

Subject: Database Management System Laboratory

EXPERIMENT NO:--02

TITLE:Implement joins and aggregate functions for tables in DB.

LEARNING OBJECTIVES:

1. To study the fundamental concepts of database management.
2. To learn the basic issues of transaction processing and concurrency control.
3. To learn a powerful, flexible and scalable general-purpose distributed database.

THEORY:

Types of SQL Joins:-

1. SQL INNER JOIN

The INNER JOIN keyword selects all rows from both the tables as long as the condition is satisfied. This keyword will create the result set by combining all rows from both the tables where the condition satisfies i.e value of the common field will be the same.

Syntax:

```
SELECT table1.column1,table1.column2,table2.column1,... FROM table1 INNER JOIN table2  
ON table1.matching_column = table2.matching_column;
```

2. SQL LEFT JOIN

A LEFT JOIN returns all rows from the left table, along with matching rows from the right table. If there is no match, NULL values are returned for columns from the right table. LEFT JOIN is also known as LEFT OUTER JOIN.

Syntax

Prepared by

Mr. N. I. Bhopale
(Subject Teacher)

Verified by

Dr. B. S. Agarkar
(HOD, Deptt. of ECE)

```
SELECT table1.column1,table1.column2,table2.column1,...  
FROM table1  
LEFT JOIN table2  
ON table1.matching_column = table2.matching_column;
```

3. SQL RIGHT JOIN

RIGHT JOIN returns all the rows of the table on the right side of the join and matching rows for the table on the left side of the join. It is very similar to LEFT JOIN for the rows for which there is no matching row on the left side, the result-set will contain null. RIGHT JOIN is also known as RIGHT OUTER JOIN.

Syntax

```
SELECT table1.column1,table1.column2,table2.column1,...  
FROM table1  
RIGHT JOIN table2  
ON table1.matching_column = table2.matching_column;
```

4. SQL FULL JOIN

FULL JOIN creates the result-set by combining results of both LEFT JOIN and RIGHT JOIN. The result-set will contain all the rows from both tables. For the rows for which there is no matching, the result-set will contain NULL values.

Syntax

```
SELECT table1.column1,table1.column2,table2.column1,...  
FROM table1  
FULL JOIN table2  
ON table1.matching_column = table2.matching_column;
```

Prepared by

Mr. N. I. Bhopale
(Subject Teacher)

Verified by

Dr. B. S. Agarkar
(HOD, Deptt. of ECE)

Sanjivani Rural Education Society's College of Engineering, Kopargaon
Department of Electronics and Computer Engineering
TITLE: Experiment Write-up (EW)

SQL Aggregate Functions:-

- **AVG()** - Returns the average value
- **COUNT()** - Returns the number of rows
- **FIRST()** - Returns the first value
- **LAST()** - Returns the last value
- **MAX()** - Returns the largest value
- **MIN()** - Returns the smallest value
- **SUM()** - Returns the sum

Example:

mysql> show databases;

Prepared by

Mr. N. I. Bhopale
(Subject Teacher)

Verified by

Dr. B. S. Agarkar
(HOD, Deptt. of ECE)

Sanjivani Rural Education Society's College of Engineering, Kopergaon
Department of Electronics and Computer Engineering
TITLE: Experiment Write-up (EW)

Database
information_schema
buysale
car
college
college1
db
dummy
dummy123
employee
learn
localdb
localdb1
localdb2
localdb3
mysql
mytransaction
new_dummy
newdummy
performance_schema
quiz
sakila
sample
sanjivani
sanjivani1
sanjivani123
sanjivani2
sanjivanidb
scoe
student
sys
test
test1
trade
transaction
userdb
world
xyz

Prepared by

Mr. N. I. Bhopale
(Subject Teacher)

Verified by

Dr. B. S. Agarkar
(HOD, Deptt. of ECE)

Sanjivani Rural Education Society's College of Engineering, Kopergaon
Department of Electronics and Computer Engineering
TITLE: Experiment Write-up (EW)

```
mysql> create database school; Query OK, 1 row
affected (0.01 sec)
mysql> use school;
Database changed
mysql> create table student (rn int primary key not null,
-> name varchar(22) not null,
-> marks varchar(22) not null);
Query OK, 0 rows affected (0.06 sec)
```

```
mysql> insert into student (rn,name,marks) values
-> (1,"sarthak",55),
-> (2,"nilesh",77),
-> (3,"shruti",66);
Query OK, 3 rows affected (0.00 sec)
```

```
mysql> select * from student;
```

rn	name	marks
1	sarthak	55
2	nilesh	77
3	shruti	66

```
mysql> select * from student order by rn desc;
```

rn	name	marks
3	shruti	66
2	nilesh	77
1	sarthak	55

```
mysql> select * from student order by rn asc;
```

Prepared by

Mr. N. I. Bhopale
(Subject Teacher)

Verified by

Dr. B. S. Agarkar
(HOD, Deptt. of ECE)

Sanjivani Rural Education Society's College of Engineering, Kopargaon
Department of Electronics and Computer Engineering
TITLE: Experiment Write-up (EW)

rn	name	marks
1	sarthak	55
2	nilesh	77
3	shruti	66

mysql> select distinct(name) from student;

name
sarthak
nilesh
shruti

mysql> select * from student;

rn	name	marks
1	sarthak	55
2	nilesh	77
3	shruti	66
4	nilesh	88

mysql> select min(marks) from student;

min(marks)
55

mysql> select max(marks),name,rn from student;

max(marks)	name	rn
88	sarthak	1

Prepared by

Verified by

Mr. N. I. Bhopale
(Subject Teacher)

Dr. B. S. Agarkar
(HOD, Deptt. of ECE)

Sanjivani Rural Education Society's College of Engineering, Kopergaon
Department of Electronics and Computer Engineering
TITLE: Experiment Write-up (EW)

mysql> select avg(marks) from student;

avg(marks)
71.5

mysql> insert into student (rn,name,marks) values(4,"abc",66);

Query OK, 1 row affected (0.02 sec)

mysql> select * from student;

rn	name	marks
1	sarthak	55
2	nilesh	77
3	shruti	66
4	abc	66

mysql> alter table student rename stude;

Query OK, 0 rows affected (0.03 sec)

mysql> select * from stude;

rn	name	marks
1	sarthak	55
2	nilesh	77
3	shruti	66

mysql> update stude set marks=89 where rn=1; Query OK, 1 row affected (0.00 sec)

Rows matched: 1 Changed: 1 Warnings: 0

Prepared by

Mr. N. I. Bhopale
(Subject Teacher)

Verified by

Dr. B. S. Agarkar
(HOD, Deptt. of ECE)

Sanjivani Rural Education Society's College of Engineering, Kopergaon
Department of Electronics and Computer Engineering
TITLE: Experiment Write-up (EW)

```
mysql> select * from stude;
```

rn	name	marks
1	sarthak	89
2	nilesh	77
3	shruti	66

NOTE : Please ensure that you also add the Industrial Problem (2) in your submission/document along with the existing content.

References for Theory:

1. Silberschatz A., Korth H., Sudarshan S., "Database System Concepts", MGH
2. Connally T, Begg C., "Database Systems", Pearson Education
3. Raghurama Krishan, "Database Management Systems", McGrawHill
4. S.K.Singh, "Database Systems : Concepts, Design and Application", Pearson

CONCLUSION: _____

Prepared by

Mr. N. I. Bhopale
(Subject Teacher)

Verified by

Dr. B. S. Agarkar
(HOD, Deptt. of ECE)