Index

S.No.	Experiment	Signature
1.	To study DDL and DML commands	
2.	To study TCL and DCL commands	
3.	Write SQL queries to create the following database: Flights(flno, from, to, distance, departs) Aircraft(aid, aname, range) Certified(eid, aid) Employees(eid, ename, salary) By definition, pilots are those employees who are certified on at least one aircraft. An aircraft can be used for any flight provided it has sufficient range. Pilots can pilot any flight provided they are certified on an aircraft with sufficient range.	
4.	For the database created in ques 3, perform the following queries: a. Find eid's of pilots who are certified on some Boeing. b. Find names of pilots who are certified on some Boeing. c. Find aid's of aircraft that can fly non-stop from LA to NY. Assume you don't already know the distance. d. Find flno of flights that can be piloted by every pilot whose salary is over $100,000.	
5.	For the database created in ques 3, perform the following queries: a. Find names of pilots who can operate planes with a range greater than 3,000 miles, but are not certified on any Boeing.	

	b. Find eid of employee(s) with the highest salary. c. Find eid of employee(s) with the second highest salary. d. Find eid's of employees certified on exactly three aircraft.	
6.	Write SQL queries to create the following database: Author (authorname, citizenship, birthyear) Book(isbn, title, authorname) Topic(isbn, subject) Branch(libname, city) Instock(isbn, libname, quantity)	
7.	For the database created in ques 6, perform the following queries: a. Give all authors born after 1940. b. Give the names of libraries in Sydney. c. Give the cities where each book is held. d. Give the title of each book on the topic of either alcohol or drugs.	
8.	For the database created in ques 6, perform the following queries: a. Give the title and author of each book of which at least two copies are held in a branch located in Melbourne. b. Give the name of each Italian author who wrote an autobiography. c. Give the total number of books in stock in the branch called Fisher. d. This could mean the number of different titles, or the number of physical copies. e. Give the total number of books in stock in the branches located in Sydney.	
9.	Write SQL queries to create the following database: o Suppliers*(sid:integer,sname:varchar(10),address: varchar(10),)* o Parts*(pid: integer, pname: varchar(10), color: varchar(10))* o Catalog*(sid: integer, pid: integer, cost: real)*	

10.	For the database created in ques 8, perform the following queries: a. After creating the table I realized that I forgot to mention that only green, red, blue and black color parts are allowed in parts relation. Write a query to add this constraint in parts relation. b. For every supplier that only supplies green parts, print the name of the supplier and the total number of parts that she supplies. c. Increase the cost of each product by 10% which are green in color and are supplied by supplier = “John”. d. Authorize a user “Mary” to update color in “Parts” table. She can also pass the authorization to other users.	
11.	For the database created in ques 8, perform the following queries: a. Delete all the parts supplied by supplier “King”. b. Drop the attribute color from parts relation. c. Create a view for customers so that they can see only the supplier names, the name of parts supplied by that supplier and their cost.	

Experiment - 1

Aim: To study DDL and DCL commands

Data Definition Language (DDL): DDL changes the structure of the table like creating a table, deleting a table, altering a table, etc. All the command of DDL are auto-committed that means it permanently save all the changes in the database.

Here are some commands that come under DDL:

1. CREATE
2. ALTER
3. DROP
4. TRUNCATE

1. CREATE : It is used to create a new table in the database.
    a. Syntax: CREATE TABLE TABLE_NAME (COLUMN_NAME DATATYPES[,….]);
    b. Example:
        i. CREATE TABLE EMPLOYEE(Name VARCHAR2(20), Email VARCHAR2(100), DOB DATE);
        ii. CREATE DATABASE UNIVERSITY;
2. DROP: It is used to delete both the structure and record stored in the table.
    a. Syntax: DROP TABLE table_name;
    b. Example:DROP TABLE EMPLOYEE;
3. ALTER: It is used to alter the structure of the database. This change could be either to modify the characteristics of an existing attribute or probably to add a new attribute.
    a. Syntax:
        i. To add a new column in the table : ALTER TABLE table_name ADD column_name COLUMN-definition;
        ii. To modify existing column in the table:ALTER TABLE table_name MODIFY(column_definitions….);
    b. EXAMPLE:
        i. ALTER TABLE STU_DETAILS ADD(ADDRESS VARCHAR2(20));
        ii. ALTER TABLE STU_DETAILS MODIFY (NAME VARCHAR2(20));

4. TRUNCATE: It is used to delete all the rows from the table and free the space containing the table.
    a. Syntax: TRUNCATE TABLE table_name;
    b. Example: TRUNCATE TABLE EMPLOYEE;

Data Manipulation Language (DML): DML commands are used to modify the database. It is responsible for all form of changes in the database. The command of DML is not auto-committed that means it can't permanently save all the changes in the database. They can be rollback.

Here are some commands that come under DML:

1. INSERT
2. UPDATE
3. DELETE

1. INSERT: INSERT statement is a SQL query. It is used to insert data into the row of a table.
    a. Syntax:
        i. INSERT INTO TABLE_NAME(col1, col2, col3,…. Col N) VALUES (value1, value2, value3, …. valueN);
        OR
        INSERT INTO TABLE_NAME VALUES (value1, value2, value3, …. valueN);
    b. Example: INSERT INTO myDB(Stu_Id, Stu_Name, Stu_Marks) VALUES (104, "Anmol", 89);
2. UPDATE: This command is used to update or modify the value of a column in the table.
    a. Syntax: UPDATE table_name SET [column_name1= value1,…column_nameN = valueN] [WHERE CONDITION]
    b. Example: UPDATE students SET Stu_Name = "Anmol Singh" WHERE Stu_Id = 104

3. DELETE: It is used to remove one or more row from a table.
    a. Syntax: DELETE FROM table_name [WHERE condition];
    b. Example: DELETE FROM myDB WHERE Stu_Id = 104

Experiment - 2

AIM: To study DCL and TCL Commands

Data Control Language (DCL): DCL commands are used to grant and take back authority from any database user.

Here are some commands that come under DCL:

1. Grant
2. Revoke

1. Grant: It is used to give user access privileges to a database.
    a. Syntax : GRANT SELECT, UPDATE ON MY_TABLE TO SOME_USER, ANOTHER_USER;
    b. Example: GRANT SELECT ON Users TO'Ayush'@'localhost;
2. Revoke: It is used to take back permissions from the user.
    a. Syntax: REVOKE privilege_nameON object_nameFROM {user_name |PUBLIC |role_name}
    b. Example: REVOKE SELECT, UPDATE ON student FROM IT1,IT2;

Transaction Control Language (TCL): TCL commands can only use with DML commands like INSERT, DELETE and UPDATE only. These operations are automatically committed in the database that's why they cannot be used while creating tables or dropping them.

Here are some commands that come under TCL:

1. COMMIT
2. ROLLBACK
3. SAVEPOINT

1. Commit: Commit command is used to save all the transactions to the database.
    a. Syntax: COMMIT;
    b. Example: DELETE FROM CUSTOMERS WHERE AGE = 25;

COMMIT;

2. Rollback: Rollback command is used to undo transactions that have not already been saved to the database.
    a. Syntax: ROLLBACK;
    b. Example: DELETE FROM CUSTOMERS WHERE AGE = 25;
       ROLLBACK;
3. SAVEPOINT: It is used to roll the transaction back to a certain point without rolling back the entire transaction.
    a. Syntax: SAVEPOINT SAVEPOINT_NAME;

Experiment – 3

Aim –

Write SQL queries to create the following database:

Flights(flno, from, to, distance, departs)
Aircraft(aid, aname, range)
Certified(eid, aid)
Employees(eid, ename, salary)

By definition, pilots are those employees who are certified on at least one aircraft. An aircraft can be used for any flight provided it has sufficient range. Pilots can pilot any flight provided they are certified on an aircraft with sufficient range.

Software Used – MySQL Workbench

Code –

```
CREATE TABLE Flights (
   flno INT PRIMARY KEY,
   `from` VARCHAR(20),
   `to` VARCHAR(20),
   distance INT,
   departs TIME
);
CREATE TABLE Aircraft (
   aid INT PRIMARY KEY,
   aname VARCHAR(20),
   `range` INT
);
CREATE TABLE Employees (
   eid INT PRIMARY KEY,
   ename VARCHAR(20),
```

```
    salary INT
);
CREATE TABLE Certified (
    eid INT,
    aid INT,
    PRIMARY KEY (eid , aid),
    FOREIGN KEY (eid)
        REFERENCES Employees (eid),
    FOREIGN KEY (aid)
        REFERENCES Aircraft (aid)
);
INSERT INTO Flights VALUES
(101, "LA", "NY", 4362, "12:00:00"),
(201, "ATL", "ORD", 975, "15:20:00"),
(301, "LAX", "JFK", 4000, "13:45:00"),
(401, "ORD", "MIA", 1930, "08:15:00"),
(501, "DEN", "CLT", 2146, "22:50:00"),
(601, "ATL", "CLT", 363, "18:18:00"),
(701, "ORD", "LAX", 2800, "06:32:00"),
(801, "DEL", "FRA", 18000,'07:15:00'),
(802, "BLR", "FRA", 19500,'10:00:00'),
(803, "BLR", "FRA", 17000,'12:00:00');
INSERT INTO Aircraft VALUES
(123,"Airbus",1000),
(302,"Boeing",5000),
```

(306,"Jet01",5000),

(378,"Airbus380",8000),

(456,"Aircraft",500),

(789,"Aircraft02",800),

(951,"Aircraft03",1000);

INSERT INTO Employees VALUES

(1,'Roy',30000),

(2,'Jim',85000),

(3,'Michael',50000),

(4,'Harry',45000),

(5,'Ron',90000),

(6,'Josh',75000),

(7,'Pam',100000);

INSERT INTO Certified VALUES

(1,123),

(2,123),

(1,302),

(5,302),

(7,302),

(1,306),

(2,306),

(1,378),

(2,378),

(4,378),

(6,456),

(3,456),

(5,789),

(6,789),

(3,951),

(1,951),

(1,789);

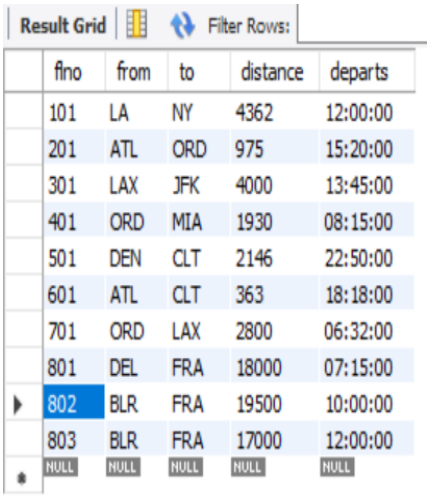
SELECT * FROM Flights;

SELECT * FROM Aircraft;

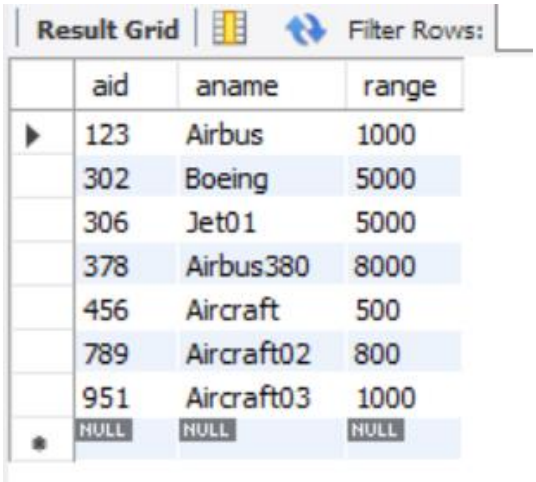
SELECT * FROM Employees;

SELECT * FROM Certified;

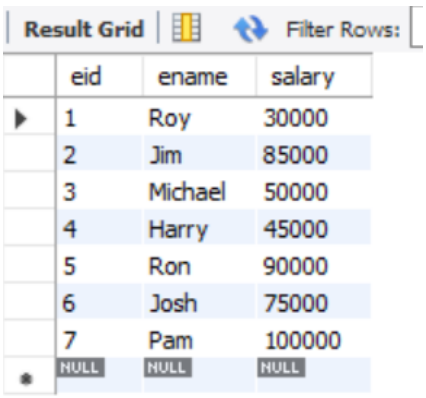
Output –



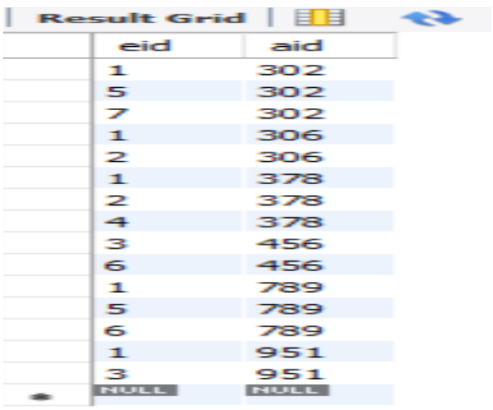
flno	from	to	distance	departs
101	LA	NY	4362	12:00:00
201	ATL	ORD	975	15:20:00
301	LAX	JFK	4000	13:45:00
401	ORD	MIA	1930	08:15:00
501	DEN	CLT	2146	22:50:00
601	ATL	CLT	363	18:18:00
701	ORD	LAX	2800	06:32:00
801	DEL	FRA	18000	07:15:00
802	BLR	FRA	19500	10:00:00
803	BLR	FRA	17000	12:00:00
NULL	NULL	NULL	NULL	NULL



aid	aname	range
123	Airbus	1000
302	Boeing	5000
306	Jet01	5000
378	Airbus380	8000
456	Aircraft	500
789	Aircraft02	800
951	Aircraft03	1000
NULL	NULL	NULL



eid	ename	salary
1	Roy	30000
2	Jim	85000
3	Michael	50000
4	Harry	45000
5	Ron	90000
6	Josh	75000
7	Pam	100000
NULL	NULL	NULL



eid	aid
1	302
5	302
7	302
1	306
2	306
1	378
2	378
4	378
3	456
6	456
1	789
5	789
6	789
1	951
3	951
NULL	NULL

Experiment – 4

Aim –

For the database created in ques 3, perform the following queries:

a. Find eid's of pilots who are certified on some Boeing.

b. Find names of pilots who are certified on some Boeing.

c. Find aid's of aircraft that can fly non-stop from LA to NY. Assume you don't already know the distance.

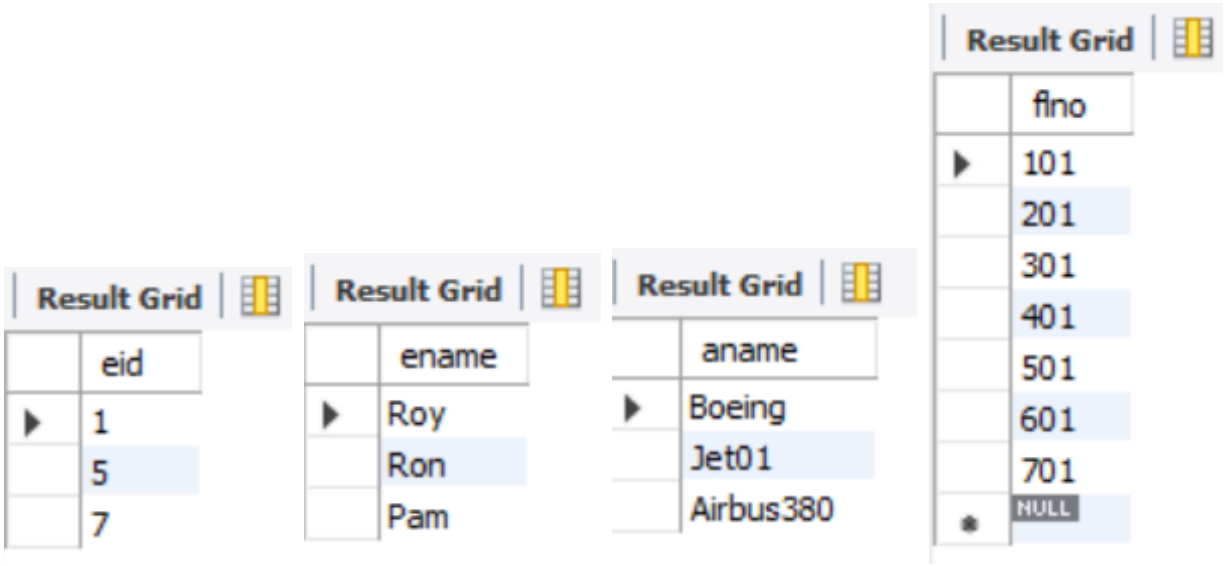
d. Find flno of flights that can be piloted by every pilot whose salary is over $100,000.

Software Used – MySQL Workbench

Code –

```
-- a)

SELECT  eid FROM Certified, Aircraft WHERE aname = 'Boeing' AND Certified.aid = Aircraft.aid;

-- b)

SELECT ename FROM Employees WHERE eid IN

(SELECT  eid FROM Certified, Aircraft WHERE aname = 'Boeing' AND Certified.aid = Aircraft.aid);

-- c)

SELECT aname FROM Aircraft WHERE

`range` >= (SELECT  distance FROM Flights WHERE `from` = 'LA' AND `to` = 'NY');

-- d)

SELECT flno FROM Flights WHERE distance <= (SELECT  `range` FROM Aircraft, Certified WHERE

Aircraft.aid = Certified.aid AND Certified.eid IN (SELECT eid FROM Employees WHERE salary >= 100000));
```

Output –



Result Grid

eid
1
5
7

Result Grid

ename
Roy
Ron
Pam

Result Grid

aname
Boeing
Jet01
Airbus380

Result Grid

fino
101
201
301
401
501
601
701
NULL

Experiment – 5

Aim –

For the database created in ques 3, perform the following queries:

a. Find names of pilots who can operate planes with a range greater than 3,000 miles, but are not certified on any Boeing.

b. Find eid of employee(s) with the highest salary.

c. Find eid of employee(s) with the second highest salary.

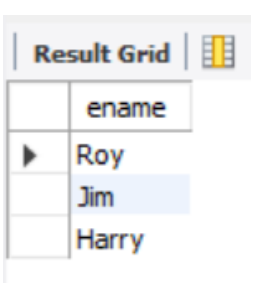
d. Find eid's of employees certified on exactly three aircraft.

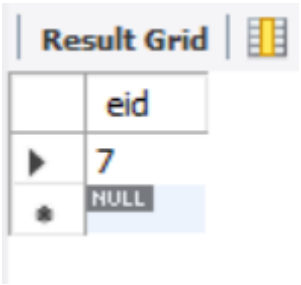
Software Used – MySQL Workbench

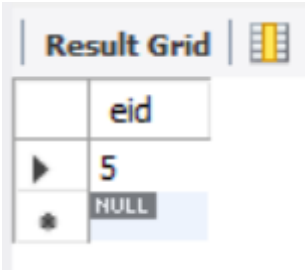
Code –

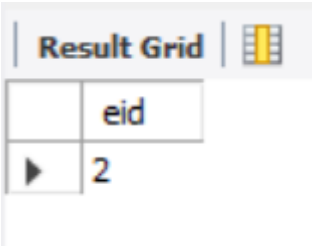
```
-- a)

SELECT  ename FROM Employees WHERE eid IN (SELECT eid FROM Certified, Aircraft WHERE

`range` > 3000 AND aname != 'Boeing' AND Certified.aid = Aircraft.aid);

-- b)

SELECT eid FROM Employees WHERE salary = (SELECT MAX(salary) FROM Employees);

-- c)

SELECT eid FROM Employees WHERE salary = (SELECT MAX(salary) FROM Employees

WHERE salary != (SELECT MAX(salary) FROM Employees));

-- d)

SELECT eid FROM (SELECT  eid, COUNT(eid) AS `count` FROM Certified GROUP BY eid) AS CountDetail
WHERE `count` = 3;
```

Output –









Experiment – 6

Aim –

Write SQL queries to create the following database:

Author (authorname, citizenship, birthyear)
Book(isbn, title, authorname)
Topic(isbn, subject)
Branch(libname, city)
Instock(isbn, libname, quantity)

Software Used – MySQL Workbench

Code –

```
CREATE TABLE Author (
   authorname VARCHAR(20) PRIMARY KEY,
   citizenship VARCHAR(20),
   birthyear VARCHAR(20)
);

CREATE TABLE Topic (
   isbn VARCHAR(20) PRIMARY KEY,
   `subject` VARCHAR(20)
);

CREATE TABLE Book (
   isbn VARCHAR(20) REFERENCES topic,
   title VARCHAR(50),
   authorname VARCHAR(20) REFERENCES author,
   PRIMARY KEY (title)
```

```
);

CREATE TABLE Branch (
   libname VARCHAR(20) PRIMARY KEY,
   city VARCHAR(20)
);

CREATE TABLE Instock (
   isbn VARCHAR(20) REFERENCES book,
   libname VARCHAR(20) REFERENCES branch,
   quantity INTEGER,
   PRIMARY KEY (isbn , libname)
);

INSERT INTO Author VALUES
('Akshat', 'USA', '1956'),
('Amrutanshu', 'Nepal', '1930'),
('Ansh', 'India', '2000'),
('Eshaan', 'Pakistan', '1995'),
('Harsh', 'Australia', '1919'),
('Laura McKowen', 'Italian', '1929'),
('Nelson Algren', 'Italian', '1948');

INSERT INTO Topic VALUES
('11', 'Ecology'),
```

('12', 'Physics'),

('13', 'Medicine'),

('14', 'Maths'),

('15', 'Alcohol'),

('16', 'Drugs');

INSERT INTO Book VALUES

('12', 'HC Verma', 'Harsh'),

('T1', 'Environmental Status', 'Akshat'),

('13', 'Human Medics', 'Eshaan'),

('13', 'Medical Science', 'Amrutanshu'),

('14', 'RD Sharma', 'Ansh'),

('15', 'The Surprising Magic of a Sober Life', 'Laura McKowen'),

('16', 'The Man with the Golden Arm is a novel', 'Nelson Algren'),

('71', 'Push Off From Here: Nine Essential Truths to Get You Through Life (and Everything Else)', 'Laura McKowen');

INSERT INTO branch VALUES

('Fisher', 'Sydney'),

('Laxmi', 'New Delhi'),

('Central Lib', 'London'),

('Book Home', 'Sydney'),

('National Lib', 'New York'),

('Main Lib', 'Melbourne');

INSERT INTO Instock VALUES

('71', 'Book Home',5),

('T1', 'Central Lib',15),

('12', 'Fisher',10),

('13', 'Laxmi',7),

('13', 'Fisher',11),

('13', 'National Lib', 14),

('14', 'National Lib',17),

('14', 'Fisher',8),

('14', 'Laxmi',8),

('16', 'National Lib',15),

('15', 'Book Home',9),

('16', 'Central Lib',5),

('12', 'Main Lib',12),

('71', 'Main Lib',2);

SELECT * FROM Author;

SELECT * FROM topic;

SELECT * FROM Book;

SELECT * FROM Branch;

SELECT * FROM Instock;

Output –

Result Grid | Filter Rows:

authorname	citizenship	birthyear
Akshat	USA	1956
Amrutanshu	Nepal	1930
Ansh	India	2000
Eshaan	Pakistan	1995
Harsh	Australia	1919
Laura McKowen	Italian	1929
Nelson Algren	Italian	1948
NULL	NULL	NULL

Result Grid | Filter Rows:

isbn	subject
11	Ecology
12	Physics
13	Medicine
14	Maths
15	Alcohol
16	Drugs
71	Autobiography
NULL	NULL

Result Grid | Filter Rows: | Edit:

isbn	title	authorname
T1	Environmental Status	Akshat
12	HC Verma	Harsh
13	Human Medics	Eshaan
13	Medical Science	Amrutanshu
71	Push Off From Here	Laura McKowen
14	RD Sharma	Ansh
16	The Man with the Golden Arm is a novel	Nelson Algren
15	The Surprising Magic of a Sober Life	Laura McKowen
NULL	NULL	NULL

Result Grid | Filter Rows:

libname	city
Book Home	Sydney
Central Lib	London
Fisher	Sydney
Laxmi	New Delhi
Main Lib	Melbourne
National Lib	New York
NULL	NULL

Result Grid | Filter Rows:

isbn	libname	quantity
12	Fisher	10
12	Main Lib	12
13	Fisher	11
13	Laxmi	7
13	National Lib	14
14	Fisher	8
14	Laxmi	8
14	National Lib	17
15	Book Home	9
16	Central Lib	5
16	National Lib	15
71	Book Home	5
71	Main Lib	2
T1	Central Lib	15
NULL	NULL	NULL

Experiment – 7

Aim –

For the database created in ques 6, perform the following queries:

a. Give all authors born after 1940.
b. Give the names of libraries in Sydney.
c. Give the cities where each book is held.
d. Give the title of each book on the topic of either alcohol or drugs

Software Used – MySQL Workbench

Code –

```
-- a)

SELECT authorname FROM Author WHERE birthyear > 1940;

-- b)

SELECT libname FROM Branch WHERE city = 'Sydney';

-- c)

SELECT  Instock.isbn, title, city FROM Branch, Instock, Book

WHERE Branch.libname = Instock.libname AND Book.isbn = Instock.isbn;

-- d)

SELECT title FROM Book,Topic WHERE Book.isbn = Topic.isbn AND `subject` IN ('Alcohol' , 'Drugs');
```

Output –

Result Grid

authorname
Akshat
Ansh
Eshaan
Nelson Algren
NULL

Result Grid

libname
Book Home
Fisher
NULL

Result Grid | Filter Rows:

isbn	title	city
T1	Environmental Status	London
12	HC Verma	Sydney
12	HC Verma	Melbourne
13	Human Medics	Sydney
13	Human Medics	New Delhi
13	Human Medics	New York
13	Medical Science	Sydney
13	Medical Science	New Delhi
13	Medical Science	New York
71	Push Off From Here	Sydney
71	Push Off From Here	Melbourne
14	RD Sharma	Sydney
14	RD Sharma	New Delhi
14	RD Sharma	New York
16	The Man with the Go...	London
16	The Man with the Go...	New York
15	The Surprising Magic...	Sydney

Result Grid | Filter Rows:

title
The Man with the Golden Arm is a novel
The Surprising Magic of a Sober Life

Experiment – 8

Aim –

For the database created in ques 6, perform the following queries:

a. Give the title and author of each book of which at least two copies are held in a branch located in Melbourne.

b. Give the name of each Italian author who wrote an autobiography.

c. Give the total number of books in stock in the branch called Fisher.

d. This could mean the number of different titles, or the number of physical copies.

e. Give the total number of books in stock in the branches located in Sydney.

Software Used – MySQL Workbench

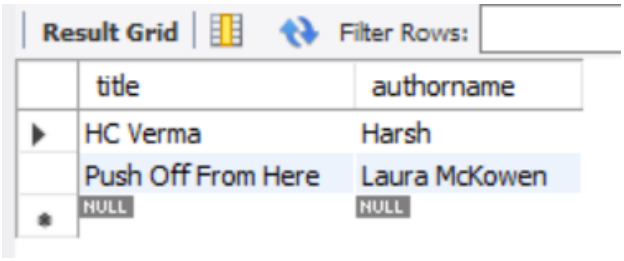
Code –

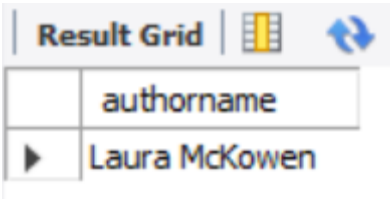
```
-- a)

SELECT  title, authorname FROM Book WHERE isbn IN

(SELECT isbn FROM Branch, Instock WHERE city = 'Melbourne' AND quantity >= 2 AND Branch.libname = Instock.libname);

-- b)

SELECT Book.authorname FROM Book, Topic, Author

WHERE Book.isbn = Topic.isbn AND `subject` = 'Autobiography' AND Author.authorname = Book.authorname  AND citizenship = 'Italian';

-- c)

SELECT  libname AS Branch, SUM(quantity) AS Stock FROM Instock WHERE libname = 'Fisher' GROUP BY libname;

-- d) & -- e)

SELECT  title, SUM(Quantity.quantity) AS Stock FROM

(SELECT isbn, quantity FROM Branch, Instock WHERE city = 'Sydney' AND Branch.libname = Instock.libname) AS Quantity,Book WHERE Quantity.isbn = Book.isbn GROUP BY Quantity.isbn;
```

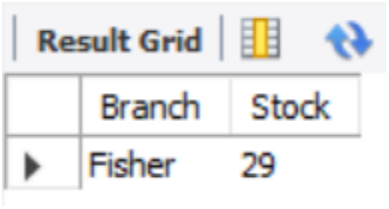
SELECT SUM(quantity) AS TotalSydneyStock FROM Branch, Instock WHERE city = 'Sydney' AND Branch.libname = Instock.libname;

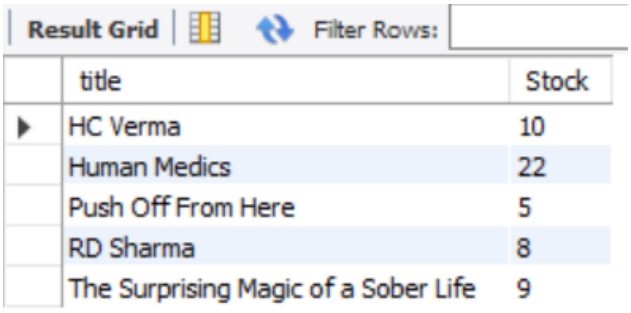
SELECT Branch.libname, SUM(quantity) AS Stock FROM Branch, Instock WHERE city = 'Sydney' AND Branch.libname = Instock.libname GROUP BY Branch.libname;

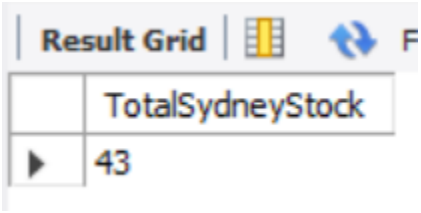
Output –

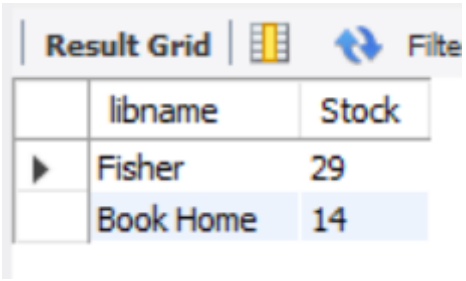












Experiment – 9

Aim –

Write SQL queries to create the following database:

Suppliers(*sid: integer,sname:varchar(10),address: varchar(10),)*

Parts(*pid: integer, pname: varchar(10), color: varchar(10))*

Catalog(*sid: integer, pid: integer, cost: real)*

Software Used – MySQL Workbench

Code –

```
CREATE TABLE Suppliers (
   sid INTEGER PRIMARY KEY,
   sname VARCHAR(30),
   address VARCHAR(50)
);


CREATE TABLE Parts (
   pid INTEGER PRIMARY KEY,
   pname VARCHAR(40),
   color VARCHAR(20)
);


CREATE TABLE Catalog (
   sid INTEGER REFERENCES suppliers,
   pid INTEGER REFERENCES parts,
   cost REAL
);
```

```
INSERT INTO Parts VALUES
(1, 'Left Handed Bacon Stretcher Cover', 'Red'),
(2, 'Smoke Shifter End', 'Black'),
(3, 'Acme Widget Washer', 'Red'),
(4, 'Acme Widget Washer', 'Blue'),
(5, 'I Brake for Crop Circles Sticker', 'Black'),
(6, 'Anti-Gravity Turbine Generator', 'Blue'),
(7, 'Anti-Gravity Turbine Generator', 'Red'),
(8, 'Fire Hydrant Cap', 'Red'),
(9, '7 Segment Display', 'Green');

INSERT INTO Suppliers VALUES
(1, 'Acme Widget Suppliers', '1 Grub St., Potemkin Village, IL 61801'),
(2, 'Big Red Tool and Die', '4 My Way, Bermuda Shorts, OR 90305'),
(3, 'Perfunctory Parts', '99999 Short Pier, Terra Del Fuego, TX 41299'),
(4, 'Alien Aircaft Inc.', '2 Groom Lake, Rachel, NV 51902'),
(5,'King', 'S487/04 street, Melbourne');

INSERT INTO Catalog VALUES
(1, 3, 0.50),
(1, 4, 0.50),
(1, 8, 11.70),
(2, 3, 0.55),
(2, 8, 7.95),
```

(2, 1, 16.50),

(3, 8, 12.50),

(3, 9, 1.00),

(4, 5, 2.20),

(4, 6, 1247548.23),

(4, 7, 1247548.23),

(5, 7, 15.89),

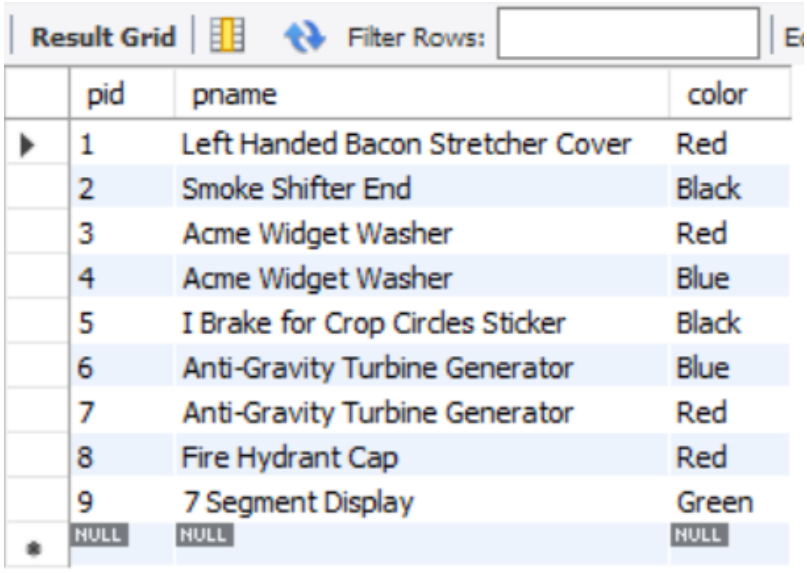
(5, 3, 1.89);

SELECT * FROM Parts;

SELECT * FROM Suppliers;

SELECT * FROM Catalog;

Output –



	pid	pname	color
▶	1	Left Handed Bacon Stretcher Cover	Red
	2	Smoke Shifter End	Black
	3	Acme Widget Washer	Red
	4	Acme Widget Washer	Blue
	5	I Brake for Crop Circles Sticker	Black
	6	Anti-Gravity Turbine Generator	Blue
	7	Anti-Gravity Turbine Generator	Red
	8	Fire Hydrant Cap	Red
	9	7 Segment Display	Green
*	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Expo

sid	sname	address
1	Acme Widget Suppliers	1 Grub St., Potemkin Village, IL 61801
2	Big Red Tool and Die	4 My Way, Bermuda Shorts, OR 90305
3	Perfunctory Parts	99999 Short Pier, Terra Del Fuego, TX 41299
4	Alien Aircaft Inc.	2 Groom Lake, Rachel, NV 51902
5	King	S487/04 street, Melbourne
NULL	NULL	NULL

Result Grid | Filter Rows:

sid	pid	cost
1	3	0.5
1	4	0.5
1	8	11.7
2	3	0.55
2	8	7.95
2	1	16.5
3	8	12.5
3	9	1
4	5	2.2
4	6	1247548.23
4	7	1247548.23
5	7	15.89
5	3	1.89

Experiment – 10

Aim –

For the database created in ques 9, perform the following queries:

a. After creating the table I realized that I forgot to mention that only green, red, blue and black color parts are allowed in parts relation. Write a query to add this constraint in parts relation.

b. For every supplier that only supplies green parts, print the name of the supplier and the total number of parts that she supplies.

c. Increase the cost of each product by 10% which are green in color and are supplied by supplier = “John”.

d. Authorize a user “Mary” to update color in “Parts” table. She can also pass the authorization to other users.

Software Used – MySQL Workbench

Code –

```
-- a)

ALTER TABLE Parts ADD CONSTRAINT ColorConstraint CHECK(color IN ("Green", "Red", "Blue", "Black"));

-- b)

SELECT sid, count(pid) as NumberOfPartsSelling  FROM Catalog WHERE

sid IN (SELECT DISTINCT T1.sid FROM Catalog AS T1 WHERE NOT EXISTS

( (SELECT pid FROM Parts WHERE color = 'Green' AND pid NOT IN

((SELECT T2.pid FROM Catalog AS T2 WHERE T1.sid = T2.sid)))));

-- c)

UPDATE Catalog  SET cost = cost + 0.1 * cost WHERE

 sid IN (SELECT  sid FROM Suppliers WHERE  sname = 'John') AND pid IN (SELECT  pid FROM Parts WHERE
color = 'Green');

-- d)

CREATE USER 'Mary'@'localhost' IDENTIFIED BY 'root123';

 GRANT UPDATE (COLOR) ON Parts TO 'Mary'@'localhost';
```

Output –

```
175  22:47:52  alter table Parts add constraint ColorConstraint Check(color in ("Green", "Red", "Bl...  9 row(s) affected Records: 9  Duplicates: 0  Warnings: 0  0.078 sec
```

Result Grid | Filter Rows:

	sid	NumberOfPartsSelling
▶	3	2

```
180  00:16:59  update Catalog set cost = cost + 0.1 * cost where sid in (select sid from Suppliers w...  0 row(s) affected Rows matched: 0  Changed: 0  Warnings: 0  0.000 sec
```

Result Grid | Filter Rows:

	sid	pid	cost
▶	1	3	0.5
	1	4	0.5
	1	8	11.7
	2	3	0.55
	2	8	7.95
	2	1	16.5
	3	8	12.5
	3	9	1
	4	5	2.2
	4	6	1247548.23
	4	7	1247548.23
	5	7	15.89
	5	3	1.89

```
183  00:20:40  CREATE USER 'Mary'@'localhost' IDENTIFIED BY 'root123'  0 row(s) affected  0.047 sec
184  00:20:43  grant update (color) on Parts to 'Mary'@'localhost'  0 row(s) affected  0.015 sec
```

Experiment – 11

Aim –

For the database created in ques 9, perform the following queries:

a. Delete all the parts supplied by supplier “King”.

b. Drop the attribute color from parts relation.

c. Create a view for customers so that they can see only the supplier names, the name of parts supplied by that supplier and their cost.

Software Used – MySQL Workbench

Code –

```
-- a)

DELETE FROM Parts WHERE pid IN (SELECT pid FROM Catalog, Suppliers  WHERE sname = 'King' AND Catalog.sid = Suppliers.sid);

SELECT *FROM Parts;

-- b)

ALTER TABLE Parts DROP COLUMN color;

SELECT  * FROM Parts;

-- c)

CREATE VIEW supplier_info AS

 (SELECT sname, pname, cost FROM

 (SELECT sname, cost, pid FROM Suppliers, Catalog WHERE Suppliers.sid = Catalog.sid) AS Supply, Parts WHERE Parts.pid = Supply.pid);

SELECT * FROM  supplier_info;
```

Output –

187 00:29:41 delete from parts where pid in (select pid from Catalog, Suppliers where sname = "ki... 2 row(s) affected 0.016 sec

pid	pname	color
1	Left Handed Bacon Stretcher Cover	Red
2	Smoke Shifter End	Black
4	Acme Widget Washer	Blue
5	I Brake for Crop Circles Sticker	Black
6	Anti-Gravity Turbine Generator	Blue
8	Fire Hydrant Cap	Red
9	7 Segment Display	Green
NULL	NULL	NULL

189 00:32:08 alter table Parts drop column color 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0 0.093 sec

Result Grid Filter Rows:

pid	pname
1	Left Handed Bacon Stretcher Cover
2	Smoke Shifter End
4	Acme Widget Washer
5	I Brake for Crop Circles Sticker
6	Anti-Gravity Turbine Generator
8	Fire Hydrant Cap
9	7 Segment Display
NULL	NULL

Result Grid Filter Rows: Export: Wrap Cell C

sname	pname	cost
Acme Widget Suppliers	Acme Widget Washer	0.5
Acme Widget Suppliers	Fire Hydrant Cap	11.7
Big Red Tool and Die	Fire Hydrant Cap	7.95
Big Red Tool and Die	Left Handed Bacon Stretcher Cover	16.5
Perfunctory Parts	Fire Hydrant Cap	12.5
Perfunctory Parts	7 Segment Display	1
Alien Aircaft Inc.	I Brake for Crop Circles Sticker	2.2
Alien Aircaft Inc.	Anti-Gravity Turbine Generator	1247548.23