National University of Computer and Emerging Sciences

Lab Manual 3

“Introduction to SQL Retrieval, Set Operations, Joins”

Database Systems

Spring 2022

Department of Computer Science
FAST-NU, Lahore, Pakistan

Table of Contents

Objective

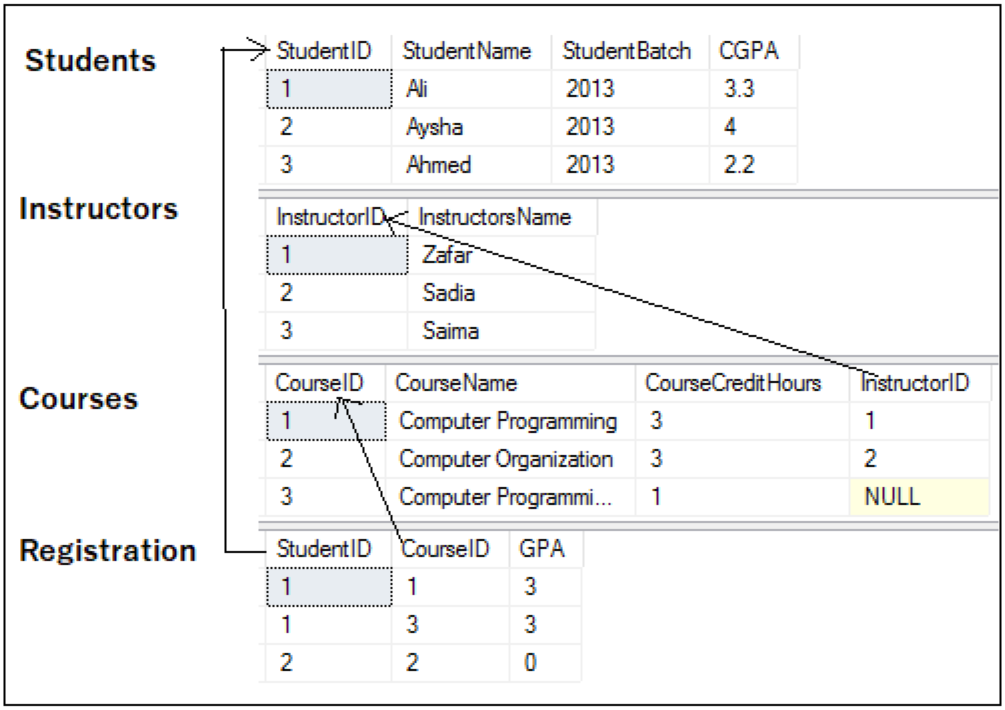
The purpose of this lab activity is to familiarize you with the concept of SQL Set Operations and Joins.

Task Distribution

Total Time	150 Minutes
Select From Where	15 Minutes
Set Operations	5 Minutes
Joins	10 Minutes
Miscellaneous Functions	10 Minutes
Exercise	110 Minutes

1. 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. Script to create this schema is given in Lab4ExampleSchema.sql file



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

a) Most Basic Select:

SELECT *
FROM <table Name>

* after select means that all columns will be retrieved

Try this

Results

b) Retrieving certain Columns from Select

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

```
SELECT ColumnX, ColumnY, ColumnZ
FROM <tableName>
```

Try this

Results

c) Retrieving certain Rows from Select- WHERE CLAUSE

Like Selection in RA, rows are filter 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 on from clause, or derived columns.

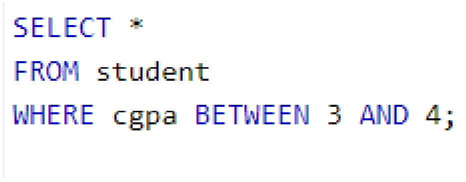
```
SELECT *
FROM <table Name>
where <conditions>
```

Try this

Results

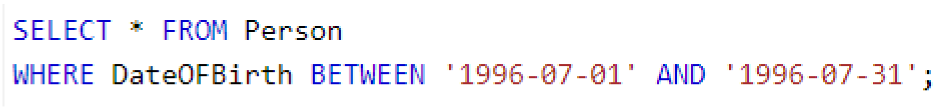
d) Between condition

The BETWEEN operator selects values within a given range. The values can be numbers, text, or dates. (inclusive boundary values)



```
SELECT *
FROM student
WHERE cgpa BETWEEN 3 AND 4;
```

You can also check the range for date data types



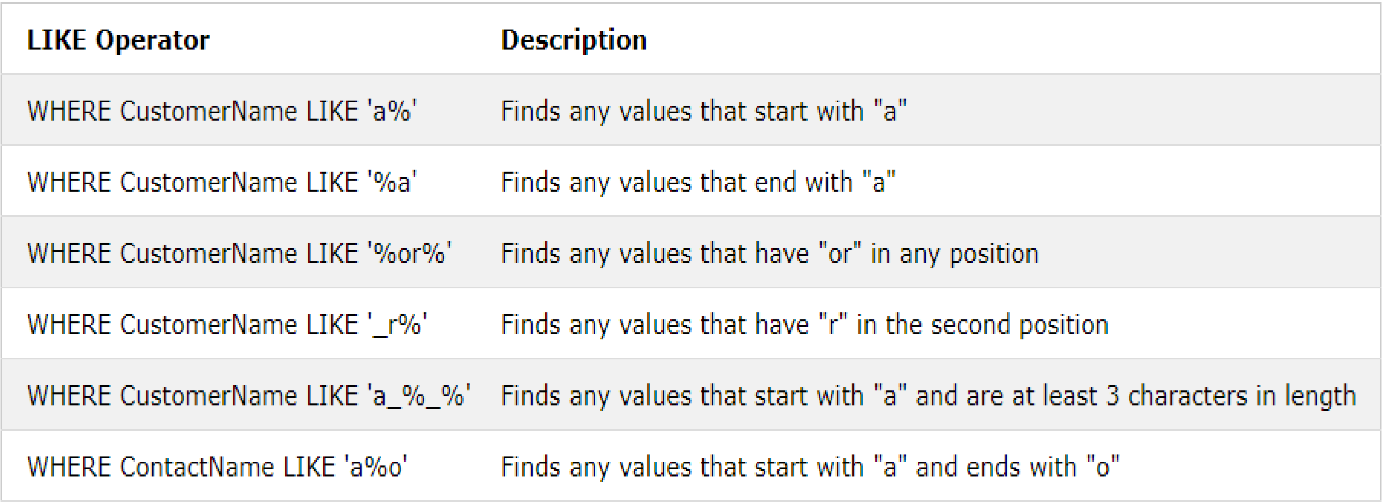
```
SELECT * FROM Person
WHERE DateOFBirth BETWEEN '1996-07-01' AND '1996-07-31';
```

e) Like operator

The LIKE operator is used in a WHERE clause to search for a specified pattern in a column.

There are two wildcards used in conjunction with the LIKE operator:

- % - The percent sign represents zero, one, or multiple characters
- _ - The underscore represents a single character

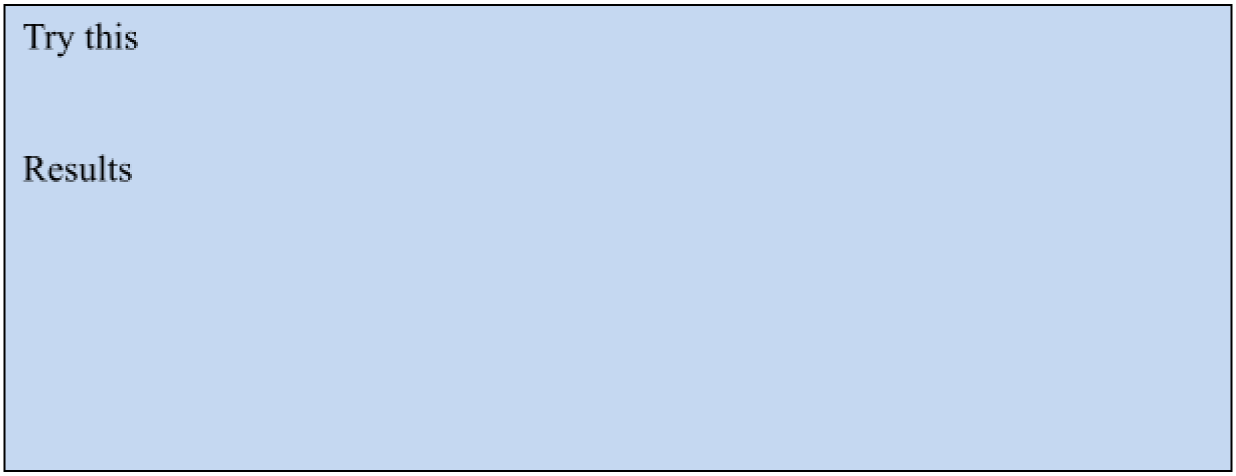


LIKE Operator	Description
WHERE CustomerName LIKE 'a%'	Finds any values that start with "a"
WHERE CustomerName LIKE '%a'	Finds any values that end with "a"
WHERE CustomerName LIKE '%or%'	Finds any values that have "or" in any position
WHERE CustomerName LIKE '_r%'	Finds any values that have "r" in the second position
WHERE CustomerName LIKE 'a_%_%'	Finds any values that start with "a" and are at least 3 characters in length
WHERE ContactName LIKE 'a%o'	Finds any values that start with "a" and ends with "o"

f) 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.

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



Try this

Results

2. Set operations

Result of two (or more) select queries can be combined using Set operations such as UNION, INTESECT, EXCEPT.

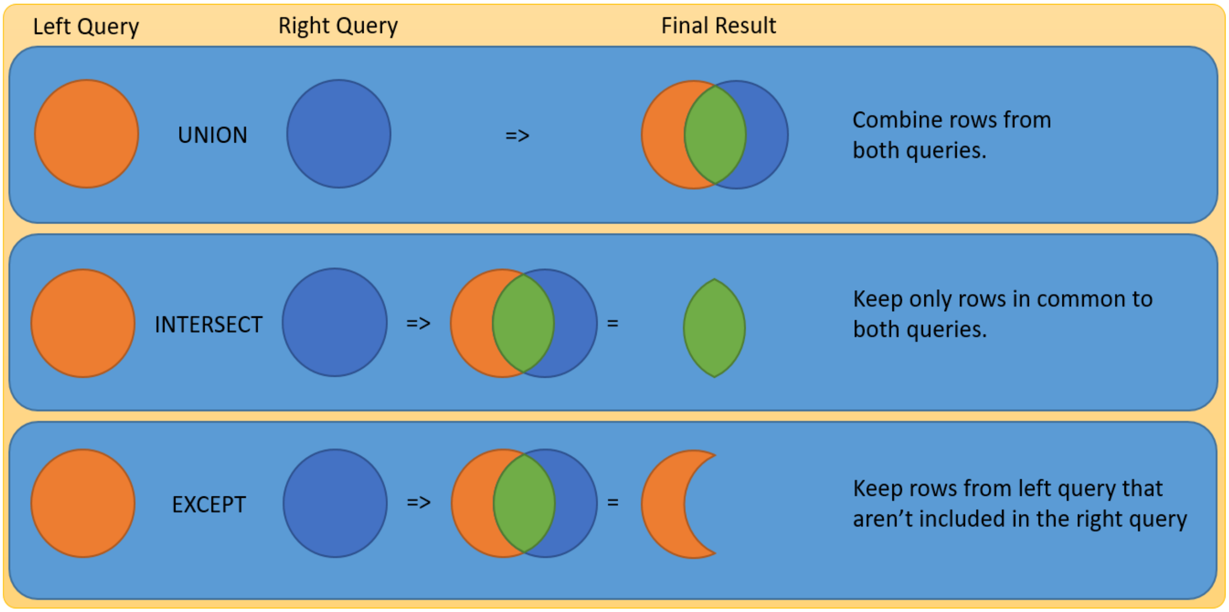
Syntax

```
Select ColumnX, ColumnY
From Table1

Union/Intersect/Except

Select ColumnA, ColumnB
From Table2
```

Visual Explanation of UNION, INTERSECT, and EXCEPT operators



Left Query

Right Query

Final Result

UNION =>

Combine rows from both queries.

INTERSECT => =

Keep only rows in common to both queries.

EXCEPT => =

Keep rows from left query that aren’t included in the right query

NOTE: The output of first select query should have same number and type of column as of second select query.

Try this- error to look out for in set operations

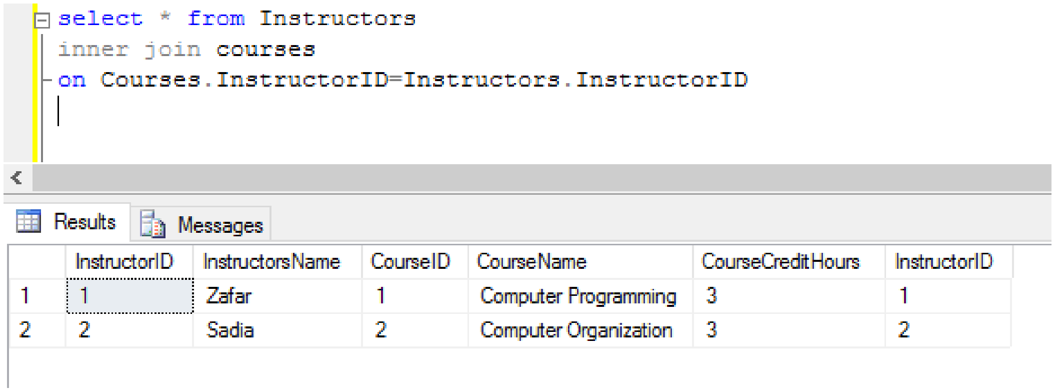
3. Join Operation

We will use the following tables in examples

a) Inner Join:

Returns only those rows that match in both tables.

SELECT *
FROM <table1> inner join <table2>
ON <Joining Condition>

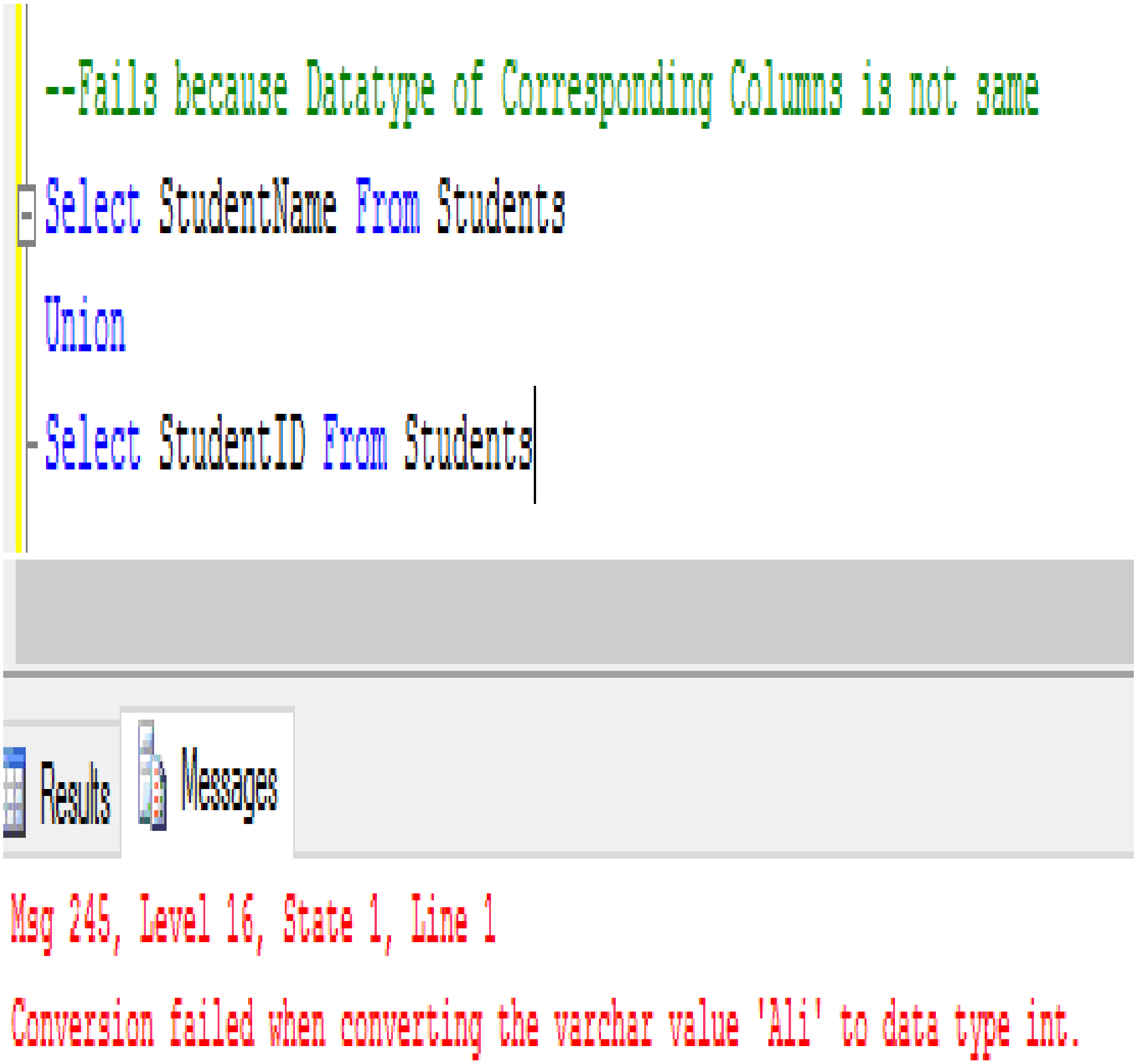


```
select * from Instructors
inner join courses
on Courses.InstructorID=Instructors.InstructorID
```

	InstructorID	InstructorsName	CourseID	CourseName	CourseCreditHours	InstructorID
1	1	Zafar	1	Computer Programming	3	1
2	2	Sadia	2	Computer Organization	3	2

b) Left/Right/Full Outer Join

Left Join: Returns all the rows of Left table with corresponding row or null row of right table
Right Join: Returns all the rows of Right table with corresponding row or null row of Left table
Full Join: Union of Left and Right Outer join
SELECT * FROM <table1> Left/Right/Full join <table2> ON <Joining Condition>



```
--Fails because Datatype of Corresponding Columns is not same
Select StudentName From Students
Union
Select StudentID From Students
```

```
Msg 245, Level 16, State 1, Line 1
Conversion failed when converting the varchar value 'Ali' to data type int.
```

Try these

c) Cross Join

It’s a cross product of two tables, no ON condition is required here

```
SELECT *  FROM <table1> cross Join <table2>
```

Try this

d) Joining More than two tables

```
SELECT *
FROM <table1>
Left/Right/Full/Inner join <table2> ON <Joining Condition>
Left/Right/Full/Inner join <table3> ON <Joining Condition>
Left/Right/Full/Inner join <table4> ON <Joining Condition>
```

Try this

Appendix

Some Useful clause

Distinct

```
--% eliminates duplicated.
Select Distinct Departments from students
```

Like

```
--% for Any string of zero or more characters.
Select * from students where studentName like '%ed%'
-- _  for Any single character.
Select * from students where studentName like 'Ahm_d'
--[] for Any single character within the specified range ([a-f]) or set
([abcdef]).
select * From Students where studentName like 'Ahm[ae]d'
--[^] for Any single character not within the specified range ([a-f]) or set
([abcdef]).
select * From Students where studentName like 'Ahm[^a]d'
```

is null/ is not null

```
select * from Course where InstructorID is null
select * from Course where InstructorID is not null
```

between

```
select * From Students where studentId between 1 and 10
select * From Orders  where orderDate between '2-2-2001' and '2-2-2010'
select *, year(orderDate)   as Year From Orders   where year(orderDate) between
2001 and 2010
```

Some usefull functions

```
isNull(col,value) - replces the null entry with value
CAST ( expression AS data_type )
CONVERT ( data_type, expression)
```

DATE FUNCTIONS

```
DATEPART(datepart, date)  --returns the datepart of date
Year(date)-- returns the Year of date
Month(date) --returns month of  date
Day(date) --returns Day from  date
DATEDIFF ( datepart , startdate , enddate ) --returns the difference in start
and end date in datepart (eg year,days ,months)
```

STRING FUNCTIONS

```
UPPER(String)
LOWER(String
LEFT(String,7) -- returns left 7 Characters
RIGHT(String,7)
LEN(String)
LTRIM (String) -- Trim the left end of string
RTRIM(String)
SUBSTRING (String, 8, 7)
CHARINDEX ('demo', String) -will return the starting index of ‘demo’ in String
REPLACE (String,'s','$') REVERSE (String)
```