LAB REPORT ON DBMS USING SQL

Summer-2019

Course Code : SE224

Course Title : Database Management System Lab.

Submitted To

NUSRAT JAHAN

Senior Lecturer @ Department Of SWE

Daffodil International University

Submitted By

NAZMUL HASSAN

Student I.D : 181-35-2491

Submission Date : 27-08-2019

Introduction :

❖ SQL is a standard language for storing, manipulating and retrieving data in databases.

What is SQL !

➢ SQL stands for Structured Query Language
➢ SQL lets you access and manipulate databases
➢ SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standardization (ISO) in 1987

What Can SQL do?

✓ SQL can execute queries against a database
✓ SQL can retrieve data from a database
✓ SQL can insert records in a database
✓ SQL can update records in a database
✓ SQL can delete records from a database
✓ SQL can create new databases
✓ SQL can create new tables in a database
✓ SQL can create stored procedures in a database
✓ SQL can create views in a database
✓ SQL can set permissions on tables, procedures, and views

RDBMS :
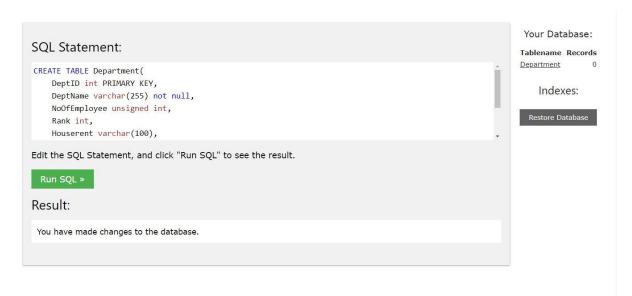
RDBMS stands for Relational Database Management System.

RDBMS is the basis for SQL, and for all modern database systems such as MS SQL Server, IBM DB2, Oracle, MySQL, and Microsoft Access.

The data in RDBMS is stored in database objects called tables. A table is a collection of related data entries and it consists of columns and rows.

We are going to create three tables with at least seven tuple in our DBMS.

➢ Creating "Department" table

SQL Statement:

```
CREATE TABLE Department(
    DeptID int PRIMARY KEY,
    DeptName varchar(255) not null,
    NoOfEmployee unsigned int,
    Rank int,
    Houserent varchar(100),
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

You have made changes to the database.

Your Database:

Tablename	Records
Department	0

Indexes:

Restore Database

➢ Inserting data into "Department" table

SQL Statement:

```
INSERT INTO Department(DeptID,DeptName,NoOfEmployee,Rank,Houserent,Medical,Gratuaty)
Values
(9,'Storage',600,5,'no','yes','no') ,(6,'Production',900,4,'no','no','yes') ,
(10,'Management',300,3,'yes','no','no'),(11,'Stock',500,4,'yes','yes','no'),
(12,'Order',22,5,'yes','yes','yes') ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

You have made changes to the database. Rows affected: 5

Your Database:

Tablename	Records
Department	5

Indexes:

Name of Index
sqlite_autoindex_Departmen

Restore Database

➢ Creating "Employee" table

SQL Statement:

```
CREATE TABLE Employee(
    employee_id int PRIMARY KEY,
    first_name varchar(255),
    last_name varchar(255),
    email varchar(255) UNIQUE,
    phone_number varchar(255),
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

You have made changes to the database.

➢ Inserting data into "Employee" table

SQL Statement:

```
INSERT INTO Employee ( employee_id , first_name , last_name , email , phone_number , hire_date ,
job_id , salary , manager_id , department_id )
Values
(100, 'Steven', 'King', 'steven.king@sqltutorial.org', '515.123.4567', '1987-06-17', 4, 24000.00,
NULL, 9) ,
(101, 'Neena', 'Kochhar', 'neena.kochhar@sqltutorial.org', '515.123.4568', '1989-09-21', 5,
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

You have made changes to the database. Rows affected: 12

➢ Creating "Product" table

SQL Statement:

```
Create table Product(
    ProductID int PRIMARY KEY,
    ProductTitle varchar(255),
    Decription varchar(500),
    ProductionDate varchar(255),
    ExpiryDate varchar(255),
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

You have made changes to the database.

Your Database:

Tablename	Records
Department	5
Employee	12
Product	0

Indexes:

Name of Index
sqlite_autoindex_Departmen
sqlite_autoindex_Employee_
sqlite_autoindex_Product_1

Restore Database

➢ Inserting data into "Product" table

SQL Statement:

```
Insert into Product(
    ProductID,ProductTitle,Decription,ProductionDate,ExpiryDate,DepID,Price
)
Values
(102,'Second Product','Here is a short description for this product','22-08-2019','22-08-
2022',6,250),
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

You have made changes to the database. Rows affected: 6

Your Database:

Tablename	Records
Department	5
Employee	12
Product	7

Indexes:

Name of Index
sqlite_autoindex_Departmen
sqlite_autoindex_Employee_
sqlite_autoindex_Product_1

Restore Database

On next page I will show all Tables structure with data view.

⊞ Department Table

SQL Statement:

```
SELECT * FROM [Department]
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 5

DeptID	DeptName	NoOfEmployee	Rank	Houserent	Medical	Gratuaty
9	Storage	600	5	no	yes	no
6	Production	900	4	no	no	yes
10	Management	300	3	yes	no	no
11	Stock	500	4	yes	yes	no
12	Order	22	5	yes	yes	yes

⊞ Employee Table

SQL Statement:

```
SELECT * FROM [Employee]
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 12

employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	manager_id	
100	Steven	King	steven.king@sqltutorial.org	515.123.4567	1987-06-17	4	24000	null	
101	Neena	Kochhar	neena.kochhar@sqltutorial.org	515.123.4568	1989-09-21	5	17000	100	
102	Lex	De Haan	lex.dehaan@sqltutorial.org	515.123.4569	1993-01-13	5	17000	100	
103	Alexander	Hunold	alexander.hunold@sqltutorial.org	590.423.4567	1990-01-	9	9000	102	

🔢 Product Table

Your Database:

Tablename	Records
Department	5
Employee	12
Product	7

Indexes:

Name of Index

sqlite_autoindex_Department_1
sqlite_autoindex_Employee_1
sqlite_autoindex_Product_1

Restore Database

SQL Statement:

```
SELECT * FROM [Product]
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 7

ProductID	ProductTitle	Decription	ProductionDate	ExpiryDate	DepID	Price
101	First Product	Here is a short description for this product	22-08-2019	22-08-2022	9	150
102	Second Product	Here is a short description for this product	22-08-2019	22-08-2022	6	250
103	Third Product	Here is a short description for this product	22-08-2019	22-08-2022	6	350
104	Forth Product	Here is a short description for this product	22-08-2019	22-08-2025	6	850
105	Fifth Product	Here is a short description for this product	22-08-2019	22-08-2022	6	250
106	Sixth Product	Here is a short description for this product	22-08-2019	22-08-2022	6	1250

Alright, as per as discussion I have created all tables.

Now we will start running query on our DBMS.

Its fun on when setup is completed.

✥ Find only Employee ID, Email & Salary from Employee table.

SQL Statement:

```
SELECT Distinct employee_id,email,salary FROM Employee
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 12

employee_id	email	salary
100	steven.king@sqltutorial.org	24000
101	neena.kochhar@sqltutorial.org	17000
102	lex.dehaan@sqltutorial.org	17000
103	alexander.hunold@sqltutorial.org	9000
104	bruce.arnst@sqltutorial.org	6000
105	david.austin@sqltutorial.org	4800

✥ Write a line of comment with proper syntax.

SQL Statement:

```
--Here is the comment start with 2 dash and a colon :
Select * From Employee ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 11

employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	manager_id	
100	Steven	King	steven.king@sqltutorial.org	515.123.4567	1987-06-17	4	24000	103	9
101	Neena	Kochhar	neena.kochhar@sqltutorial.org	515.123.4568	1989-09-21	5	17000	100	9
103	Alexander	Hunold	alexander.hunold@sqltutorial.org	590.423.4567	1990-01-03	9	9000	102	6
104	Bruce	ernst	bruce.arnst@sqltutorial.org	590.423.4568	1991-05-	9	6000	103	6

⬇ Add a new column called image on Product table.

SQL Statement:

```
Alter Table Product
    ADD image varchar(255);
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 7

ProductID	ProductTitle	Decription	ProductionDate	ExpiryDate	DepID	Price	image
101	First Product	Here is a short description for this product	22-08-2019	22-08-2022	9	150	*null*
102	Second Product	Here is a short description for this product	22-08-2019	22-08-2022	6	250	*null*
103	Third Product	Here is a short description for this product	22-08-2019	22-08-2022	6	350	*null*
104	Forth Product	Here is a short description for this product	22-08-2019	22-08-2025	6	850	*null*
105	Fifth Product	Here is a short description for this product	22-08-2019	22-08-2022	6	250	*null*

⬇ Select Product where product id is more than 102 and Price is more than 200 .

SQL Statement:

```
SELECT * FROM Product
    Where ProductID > 102 AND Price > 200 ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 5

ProductID	ProductTitle	Decription	ProductionDate	ExpiryDate	DepID	Price
103	Third Product	Here is a short description for this product	22-08-2019	22-08-2022	6	350
104	Forth Product	Here is a short description for this product	22-08-2019	22-08-2025	6	850
105	Fifth Product	Here is a short description for this product	22-08-2019	22-08-2022	6	250
106	Sixth Product	Here is a short description for this product	22-08-2019	22-08-2022	6	1250
107	Seventh Product	Here is a short description for this product	22-08-2019	22-08-2022	6	950

Find all data from Department and make sure it have medical and House rent avail.

SQL Statement:

```
SELECT * FROM Department
    Where Medical = 'yes' or Houserent = 'yes' ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 4

DeptID	DeptName	NoOfEmployee	Rank	Houserent	Medical	Gratuaty
9	Storage	600	5	no	yes	no
10	Management	300	3	yes	no	no
11	Stock	500	4	yes	yes	no
12	Order	22	5	yes	yes	yes

Find average salary of employee using aggregate function.

SQL Statement:

```
SELECT AVG(salary) FROM Employee
    Where department_id = 6 ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 1

AVG(salary)
5760

Find all price between 250-850 from product table

SQL Statement:

```
SELECT * FROM Product
    Where Price BETWEEN 250 and 850 ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 4

ProductID	ProductTitle	Decription	ProductionDate	ExpiryDate	DepID	Price
102	Second Product	Here is a short description for this product	22-08-2019	22-08-2022	6	250
103	Third Product	Here is a short description for this product	22-08-2019	22-08-2022	6	350
104	Forth Product	Here is a short description for this product	22-08-2019	22-08-2025	6	850
105	Fifth Product	Here is a short description for this product	22-08-2019	22-08-2022	6	250

How many products in the Product table ?

SQL Statement:

```
SELECT COUNT(ProductID) FROM Product
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 1

COUNT(ProductID)
7

Your Database:

Tablename	Records
Department	5
Product	7
Employee	11

Indexes:

Name of Index
sqlite_autoindex_Department_1
sqlite_autoindex_Product_1
sqlite_autoindex_Employee_1
sqlite_autoindex_Employee_2

Restore Database

✛ Erase employee tuple where employee ID is 102 from Employee table

SQL Statement:

```
DELETE FROM Employee
    Where employee_id = 102 ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 11

employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	manager_id	
100	Steven	King	steven.king@sqltutorial.org	515.123.4567	1987-06-17	4	24000	103	؟
101	Neena	Kochhar	neena.kochhar@sqltutorial.org	515.123.4568	1989-09-21	5	17000	100	؟
103	Alexander	Hunold	alexander.hunold@sqltutorial.org	590.423.4567	1990-01-03	9	9000	102	؟
104	Bruce	ernst	bruce.arnst@sqltutorial.org	590.423.4568	1991-05-	9	6000	103	؟

✛ Find total product and all price using product table and group by Department ID

SQL Statement:

```
SELECT COUNT(ProductID),SUM(Price)
FROM Product
Group by DepID;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 2

COUNT(ProductID)	SUM(Price)
6	3900
1	150

Find all information of department where rank between 5-4

SQL Statement:

```
SELECT * FROM Department
    Where Rank IN(5,4) ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 4

DeptID	DeptName	NoOfEmployee	Rank	Houserent	Medical	Gratuaty
9	Storage	600	5	no	yes	no
6	Production	900	4	no	no	yes
11	Stock	500	4	yes	yes	no
12	Order	22	5	yes	yes	yes

Find employee ID , First Name , email , Department ID , house rent and gratuity by joining Department and Employee table.

SQL Statement:

```
SELECT employee_id,first_name,email,DeptID,Houserent,Gratuaty from Employee
INNER JOIN Department
ON  Employee.department_id = Department.DeptID ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 11

employee_id	first_name	email	DeptID	Houserent	Gratuaty
100	Steven	steven.king@sqltutorial.org	9	no	no
101	Neena	neena.kochhar@sqltutorial.org	9	no	no
103	Alexander	alexander.hunold@sqltutorial.org	6	no	yes
104	Bruce	bruce.arnst@sqltutorial.org	6	no	yes
105	David	david.austin@sqltutorial.org	6	no	yes
106	Vali	vali.natabala@sqltutorial.org	6	no	yes

🔸 Extract all information where last name is "King"

SQL Statement:

```sql
Select * From Employee
    Where last_name LIKE '%King';
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 1

employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	manager_id	departn
100	Steven	King	steven.king@sqltutorial.org	515.123.4567	1987-06-17	4	24000	103	9

◄ ►

🔸 Findout minimum salary of Employee

SQL Statement:

```sql
SELECT MAX(salary) FROM Employee ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 1

MAX(salary)
24000

⬇ Find maximum salary of Employee

SQL Statement:

```
SELECT MAX(salary) FROM Employee ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

[Run SQL »]

Result:

Number of Records: 1

MAX(salary)
24000

⬇ Find all product without expiry date of "22-08-2025"

SQL Statement:

```
SELECT * FROM Product
    Where NOT ExpiryDate = '22-08-2025' ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

[Run SQL »]

Result:

Number of Records: 6

ProductID	ProductTitle	Decription	ProductionDate	ExpiryDate	DepID	Price
101	First Product	Here is a short description for this product	22-08-2019	22-08-2022	9	150
102	Second Product	Here is a short description for this product	22-08-2019	22-08-2022	6	250
103	Third Product	Here is a short description for this product	22-08-2019	22-08-2022	6	350
105	Fifth Product	Here is a short description for this product	22-08-2019	22-08-2022	6	250
106	Sixth Product	Here is a short description for this product	22-08-2019	22-08-2022	6	1250
107	Seventh Product	Here is a short description for this product	22-08-2019	22-08-2022	6	950

♣ Find first name, last name, email & department ID if there manager ID is NULL.

SQL Statement:

```
SELECT first_name,last_name,email,department_id FROM Employee
    Where manager_id IS NULL ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 1

first_name	last_name	email	department_id
Steven	King	steven.king@sqltutorial.org	9

♣ Show department table in descending order.

SQL Statement:

```
SELECT * FROM Department
Order By DeptID DESC ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 5

DeptID	DeptName	NoOfEmployee	Rank	Houserent	Medical	Gratuaty
12	Order	22	5	yes	yes	yes
11	Stock	500	4	yes	yes	no
10	Management	300	3	yes	no	no
9	Storage	600	5	no	yes	no
6	Production	900	4	no	no	yes

Find sum of all salary of Employee

SQL Statement:

```sql
SELECT sum(salary) FROM Employee ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 1

sum(salary)
106700

Put manager ID as 103 if there any NULL field for employee on manager ID column.

SQL Statement:

```sql
Update Employee
    SET manager_id = 103
    Where manager_id is null ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 12

employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	manager_id	
100	Steven	King	steven.king@sqltutorial.org	515.123.4567	1987-06-17	4	24000	103	9
101	Neena	Kochhar	neena.kochhar@sqltutorial.org	515.123.4568	1989-09-21	5	17000	100	9
102	Lex	De Haan	lex.dehaan@sqltutorial.org	515.123.4569	1993-01-13	5	17000	100	9
103	Alexander	Hunold	alexander.hunold@sqltutorial.org	590.423.4567	1990-01-	9	9000	102	6

➕ Find all information of salary where salary is more than 20000

SQL Statement:

```
SELECT * FROM Employee
    Where salary > 20000
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 1

employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	manager_id	departm
100	Steven	King	steven.king@sqltutorial.org	515.123.4567	1987-06-17	4	24000	*null*	9

➕ Find first name, email, salary from Employee table where salary is more than 15000 and you must have to use subquery.

SQL Statement:

```
SELECT first_name,email,salary
    From Employee
    where employee_id IN (Select employee_id where salary > 15000);
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 2

first_name	email	salary
Steven	steven.king@sqltutorial.org	24000
Neena	neena.kochhar@sqltutorial.org	17000

➕ Find all tuple from department where No of employee is greater than 500 also they have Medical facility.

SQL Statement:

```
SELECT * FROM Department
    Where NoOfEmployee >500 AND Medical = 'yes' ;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 1

DeptID	DeptName	NoOfEmployee	Rank	Houserent	Medical	Gratuaty
9	Storage	600	5	no	yes	no

➕ Erase all data from employee table.

SQL Statement:

```
DROP TABLE Employee
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

You have made changes to the database.

Your Database:

Tablename	Records
Department	5
Product	7

Indexes:

Name of Index
sqlite_autoindex_Department_1
sqlite_autoindex_Product_1

Restore Database

[THE END]