, CourseName, CourseCreditHours, CourseId * CourseCreditHours
AS UpdatedCourseCreditHours from Courses

select StudentId, StudentName, StudentBatch, CGPA, CGPA / 2
AS UpdatedCGPA from Students

select StudentId, StudentName, StudentBatch, CGPA / StudentId
AS UpdatedCGPA from Students
```

	CourseId	CourseName	CourseCreditHours	UpdatedCourseCreditHours
1	1	Computer Programming	3	4
2	2	Computer Organization and Assembly	3	4
3	3	Computer Programming Lab	1	2

	CourseId	CourseName	CourseCreditHours	UpdatedCourseCreditHours
1	1	Computer Programming	3	4
2	2	Computer Organization and Assembly	3	5
3	3	Computer Programming Lab	1	4

	CourseId	CourseName	CourseCreditHours	UpdatedCourseCreditHours
1	1	Computer Programming	3	6
2	2	Computer Organization and Assembly	3	6
3	3	Computer Programming Lab	1	2

	CourseId	CourseName	CourseCreditHours	UpdatedCourseCreditHours
1	1	Computer Programming	3	3
2	2	Computer Organization and Assembly	3	6
3	3	Computer Programming Lab	1	3

	StudentId	StudentName	StudentBatch	CGPA	UpdatedCGPA
1	1	Ali	2013	3.3	1.65
2	2	Aysha	2013	4	2
3	3	Ahmed	2013	2.2	1.1

	StudentId	StudentName	StudentBatch	UpdatedCGPA
1	1	Ali	2013	3.3
2	2	Aysha	2013	2
3	3	Ahmed	2013	0.733333333...