

DBMS LAB RECORD

Name: Sakshi P Khandoba

USN: 1BM19CS139

PROGRAM 1 : INSURANCE DATABASE

Consider the Insurance database given below. The primary keys are underlined and the data types are specified.

PERSON (driver-id #: String, name: String, address: String)

CAR (regno: String, model: String, year: int)

ACCIDENT (report-number: int, date: date, location: String)

OWNS (driver-id #: String, Regno: String)

PARTICIPATED (driver-id: String, regno: String, report-number:int, damage-amount:int)

- 1) Create the above tables by properly specifying the primary keys and the foreign keys.

The screenshot shows the MySQL Workbench interface with the SQL tab selected. A query window contains the following SQL code:

```
1 CREATE TABLE PERSON(driver_id char(20), name char(20), address char(50), PRIMARY KEY(driver_id));
2 CREATE TABLE CAR(reg_no char(20), model char(20), year int, PRIMARY KEY(reg_no));
3 CREATE TABLE ACCIDENT(report_number int, date date, location char(50), PRIMARY KEY(report_number));
4 CREATE TABLE OWNS(driver_id char(20), reg_no char(20), FOREIGN KEY(driver_id) REFERENCES PERSON(driver_id) ON DELETE SET NULL ON UPDATE CASCADE, FOREIGN KEY(reg_no)
    REFERENCES CAR(reg_no) ON DELETE SET NULL ON UPDATE CASCADE);
5 CREATE TABLE PARTICIPATED(driver_id char(20), reg_no char(20), report_number int, damage_amount int, FOREIGN KEY(driver_id) REFERENCES PERSON(driver_id) ON DELETE SET NULL
ON UPDATE CASCADE, FOREIGN KEY(reg_no) REFERENCES CAR(reg_no) ON DELETE SET NULL ON UPDATE CASCADE, FOREIGN KEY(report_number) REFERENCES ACCIDENT(report_number) ON DELETE
SET NULL ON UPDATE CASCADE);
```

The screenshot shows the MySQL Workbench interface with the Structure tab selected. It displays the five tables created:

Table	Action	Rows	Type	Collation	Size	Overhead
accident		0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
car		0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
owns		0	InnoDB	utf8mb4_general_ci	48.0 KiB	-
participated		0	InnoDB	utf8mb4_general_ci	64.0 KiB	-
person		5	InnoDB	utf8mb4_general_ci	16.0 KiB	-
5 tables	Sum				160.0 KiB	0 B

2) Enter at least 5 tuples for each relation.

'PERSON' table:

Server: 127.0.0.1 » Database: insurance » Table: person

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Run SQL query/queries on table insurance.person:

```
1 INSERT INTO person(driverid, name, address) VALUES ('D102','Abhishek','Hebbal');
2 INSERT INTO person(driverid, name, address) VALUES ('D101','Sakshi','Basavangudi');
3 INSERT INTO person(driverid, name, address) VALUES ('D104','Taniya','Sahakarnagar');
4 INSERT INTO person(driverid, name, address) VALUES ('D103','Akshay','Yelahanka');
5 INSERT INTO person(driverid, name, address) VALUES ('D105','Shreya','JP Nagar');
```

Server: 127.0.0.1 » Database: insurance » Table: person

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Show query box

1 row inserted. (Query took 0.4008 seconds.)

INSERT INTO PERSON(driver_id, name, address) VALUES ('D102','Abhishek','Hebbal')

[Edit inline] [Edit] [Create PHP code]

1 row inserted. (Query took 0.0497 seconds.)

INSERT INTO PERSON(driver_id, name, address) VALUES ('D101','Sakshi','Basavangudi')

[Edit inline] [Edit] [Create PHP code]

1 row inserted. (Query took 0.0536 seconds.)

INSERT INTO PERSON(driver_id, name, address) VALUES ('D104','Taniya','Sahakarnagar')

[Edit inline] [Edit] [Create PHP code]

1 row inserted. (Query took 0.1037 seconds.)

INSERT INTO PERSON(driver_id, name, address) VALUES ('D103','Akshay','Yelahanka')

[Edit inline] [Edit] [Create PHP code]

1 row inserted. (Query took 0.0430 seconds.)

INSERT INTO PERSON(driver_id, name, address) VALUES ('D105','Shreya','JP Nagar')

[Edit inline] [Edit] [Create PHP code]

Server: 127.0.0.1 » Database: insurance » Table: person

Browse Structure SQL Search Insert Export Import

Showing rows 0 - 4 (5 total, Query took 0.0005 seconds.)

SELECT * FROM `person`

Show all | Number of rows: All ▾ Filter rows: Search this table Sort by

+ Options

	driver_id	name	address
<input type="checkbox"/>	Edit Copy Delete D101	Sakshi	Basavangudi
<input type="checkbox"/>	Edit Copy Delete D102	Abhishek	Hebbal
<input type="checkbox"/>	Edit Copy Delete D103	Akshay	Yelahanka
<input type="checkbox"/>	Edit Copy Delete D104	Taniya	Sahakarnagar
<input type="checkbox"/>	Edit Copy Delete D105	Shreya	JP Nagar

'CAR' table:

Server: 127.0.0.1 » Database: insurance » Table: car

Browse Structure SQL Search Insert Export Import Privileges

Run SQL query/queries on table insurance.car: [?](#)

```
1 INSERT INTO CAR(reg_no, model, year) VALUES ('R201','Baleno',2012);
2 INSERT INTO CAR(reg_no, model, year) VALUES ('R202','Swift',2010);
3 INSERT INTO CAR(reg_no, model, year) VALUES ('R203','Maruti',2011);
4 INSERT INTO CAR(reg_no, model, year) VALUES ('R204','Honda',2008);
5 INSERT INTO CAR(reg_no, model, year) VALUES ('R205','Hyundai',2009);
```

Server: 127.0.0.1 » Database: insurance » Table: car

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Show query box

✓ 1 row inserted. (Query took 0.1359 seconds.)

```
INSERT INTO CAR(reg_no, model, year) VALUES ('R201','Baleno',2012)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0294 seconds.)

```
INSERT INTO CAR(reg_no, model, year) VALUES ('R202','Swift',2010)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0708 seconds.)

```
INSERT INTO CAR(reg_no, model, year) VALUES ('R203','Maruti',2011)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0316 seconds.)

```
INSERT INTO CAR(reg_no, model, year) VALUES ('R204','Honda',2008)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0325 seconds.)

```
INSERT INTO CAR(reg_no, model, year) VALUES ('R205','Hyundai',2009)
```

[Edit inline] [Edit] [Create PHP code]

Server: 127.0.0.1 » Database: insurance » Table: car

Browse Structure SQL Search Insert Export Import Privileges

✓ Showing rows 0 - 4 (5 total, Query took 0.0008 seconds.)

```
SELECT * FROM `car`
```

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

+ Options

	reg_no	model	year
<input type="checkbox"/>	R201	Baleno	2012
<input type="checkbox"/>	R202	Swift	2010
<input type="checkbox"/>	R203	Maruti	2011
<input type="checkbox"/>	R204	Honda	2008
<input type="checkbox"/>	R205	Hyundai	2009

'ACCIDENT' table:

Server: 127.0.0.1 » Database: insurance » Table: accident

Browse Structure SQL Search Insert Export Import Privileges Operations

Run SQL query/queries on table insurance.accident:

```
1 INSERT INTO ACCIDENT(report_number, date, location) VALUES (1001,'2020-08-14','MG Road');
2 INSERT INTO ACCIDENT(report_number, date, location) VALUES (1020,'2021-04-13','Church Street');
3 INSERT INTO ACCIDENT(report_number, date, location) VALUES (1031,'2021-04-01','Yelahanka');
4 INSERT INTO ACCIDENT(report_number, date, location) VALUES (1016,'2019-12-13','Whitefield');
5 INSERT INTO ACCIDENT(report_number, date, location) VALUES (1045,'2020-05-21','Basavanagudi');
```

Server: 127.0.0.1 » Database: insurance » Table: accident

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Show query box

✓ 1 row inserted. (Query took 0.2212 seconds.)

```
INSERT INTO ACCIDENT(report_number, date, location) VALUES (1001,'2020-08-14','MG Road')
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0863 seconds.)

```
INSERT INTO ACCIDENT(report_number, date, location) VALUES (1020,'2021-04-13','Church Street')
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.1024 seconds.)

```
INSERT INTO ACCIDENT(report_number, date, location) VALUES (1031,'2021-04-01','Yelahanka')
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0406 seconds.)

```
INSERT INTO ACCIDENT(report_number, date, location) VALUES (1016,'2019-12-13','Whitefield')
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0421 seconds.)

```
INSERT INTO ACCIDENT(report_number, date, location) VALUES (1045,'2020-05-21','Basavanagudi')
```

[Edit inline] [Edit] [Create PHP code]

Server: 127.0.0.1 » Database: insurance » Table: accident

Browse Structure SQL Search Insert Export Import Privileges Operations

Showing rows 0 - 4 (5 total, Query took 0.0007 seconds.)

```
SELECT * FROM `accident`
```

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

+ Options

	report_number	date	location
<input type="checkbox"/>	1001	2020-08-14	MG Road
<input type="checkbox"/>	1016	2019-12-13	Whitefield
<input type="checkbox"/>	1020	2021-04-13	Church Street
<input type="checkbox"/>	1031	2021-04-01	Yelahanka
<input type="checkbox"/>	1045	2020-05-21	Basavanagudi

'OWNS' table:

Server: 127.0.0.1 » Database: insurance » Table: owns

Browse Structure SQL Search Insert Export Import Privileges

Run SQL query/queries on table insurance.owns: [?](#)

```
1 INSERT INTO OWNS(driver_id, reg_no) VALUES ('D102','R201');
2 INSERT INTO OWNS(driver_id, reg_no) VALUES ('D103','R204');
3 INSERT INTO OWNS(driver_id, reg_no) VALUES ('D105','R202');
4
```

Server: 127.0.0.1 » Database: insurance » Table: owns

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Show query box

✓ 1 row inserted. (Query took 0.1512 seconds.)

INSERT INTO OWNS(driver_id, reg_no) VALUES ('D102','R201')

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0264 seconds.)

INSERT INTO OWNS(driver_id, reg_no) VALUES ('D103','R204')

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0684 seconds.)

INSERT INTO OWNS(driver_id, reg_no) VALUES ('D105','R202')

[Edit inline] [Edit] [Create PHP code]

Server: 127.0.0.1 » Database: insurance » Table: owns

Browse Structure SQL Search Insert Export

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy a

✓ Showing rows 0 - 2 (3 total, Query took 0.0007 seconds.)

SELECT * FROM `owns`

Show all | Number of rows: 25 [▼](#) Filter rows:

+ Options

driver_id	reg_no
D102	R201
D103	R204
D105	R202

'PARTICIPATED' table:

Server: 127.0.0.1 » Database: insurance » Table: participated

Browse Structure SQL Search Insert Export Import Privileges Operations

Run SQL query/queries on table insurance.participated: [?](#)

```

1 INSERT INTO PARTICIPATED(driver_id, reg_no, report_number, damage_amount) VALUES ('D102','R201',1020,1500);
2 INSERT INTO PARTICIPATED(driver_id, reg_no, report_number, damage_amount) VALUES ('D103','R204',1031,2000);
3 INSERT INTO PARTICIPATED(driver_id, reg_no, report_number, damage_amount) VALUES ('D104','R205',1045,3000) ;
4

```

Server: 127.0.0.1 » Database: insurance » Table: participated

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Show query box

✓ 1 row inserted. (Query took 0.1562 seconds.)

```
INSERT INTO PARTICIPATED(driver_id, reg_no, report_number, damage_amount) VALUES ('D102','R201',1020,1500)
```

[\[Edit inline\]](#) [\[Edit\]](#) [\[Create PHP code\]](#)

✓ 1 row inserted. (Query took 0.0278 seconds.)

```
INSERT INTO PARTICIPATED(driver_id, reg_no, report_number, damage_amount) VALUES ('D103','R204',1031,2000)
```

[\[Edit inline\]](#) [\[Edit\]](#) [\[Create PHP code\]](#)

✓ 1 row inserted. (Query took 0.1018 seconds.)

```
INSERT INTO PARTICIPATED(driver_id, reg_no, report_number, damage_amount) VALUES ('D104','R205',1045,3000)
```

[\[Edit inline\]](#) [\[Edit\]](#) [\[Create PHP code\]](#)

Server: 127.0.0.1 » Database: insurance » Table: participated

Browse Structure SQL Search Insert Export Import Privileges

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 2 (3 total, Query took 0.0006 seconds.)

```
SELECT * FROM `participated`
```

Show all | Number of rows: 25 Filter rows: Sort by key:

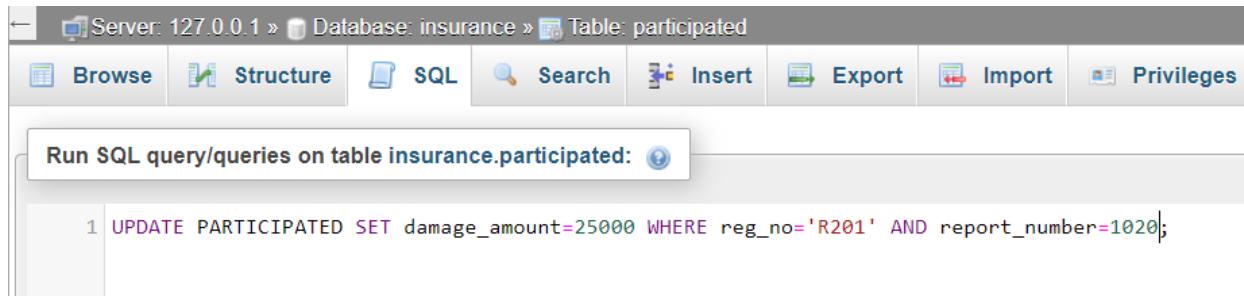
+ Options

driver_id	reg_no	report_number	damage_amount
D102	R201	1020	1500
D103	R204	1031	2000
D104	R205	1045	3000

3) Demonstrate how you

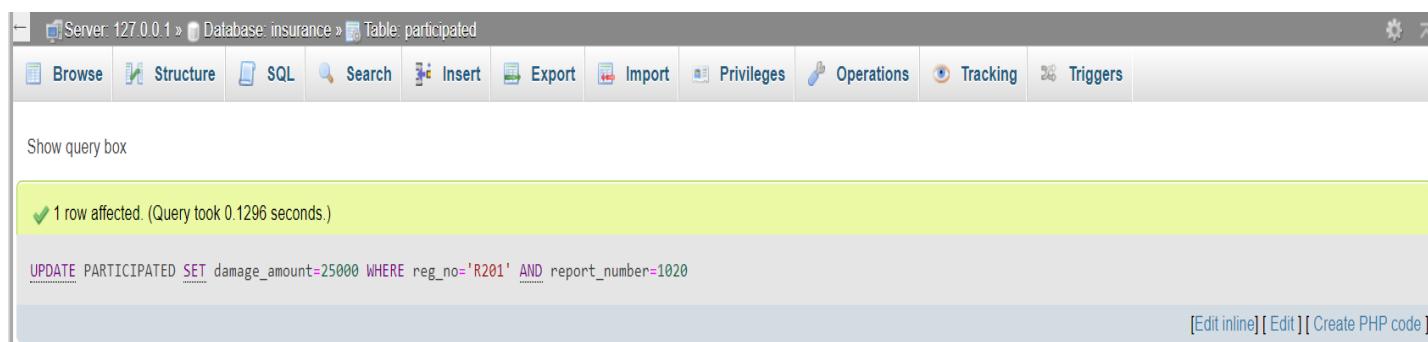
- a) Update the damage amount for the car with a specific Regno in the accident with report number 12 to 25000.

Query: UPDATE PARTICIPATED SET damage_amount=25000 WHERE reg_no='R201' AND report_number=1020;



The screenshot shows the phpMyAdmin interface for the 'participated' table in the 'insurance' database. The SQL tab is selected. A query is entered in the text area:

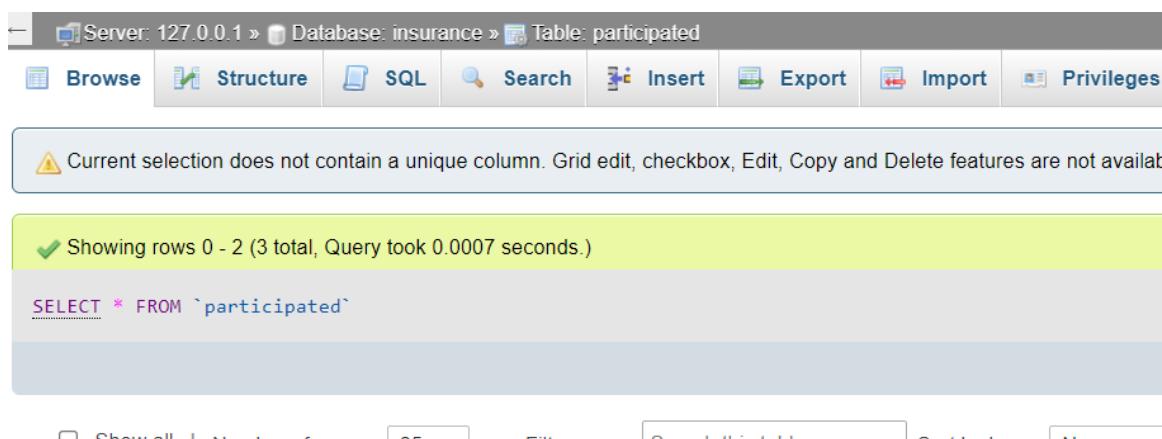
```
1 UPDATE PARTICIPATED SET damage_amount=25000 WHERE reg_no='R201' AND report_number=1020;
```



The screenshot shows the phpMyAdmin interface after executing the update query. The status bar at the top indicates "1 row affected. (Query took 0.1296 seconds.)". The SQL tab shows the executed query:

```
UPDATE PARTICIPATED SET damage_amount=25000 WHERE reg_no='R201' AND report_number=1020
```

At the bottom right, there are links: [Edit inline] [Edit] [Create PHP code].



The screenshot shows the phpMyAdmin interface after executing a select query. A warning message in a yellow box states: "⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available". The status bar at the top indicates "Showing rows 0 - 2 (3 total, Query took 0.0007 seconds.)". The SQL tab shows the executed query:

```
SELECT * FROM `participated`
```

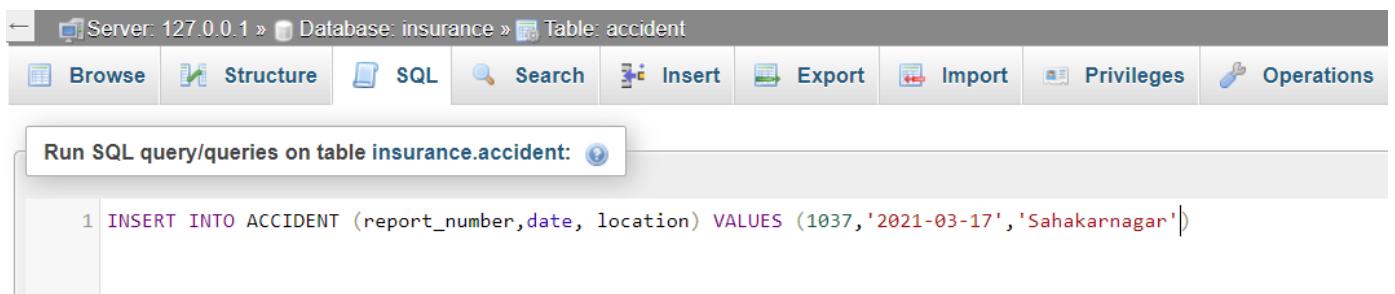
Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

driver_id	reg_no	report_number	damage_amount
D102	R201	1020	25000
D103	R204	1031	2000
D104	R205	1045	3000

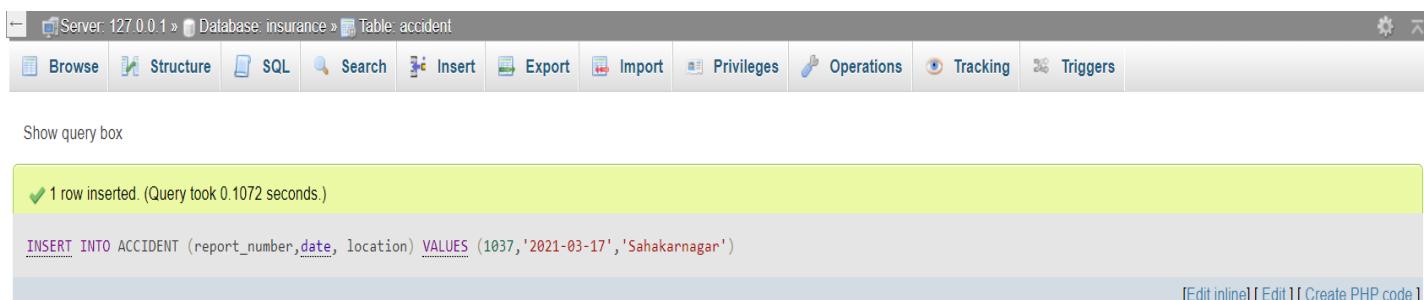
b) Add a new accident to the database.

Query: INSERT INTO ACCIDENT (report_number, date, location) VALUES (1037, '2021-03-17', 'Sahakarnagar');

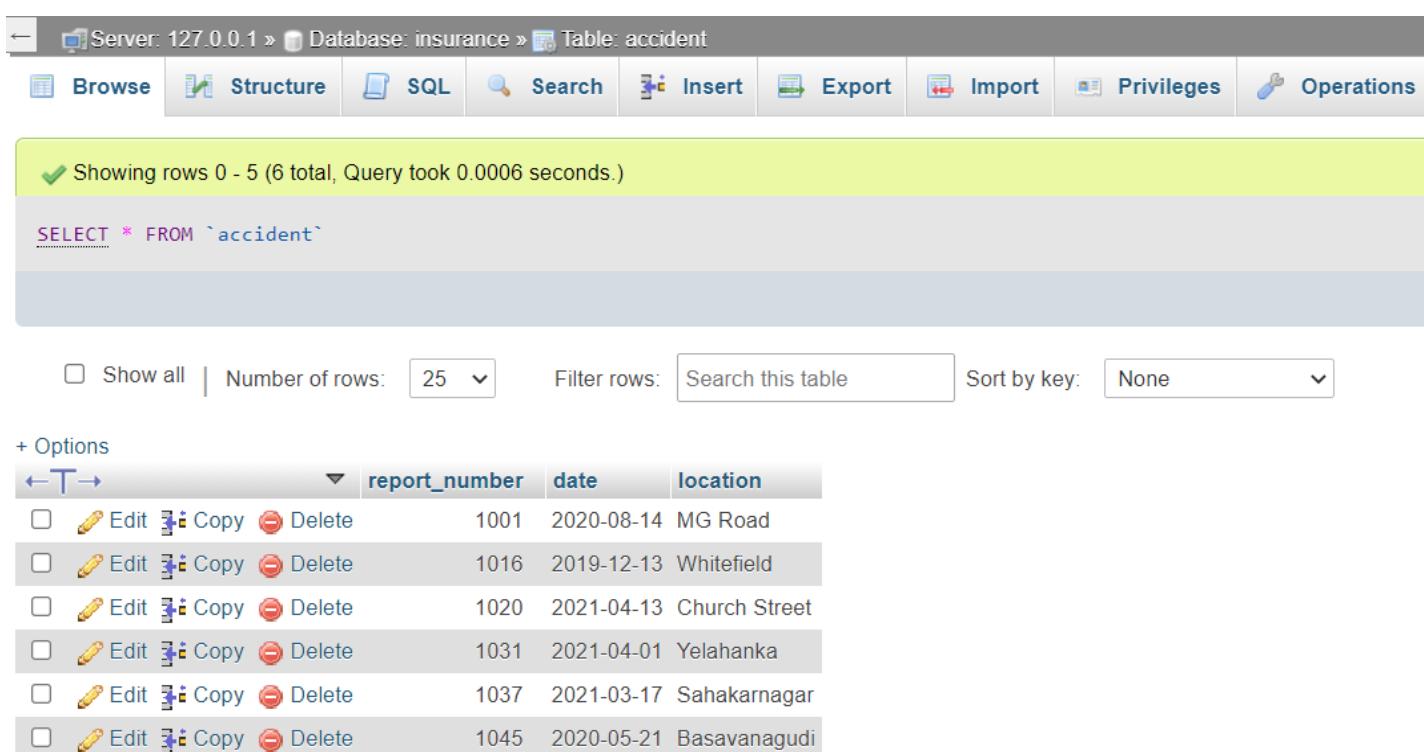


The screenshot shows the phpMyAdmin interface for a database named 'insurance'. The 'Table: accident' is selected. The top navigation bar includes 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', and 'Operations'. Below the navigation is a search bar for 'Run SQL query/queries on table insurance.accident:'. The SQL query entered is:

```
1 INSERT INTO ACCIDENT (report_number, date, location) VALUES (1037, '2021-03-17', 'Sahakarnagar')
```



The screenshot shows the results of the executed query. A green success message at the top states '1 row inserted. (Query took 0.1072 seconds.)'. Below it, the query is repeated: 'INSERT INTO ACCIDENT (report_number, date, location) VALUES (1037, '2021-03-17', 'Sahakarnagar')'. At the bottom right, there are links for '[Edit inline]', '[Edit]', and '[Create PHP code]'. The status bar at the bottom says 'Show query box'.



The screenshot shows the 'Browse' tab for the 'accident' table. The top navigation bar includes 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', and 'Operations'. A message at the top says 'Showing rows 0 - 5 (6 total, Query took 0.0006 seconds.)'. Below is a list of accidents with columns: report_number, date, and location. The row for report number 1037 is highlighted. At the bottom, there are filters for 'Show all' (checkbox), 'Number of rows' (dropdown set to 25), 'Filter rows' (text input 'Search this table'), 'Sort by key' (dropdown set to 'None'), and a 'None' dropdown. The table data is as follows:

	report_number	date	location
<input type="checkbox"/>	1001	2020-08-14	MG Road
<input type="checkbox"/>	1016	2019-12-13	Whitefield
<input type="checkbox"/>	1020	2021-04-13	Church Street
<input type="checkbox"/>	1031	2021-04-01	Yelahanka
<input type="checkbox"/>	1037	2021-03-17	Sahakarnagar
<input type="checkbox"/>	1045	2020-05-21	Basavanagudi

- 4) Find the total number of people who owned cars that were involved in accidents in 2021.

Query: `SELECT COUNT(DISTINCT driver_id) CNT FROM PARTICIPATED A, ACCIDENT B WHERE A.report_number=B.report_number AND B.date LIKE '2021%';`

Server: 127.0.0.1 » Database: insurance » Table: accident

Browse Structure SQL Search Insert Export Import Privileges

Run SQL query/queries on table insurance.accident:

```
1 SELECT COUNT(DISTINCT driver_id) CNT FROM PARTICIPATED A, ACCIDENT B WHERE
2 A.report_number=B.report_number AND B.date LIKE '2021%';
```

Server: 127.0.0.1 » Database: insurance » Table: accident

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking

Show query box

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Your SQL query has been executed successfully.

```
SELECT COUNT(DISTINCT driver_id) CNT FROM PARTICIPATED A, ACCIDENT B WHERE A.report_number=B.report_number AND B.date LIKE '2021%'
```

+ Options Profiling

CNT
2

5) Find the number of accidents in which cars belonging to a specific model (say 'Hyundai') were involved.

Query: `SELECT COUNT(report_number) CNT FROM PARTICIPATED A, CAR B WHERE A.reg_no=B.reg_no AND model='Hyundai';`

The screenshot shows the MySQL Workbench interface. The title bar indicates the connection is to 'Server: 127.0.0.1 » Database: insurance » Table: car'. The toolbar has tabs for Browse, Structure, SQL, Search, Insert, Export, Import, and Privileges. Below the toolbar is a search bar with the placeholder 'Run SQL query/queries on table insurance.car:'. The main area contains the following SQL code:

```
1 SELECT COUNT(report_number) CNT FROM PARTICIPATED A, CAR B WHERE A.reg_no=B.reg_no AND
2 model='Hyundai';
```

The screenshot shows the MySQL Workbench interface after executing the SQL query. The title bar and toolbar are identical to the previous screenshot. The main area displays a message: 'Your SQL query has been executed successfully.' Below this, the executed query is shown again: `SELECT COUNT(report_number) CNT FROM PARTICIPATED A, CAR B WHERE A.reg_no=B.reg_no AND model='Hyundai'`. At the bottom left, there is a '+ Options' button and a result table with one row. The table has a single column 'CNT' with the value '1'.

PROGRAM 2 : BANKING ENTERPRISE DATABASE

Consider the following database for a banking enterprise.

BRANCH (branch-name: String, branch-city: String, assets: real)

ACCOUNTS (accno: int, branch-name: String, balance: real)

BANKCUSTOMER (customer-name: String, customer-street: String, customer-city: String)

DEPOSITER (customer-name: String, accno: int)

LOAN (loan-number: int, branch-name: String, amount: real)

- 1) Create the above tables by properly specifying the primary keys and the foreign keys.

```
1 CREATE TABLE BRANCH(branchname varchar(20), branchcity varchar(30), assets REAL, PRIMARY KEY(branchname));
2 CREATE TABLE ACCOUNTS(accno int, branchname varchar(20), balance real, PRIMARY KEY(accno), FOREIGN KEY(branchname) REFERENCES BRANCH(branchname));
3 CREATE TABLE BANKCUSTOMER(customer_name VARCHAR(20), customer_street VARCHAR(20), customer_city VARCHAR(20), PRIMARY KEY(customer_name));
4 CREATE TABLE DEPOSITER(customer_name VARCHAR(20), accno int, FOREIGN KEY(accno) REFERENCES ACCOUNTS(accno));
5 CREATE TABLE LOAN(loannumber int, branchname VARCHAR(30), amount REAL, PRIMARY KEY(loannumber), FOREIGN KEY(branchname) REFERENCES BRANCH(branchname));
```

Table	Action	Rows	Type	Collation	Size	Overhead
accounts		0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
bankcustomer		0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
branch		0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
depositor		0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
loan		0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
5 tables	Sum	0	InnoDB	utf8mb4_general_ci	128.0 KiB	0 B

2) Enter at least 5 tuples for each relation.

'BRANCH' table:

The screenshot shows the MySQL Workbench interface with the 'branch' table selected. The top navigation bar includes 'Server: 127.0.0.1', 'Database: banking_enterprise', and 'Table: branch'. Below the navigation bar is a toolbar with 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', and 'Import' buttons. A large text area labeled 'Run SQL query/queries on table banking_enterprise.branch:' contains the following SQL code:

```
1 INSERT INTO BRANCH VALUES("SBI Chamrajpet","Bangalore",50000);
2 INSERT INTO BRANCH VALUES("SBI Residency Road","Bangalore",10000);
3 INSERT INTO BRANCH VALUES("SBI Shivaji Road","Bombay",20000);
4 INSERT INTO BRANCH VALUES("SBI Parliament Road","Delhi",10000);
5 INSERT INTO BRANCH VALUES("SBI Jantarmantar","Delhi",20000);
```

The screenshot shows the MySQL Workbench interface with the 'branch' table selected. The top navigation bar includes 'Server: 127.0.0.1', 'Database: banking_enterprise', and 'Table: branch'. Below the navigation bar is a toolbar with 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', 'Operations', 'Tracking', and 'Triggers' buttons. A message 'Show query box' is visible. The main area displays five successful INSERT queries with their execution times:

- 1 row inserted. (Query took 0.1481 seconds.)
INSERT INTO BRANCH VALUES("SBI Chamrajpet","Bangalore",50000)
[Edit inline] [Edit] [Create PHP code]
- 1 row inserted. (Query took 0.0291 seconds.)
INSERT INTO BRANCH VALUES("SBI Residency Road","Bangalore",10000)
[Edit inline] [Edit] [Create PHP code]
- 1 row inserted. (Query took 0.1043 seconds.)
INSERT INTO BRANCH VALUES("SBI Shivaji Road","Bombay",20000)
[Edit inline] [Edit] [Create PHP code]
- 1 row inserted. (Query took 0.2135 seconds.)
INSERT INTO BRANCH VALUES("SBI Parliament Road","Delhi",10000)
[Edit inline] [Edit] [Create PHP code]
- 1 row inserted. (Query took 0.0658 seconds.)
INSERT INTO BRANCH VALUES("SBI Jantarmantar","Delhi",20000)
[Edit inline] [Edit] [Create PHP code]

Server: 127.0.0.1 » Database: banking_enterprise » Table: branch

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#)

Showing rows 0 - 4 (5 total, Query took 0.0005 seconds.)

```
SELECT * FROM `branch`
```

Show all | Number of rows: 25 ▾ Filter rows: Sort by k

+ Options

	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	branchname	branchcity	assets
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	SBI Chamrajpet	Bangalore	50000
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	SBI Jantarmantar	Delhi	20000
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	SBI Parliament Road	Delhi	10000
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	SBI Residency Road	Bangalore	10000
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	SBI Shivaji Road	Bombay	20000

'ACCOUNTS' table:

Server: 127.0.0.1 » Database: banking_enterprise » Table: branch

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#)

Run SQL query/queries on table banking_enterprise.branch: [?](#)

```

1 INSERT INTO ACCOUNTS VALUES("1","SBI Chamrajpet",2000);
2 INSERT INTO ACCOUNTS VALUES("2","SBI Residency Road",5000);
3 INSERT INTO ACCOUNTS VALUES("3","SBI Shivaji Road",6000);
4 INSERT INTO ACCOUNTS VALUES("4","SBI Parliament Road",9000);
5 INSERT INTO ACCOUNTS VALUES("5","SBI Jantarmantar",8000);
6 INSERT INTO ACCOUNTS VALUES("6","SBI Shivaji Road",4000);
7 INSERT INTO ACCOUNTS VALUES("8","SBI Residency Road",4000);
8 INSERT INTO ACCOUNTS VALUES("9","SBI Parliament Road",3000);
9 INSERT INTO ACCOUNTS VALUES("10","SBI Residency Road",5000);
10 INSERT INTO ACCOUNTS VALUES("11","SBI Jantarmantar",2000);
```

Server: 127.0.0.1 » Database: banking_enterprise » Table: branch

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

✓ 1 row inserted. (Query took 0.1276 seconds.)

```
INSERT INTO ACCOUNTS VALUES("1","SBI Chamrajpet",2000)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0638 seconds.)

```
INSERT INTO ACCOUNTS VALUES("2","SBI Residency Road",5000)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.2065 seconds.)

```
INSERT INTO ACCOUNTS VALUES("3","SBI Shivaji Road",6000)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0454 seconds.)

```
INSERT INTO ACCOUNTS VALUES("4","SBI Parliament Road",9000)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0453 seconds.)

```
INSERT INTO ACCOUNTS VALUES("5","SBI Jantarmantar",8000)
```

[Edit inline] [Edit] [Create PHP code]

Server: 127.0.0.1 » Database: banking_enterprise » Table: branch

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

✓ 1 row inserted. (Query took 0.0322 seconds.)

```
INSERT INTO ACCOUNTS VALUES("6","SBI Shivaji Road",4000)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0319 seconds.)

```
INSERT INTO ACCOUNTS VALUES("8","SBI Residency Road",4000)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0313 seconds.)

```
INSERT INTO ACCOUNTS VALUES("9","SBI Parliament Road",3000)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0324 seconds.)

```
INSERT INTO ACCOUNTS VALUES("10","SBI Residency Road",5000)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0325 seconds.)

```
INSERT INTO ACCOUNTS VALUES("11","SBI Jantarmantar",2000)
```

[Edit inline] [Edit] [Create PHP code]

Server: 127.0.0.1 » Database: banking_enterprise » Table: accounts

Browse Structure SQL Search Insert Export

Showing rows 0 - 9 (10 total, Query took 0.0007 seconds.)

```
SELECT * FROM `accounts`
```

Show all Number of rows: 25 Filter rows: Search this table

+ Options

	accno	branchname	balance
<input type="checkbox"/>	1	SBI Chamrajpet	2000
<input type="checkbox"/>	2	SBI Residency Road	5000
<input type="checkbox"/>	3	SBI Shivaji Road	6000
<input type="checkbox"/>	4	SBI Parliament Road	9000
<input type="checkbox"/>	5	SBI Jantarmantar	8000
<input type="checkbox"/>	6	SBI Shivaji Road	4000
<input type="checkbox"/>	8	SBI Residency Road	4000
<input type="checkbox"/>	9	SBI Parliament Road	3000
<input type="checkbox"/>	10	SBI Residency Road	5000
<input type="checkbox"/>	11	SBI Jantarmantar	2000

'BANKCUSTOMER' table:

Server: 127.0.0.1 » Database: banking_enterprise » Table: bankcustomer

Browse Structure SQL Search Insert Export Import Privileges

Run SQL query/queries on table banking_enterprise.bankcustomer:

```

1 INSERT INTO BANKCUSTOMER VALUES("Avinash","Bull_Temple_Road","Bangalore");
2 INSERT INTO BANKCUSTOMER VALUES("Dinesh","Bannergatta_Road","Bangalore");
3 INSERT INTO BANKCUSTOMER VALUES("Satish","NationalCollege_Road","Bangalore");
4 INSERT INTO BANKCUSTOMER VALUES("Ramesh","Akbar_Road","Delhi");
5 INSERT INTO BANKCUSTOMER VALUES("Rohini","Prithviraj_Road","Delhi");

```

Server: 127.0.0.1 » Database: banking_enterprise » Table: bankcustomer

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Show query box

✓ 1 row inserted. (Query took 0.0985 seconds.)

```
INSERT INTO BANKCUSTOMER VALUES("Avinash","Bull_Temple_Road","Bangalore")
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0380 seconds.)

```
INSERT INTO BANKCUSTOMER VALUES("Dinesh","BannerGatta_Road","Bangalore")
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0448 seconds.)

```
INSERT INTO BANKCUSTOMER VALUES("Satish","NationalCollege_Road","Bangalore")
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0455 seconds.)

```
INSERT INTO BANKCUSTOMER VALUES("Ramesh","Akbar_Road","Delhi")
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0324 seconds.)

```
INSERT INTO BANKCUSTOMER VALUES("Rohini","Prithiviraj_Road","Delhi")
```

[Edit inline] [Edit] [Create PHP code]

Server: 127.0.0.1 » Database: banking_enterprise » Table: bankcustomer

Browse Structure SQL Search Insert Export Import

✓ Showing rows 0 - 4 (5 total, Query took 0.0006 seconds.)

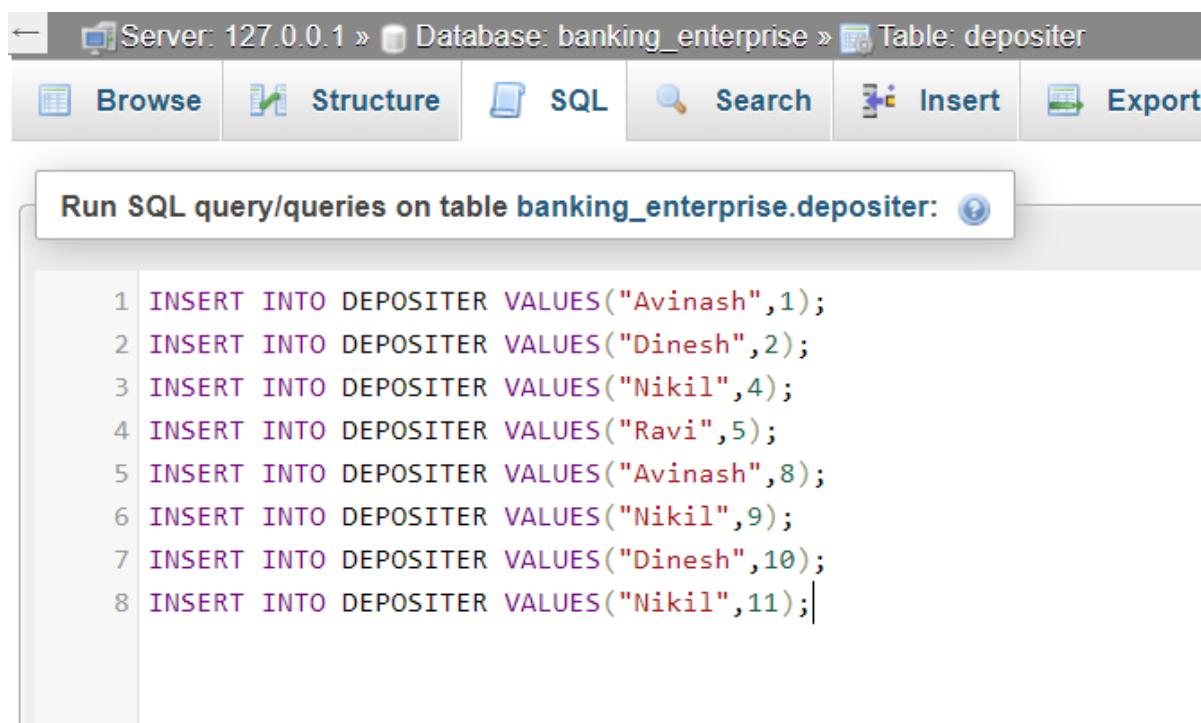
```
SELECT * FROM `bankcustomer`
```

Show all | Number of rows: 25 ▾ Filter rows: Search this table Sort by I

+ Options

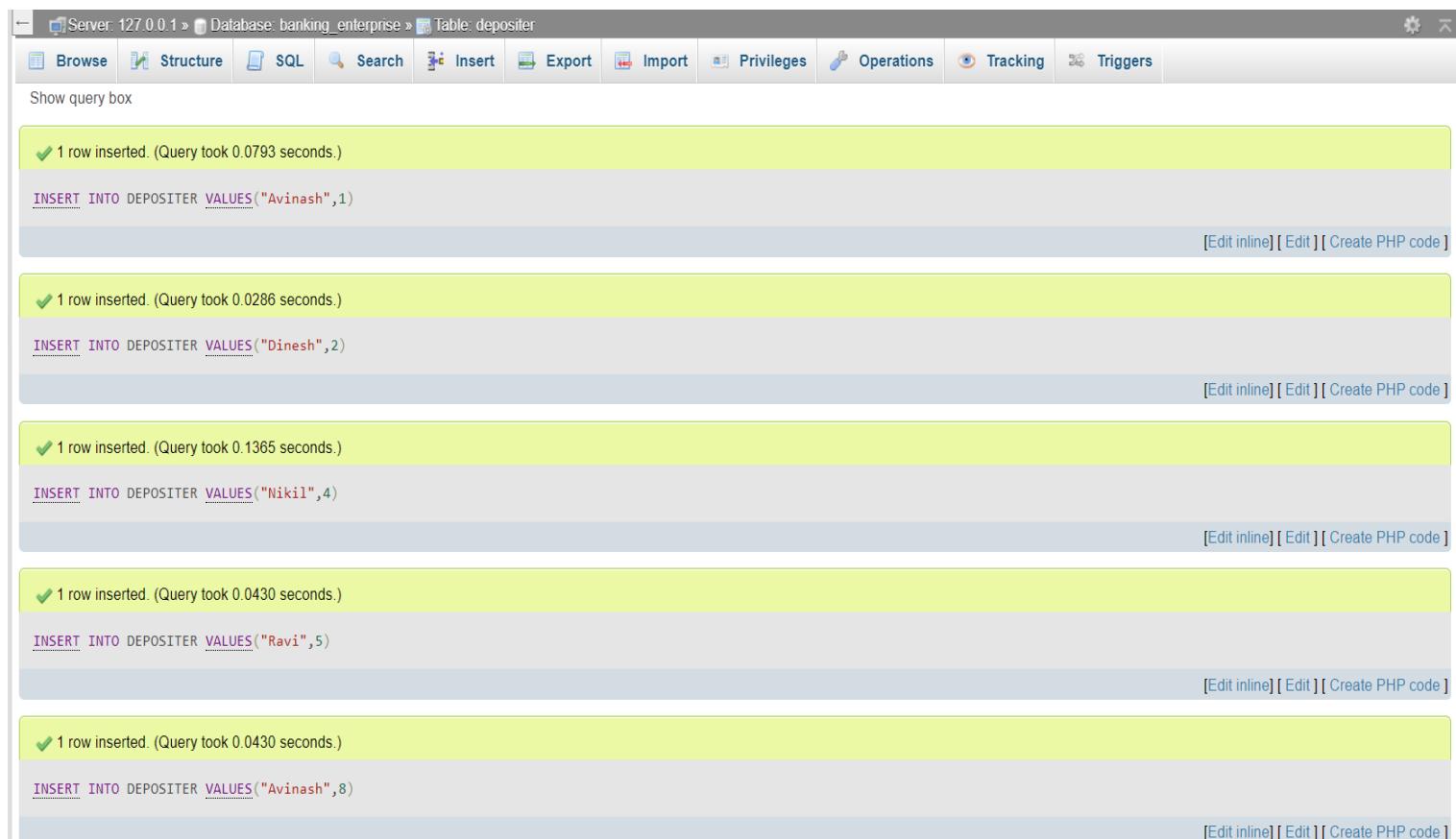
	customer_name	customer_street	customer_city
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	Avinash	Bull_Temple_Road	Bangalore
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	Dinesh	BannerGatta_Road	Bangalore
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	Ramesh	Akbar_Road	Delhi
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	Rohini	Prithiviraj_Road	Delhi
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	Satish	NationalCollege_Road	Bangalore

'DEPOSITER' table:



The screenshot shows the phpMyAdmin interface for the 'depositor' table. The top navigation bar indicates the connection is to 'Server: 127.0.0.1', the database is 'banking_enterprise', and the current table is 'depositor'. Below the navigation bar, there are tabs for 'Browse', 'Structure', 'SQL', 'Search', 'Insert', and 'Export'. The 'SQL' tab is selected. A large text area below the tabs contains the following SQL code:

```
1 INSERT INTO DEPOSITER VALUES("Avinash",1);
2 INSERT INTO DEPOSITER VALUES("Dinesh",2);
3 INSERT INTO DEPOSITER VALUES("Nikil",4);
4 INSERT INTO DEPOSITER VALUES("Ravi",5);
5 INSERT INTO DEPOSITER VALUES("Avinash",8);
6 INSERT INTO DEPOSITER VALUES("Nikil",9);
7 INSERT INTO DEPOSITER VALUES("Dinesh",10);
8 INSERT INTO DEPOSITER VALUES("Nikil",11);
```



The screenshot shows the phpMyAdmin interface for the 'depositor' table after executing the SQL code from the previous screenshot. The top navigation bar and tabs are identical. The main area displays the results of the inserted rows:

- 1 row inserted. (Query took 0.0793 seconds.)
INSERT INTO DEPOSITER VALUES("Avinash",1)
[Edit inline] [Edit] [Create PHP code]
- 1 row inserted. (Query took 0.0286 seconds.)
INSERT INTO DEPOSITER VALUES("Dinesh",2)
[Edit inline] [Edit] [Create PHP code]
- 1 row inserted. (Query took 0.1365 seconds.)
INSERT INTO DEPOSITER VALUES("Nikil",4)
[Edit inline] [Edit] [Create PHP code]
- 1 row inserted. (Query took 0.0430 seconds.)
INSERT INTO DEPOSITER VALUES("Ravi",5)
[Edit inline] [Edit] [Create PHP code]
- 1 row inserted. (Query took 0.0430 seconds.)
INSERT INTO DEPOSITER VALUES("Avinash",8)
[Edit inline] [Edit] [Create PHP code]

Server: 127.0.0.1 » Database: banking_enterprise » Table: depositer

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Privileges](#) [Operations](#) [Tracking](#) [Triggers](#)

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0430 seconds.)

```
INSERT INTO DEPOSITER VALUES("Ravi",5)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0430 seconds.)

```
INSERT INTO DEPOSITER VALUES("Avinash",8)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0429 seconds.)

```
INSERT INTO DEPOSITER VALUES("Nikil",9)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0319 seconds.)

```
INSERT INTO DEPOSITER VALUES("Dinesh",10)
```

[Edit inline] [Edit] [Create PHP code]

✓ 1 row inserted. (Query took 0.0322 seconds.)

```
INSERT INTO DEPOSITER VALUES("Nikil",11)
```

[Edit inline] [Edit] [Create PHP code]

Server: 127.0.0.1 » Database: banking_enterprise » Table: depositer

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#)

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit

✓ Showing rows 0 - 7 (8 total, Query took 0.0019 seconds.)

```
SELECT * FROM `depositor`
```

Show all | Number of rows: 25 ▾ Filter rows: Search this table

+ Options

customer_name	accno
Avinash	1
Dinesh	2
Nikil	4
Ravi	5
Avinash	8
Nikil	9
Dinesh	10
Nikil	11

'LOAN' table:

The screenshot shows the MySQL Workbench interface with the following details:

- Server: 127.0.0.1
- Database: banking_enterprise
- Table: loan
- Tab selected: SQL
- Text area content:

```
1 INSERT INTO LOAN VALUES(1,"SBI Chamrajpet",1000);
2 INSERT INTO LOAN VALUES(2,"SBI Residency Road",2000);
3 INSERT INTO LOAN VALUES(3,"SBI Shivaji Road",3000);
4 INSERT INTO LOAN VALUES(4,"SBI Parliament Road",4000);
5 INSERT INTO LOAN VALUES(5,"SBI Jantarmantar",5000);
```

The screenshot shows the MySQL Workbench interface with the following details:

- Server: 127.0.0.1
- Database: banking_enterprise
- Table: loan
- Tab selected: SQL
- Text area content:

```
1 row inserted. (Query took 0.1079 seconds.)  
INSERT INTO LOAN VALUES(1, "SBI Chamrajpet", 1000)  
[Edit inline] [Edit] [Create PHP code]  
  
1 row inserted. (Query took 0.0395 seconds.)  
INSERT INTO LOAN VALUES(2, "SBI Residency Road", 2000)  
[Edit inline] [Edit] [Create PHP code]  
  
1 row inserted. (Query took 0.1152 seconds.)  
INSERT INTO LOAN VALUES(3, "SBI Shivaji Road", 3000)  
[Edit inline] [Edit] [Create PHP code]  
  
1 row inserted. (Query took 0.0435 seconds.)  
INSERT INTO LOAN VALUES(4, "SBI Parliament Road", 4000)  
[Edit inline] [Edit] [Create PHP code]  
  
1 row inserted. (Query took 0.0432 seconds.)  
INSERT INTO LOAN VALUES(5, "SBI Jantarmantar", 5000)  
[Edit inline] [Edit] [Create PHP code]
```

✓ Showing rows 0 - 4 (5 total, Query took 0.0008 seconds.)

```
SELECT * FROM `loan`
```

 Show all

Number of rows:

25

Filter rows:

Search this table

+ Options

	loannumber	branchname	amount
<input type="checkbox"/> Edit Copy Delete	1	SBI Chamrajpet	1000
<input type="checkbox"/> Edit Copy Delete	2	SBI Residency Road	2000
<input type="checkbox"/> Edit Copy Delete	3	SBI Shivaji Road	3000
<input type="checkbox"/> Edit Copy Delete	4	SBI Parliament Road	4000
<input type="checkbox"/> Edit Copy Delete	5	SBI Jantarmantar	5000

- 3) Find all the customers who have at least two accounts at the Main branch.

Query: `SELECT C.customer_name FROM BANKCUSTOMER C WHERE EXISTS (SELECT D.customer_name, COUNT(D.customer_name) FROM DEPOSITER D, ACCOUNTS BA WHERE D.accno=BA.accno AND C.customer_name=D.customer_name AND BA.branchname='SBI Residency Road' GROUP BY D.customer_name HAVING COUNT(D.customer_name) >= 2);`

Server: 127.0.0.1 » Database: banking_enterprise

Structure SQL Search Query Export Import Operations Privileges Routines Events

Run SQL query/queries on database banking_enterprise:

```
1 SELECT C.customer_name FROM BANKCUSTOMER C WHERE EXISTS(SELECT D.customer_name, COUNT(D.customer_name) FROM DEPOSITER D,
2 ACCOUNTS BA WHERE D.accno=BA.accno AND C.customer_name=D.customer_name AND BA.branchname='SBI Residency Road' GROUP BY
3 D.customer_name HAVING COUNT(D.customer_name)>=2);
```

Server: 127.0.0.1 » Database: banking_enterprise » Table: BANKCUSTOMER

Browse Structure SQL Search Insert Export Import

Show query box

Showing rows 0 - 0 (1 total, Query took 0.0025 seconds.)

```
SELECT C.customer_name FROM BANKCUSTOMER C WHERE EXISTS(SELECT D.customer_name, COUNT(C.customer_name)=D.customer_name AND BA.branchname='SBI Residency Road' GROUP BY D.cus
```

Show all | Number of rows: 25

Filter rows: Search this table

+ Options



customer_name

Edit Copy Delete Dinesh

- 4) Find all the customers who have an account at all the branches located in a specific city.

Query: `SELECT customer_name FROM BANKCUSTOMER WHERE NOT EXISTS (SELECT branchname FROM BRANCH WHERE branchcity='Delhi');`

Server: 127.0.0.1 » Database: banking_enterprise » Table: DEPOSITER

Browse Structure SQL Search Insert Export Import Privileges

Run SQL query/queries on table banking_enterprise.DEPOSITER:

```
1 | SELECT customer_name FROM BANKCUSTOMER WHERE NOT EXISTS(SELECT branchname
2 | FROM BRANCH WHERE branchcity='Delhi');
```

Server: 127.0.0.1 » Database: banking_enterprise » Table: BANKCUSTOMER

Browse Structure SQL Search Insert Export Import Privileges Operations

Show query box

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0239 seconds.)

```
SELECT customer_name FROM BANKCUSTOMER WHERE NOT EXISTS(SELECT branchname FROM BRANCH WHERE branchcity='Delhi')
```

customer_name

Query results operations

Create view

- 5) Demonstrate how you delete all account tuples at every branch located in a specific city.

Query: `DELETE FROM ACCOUNTS WHERE branchname IN (SELECT branchname FROM BRANCH WHERE branchcity='Bombay');`

Server: 127.0.0.1 » Database: banking_enterprise

Structure SQL Search Query Export Import Operations

Run SQL query/queries on database banking_enterprise:

```
1 DELETE FROM ACCOUNTS WHERE branchname IN (SELECT branchname FROM BRANCH  
2 WHERE branchcity='Bombay');
```

Server: 127.0.0.1 » Database: banking_enterprise » Table: accounts

Browse Structure SQL Search Insert Export Import

Showing rows 0 - 7 (8 total, Query took 0.0004 seconds.)

```
SELECT * FROM `accounts`
```

Show all Number of rows: 25 Filter rows: Search this table Sort by k

+ Options

	accno	branchname	balance
<input type="checkbox"/> Edit Copy Delete	1	SBI Chamrajpet	2000
<input type="checkbox"/> Edit Copy Delete	2	SBI Residency Road	5000
<input type="checkbox"/> Edit Copy Delete	4	SBI Parliament Road	9000
<input type="checkbox"/> Edit Copy Delete	5	SBI Jantarmantar	8000
<input type="checkbox"/> Edit Copy Delete	8	SBI Residency Road	4000
<input type="checkbox"/> Edit Copy Delete	9	SBI Parliament Road	3000
<input type="checkbox"/> Edit Copy Delete	10	SBI Residency Road	5000
<input type="checkbox"/> Edit Copy Delete	11	SBI Jantarmantar	2000

PROGRAM 3 : SUPPLIER DATABASE

Consider the following schema for a Supplier database:

SUPPLIERS (sid: integer, sname: string, address: string)

PARTS (pid: integer, pname: string, color: string)

CATALOG (sid: integer, pid: integer, cost: real)

The Catalog relation lists the prices charged for parts by Suppliers. Write the following queries in SQL:

Creating tables:

The screenshot shows the MySQL Workbench interface with the database set to 'supplier'. Three queries are run in sequence:

```
CREATE TABLE SUPPLIERS (sid int, sname varchar(20), address varchar(40), PRIMARY KEY(sid))
```

```
CREATE TABLE PARTS (pid int, pname varchar(20), color varchar(20), PRIMARY KEY(pid))
```

```
CREATE TABLE CATALOG (sid int, pid int, cost real, FOREIGN KEY(sid) REFERENCES SUPPLIERS(sid) ON DELETE SET NULL ON UPDATE CASCADE, FOREIGN KEY(pid) REFERENCES PARTS(pid) ON DELETE SET NULL ON UPDATE CASCADE)
```

Each query returns an empty result set.

The screenshot shows the MySQL Workbench interface with the database set to 'supplier'. The 'Tables' tab is selected, displaying the following table information:

Table	Action	Rows	Type	Collation	Size	Overhead
catalog		0	InnoDB	utf8mb4_general_ci	48.0 KiB	-
parts		0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
suppliers		0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
3 tables	Sum	0	InnoDB	utf8mb4_general_ci	80.0 KiB	0 B

'SUPPLIERS' table:

The screenshot shows the MySQL Workbench interface with the following details:

- Server: 127.0.0.1
- Database: supplier
- Table: suppliers
- Toolbar buttons: Browse, Structure, SQL, Search, Insert, Export, Import, Privileges.
- Text area: Run SQL query/queries on table supplier.suppliers:
1 INSERT INTO `suppliers`(`sid`, `sname`, `address`) VALUES (101,'Acme Widget','Bangalore');
2 INSERT INTO `suppliers`(`sid`, `sname`, `address`) VALUES (102,'Johns','Kolkata');
3 INSERT INTO `suppliers`(`sid`, `sname`, `address`) VALUES (103,'Vimal','Mumbai');
4 INSERT INTO `suppliers`(`sid`, `sname`, `address`) VALUES (104,'Reliance','Delhi');

The screenshot shows the MySQL Workbench interface with the following details:

- Server: 127.0.0.1
- Database: supplier
- Table: suppliers
- Toolbar buttons: Browse, Structure, SQL, Search, Insert, Export.
- Message bar: Showing rows 0 - 3 (4 total, Query took 0.0006 seconds.)
- Text area: SELECT * FROM `suppliers`
- Filter controls: Show all (unchecked), Number of rows: 25, Filter rows: Search this table.
- Table data grid:

	sid	sname	address
<input type="checkbox"/>	101	Acme Widget	Bangalore
<input type="checkbox"/>	102	Johns	Kolkata
<input type="checkbox"/>	103	Vimal	Mumbai
<input type="checkbox"/>	104	Reliance	Delhi

'PARTS' table:

The screenshot shows the MySQL Workbench interface with the following details:

- Server: 127.0.0.1
- Database: supplier
- Table: parts
- Tab selected: SQL
- Text area content:

```
1 INSERT INTO `parts`(`pid`, `pname`, `color`) VALUES (201,'Book','Red');
2 INSERT INTO `parts`(`pid`, `pname`, `color`) VALUES (202,'Pen','Red');
3 INSERT INTO `parts`(`pid`, `pname`, `color`) VALUES (203,'Pencil','Green');
4 INSERT INTO `parts`(`pid`, `pname`, `color`) VALUES (204,'Mobile','Green');
5 INSERT INTO `parts`(`pid`, `pname`, `color`) VALUES (205,'Charger','Black');
```

The screenshot shows the MySQL Workbench interface with the following details:

- Server: 127.0.0.1
- Database: supplier
- Table: parts
- Tab selected: Browse
- Status message: "Showing rows 0 - 4 (5 total, Query took 0.0006 seconds.)"
- SQL query:

```
SELECT * FROM `parts`
```
- Filter options: Show all (unchecked), Number of rows: 25, Filter rows: Search
- Table data:

	pid	pname	color
<input type="checkbox"/>	201	Book	Red
<input type="checkbox"/>	202	Pen	Red
<input type="checkbox"/>	203	Pencil	Green
<input type="checkbox"/>	204	Mobile	Green
<input type="checkbox"/>	205	Charger	Black

'CATALOG' table:

Server: 127.0.0.1 » Database: supplier » Table: catalog

Browse Structure SQL Search Insert Export Import

Run SQL query/queries on table supplier.catalog:

```
1 INSERT INTO `catalog`(`sid`, `pid`, `cost`) VALUES (101,201,10);
2 INSERT INTO `catalog`(`sid`, `pid`, `cost`) VALUES (101,202,10);
3 INSERT INTO `catalog`(`sid`, `pid`, `cost`) VALUES (101,203,30);
4 INSERT INTO `catalog`(`sid`, `pid`, `cost`) VALUES (101,204,10);
5 INSERT INTO `catalog`(`sid`, `pid`, `cost`) VALUES (101,205,10);
6 INSERT INTO `catalog`(`sid`, `pid`, `cost`) VALUES (102,201,10);
7 INSERT INTO `catalog`(`sid`, `pid`, `cost`) VALUES (102,202,20);
8 INSERT INTO `catalog`(`sid`, `pid`, `cost`) VALUES (103,203,30);
9 INSERT INTO `catalog`(`sid`, `pid`, `cost`) VALUES (104,203,40);
```

Server: 127.0.0.1 » Database: supplier » Table: catalog

Browse Structure SQL Search Insert

Current selection does not contain a unique column. Grid edit, checkboxes, etc. will not work correctly.

Showing rows 0 - 8 (9 total, Query took 0.0013 seconds.)

```
SELECT * FROM `catalog`
```

Show all | Number of rows: 25 Filter rows:

+ Options

sid	pid	cost
101	201	10
101	202	10
101	203	30
101	204	10
101	205	10
102	201	10
102	202	20
103	203	30
104	203	40

1)Find the pnames of parts for which there is some supplier.

Query: `SELECT DISTINCT PNAME FROM PARTS, CATALOG WHERE CATALOG.PID=PARTS.PID AND CATALOG.SID IS NOT NULL`

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1 » **Database:** supplier
- Structure, SQL, Search, Query, Export, Import, Operations, Privileges** tabs are visible in the top menu.
- Show query box:** The query `SELECT DISTINCT PNAME FROM PARTS, CATALOG WHERE CATALOG.PID=PARTS.PID AND CATALOG.SID IS NOT NULL` is run.
- Result message:** "Showing rows 0 - 4 (5 total, Query took 0.1048 seconds.)"
- Result table:** A table with one column "PNAME" containing the values Book, Pen, Pencil, Mobile, and Charger.
- Filter options:** Show all (unchecked), Number of rows: 25, Filter rows: Search this table.
- Table structure:** A sidebar shows the structure of the PARTS table with columns PID, PNAME, and SID.

2)Find the snames of suppliers who supply every part.

Query: `SELECT S.NAME FROM SUPPLIERS S WHERE NOT EXISTS ((SELECT P.PID FROM PARTS P) EXCEPT (SELECT C.PID FROM CATALOG C WHERE C.SID=S.SID))`;

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1 » **Database:** supplier » **Table:** SUPPLIERS
- Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Tracking** tabs are visible in the top menu.
- Show query box:** The query `SELECT S.SNAME FROM SUPPLIERS S WHERE NOT EXISTS((SELECT P.PID FROM PARTS P) EXCEPT (SELECT C.PID FROM CATALOG C WHERE C.SID = S.SID))` is run.
- Result message:** "Showing rows 0 - 0 (1 total, Query took 0.0022 seconds.)"
- Result table:** An empty table with one column "SNAME".
- Filter options:** Show all (unchecked), Number of rows: 25, Filter rows: Search this table.
- Table structure:** A sidebar shows the structure of the SUPPLIERS table with columns SID, SNAME, and ADDRESS.

3)Find the snames of suppliers who supply every red part.

Query: `SELECT S.SNAME FROM SUPPLIERS S WHERE NOT EXISTS ((SELECT P.PID FROM PARTS P WHERE P.COLOR='Red') EXCEPT (SELECT C.PID FROM CATALOG C, PARTS P WHERE C.SID=S.SID AND C.PID=P.PID AND P.COLOR='Red'));`

The screenshot shows the phpMyAdmin interface for the 'supplier' database. The current table is 'SUPPLIERS'. The query executed was:

```
SELECT S.SNAME FROM SUPPLIERS S WHERE NOT EXISTS (( SELECT P.PID FROM PARTS P WHERE P.COLOR = 'Red') EXCEPT (SELECT C.PID FROM CATALOG C, PARTS P WHERE C.SID = S.SID AND C.PID = P.PID AND P.COLOR = 'Red'))
```

The results show two rows:

SNAME
Acme Widget
Johns

4)Find the pnames of parts supplied by Acme Widget Suppliers and by no one else.

Query: `SELECT P.NAME FROM PARTS P, CATALOG C, SUPPLIERS S WHERE P.PID=C.PID AND C.SID=S.SID AND S.SNAME='Acme Widget' AND NOT EXISTS (SELECT * FROM CATALOG C1, SUPPLIERS S1 WHERE P.PID=C1.PID AND C1.SID=S1.SID AND S1.SNAME<>'Acme Widget');`

The screenshot shows the phpMyAdmin interface for the 'supplier' database. The current table is 'PARTS'. The query executed was:

```
SELECT P.PNAME FROM PARTS P, CATALOG C, SUPPLIERS S WHERE P.PID = C.PID AND C.SID = S.SID AND S.SNAME = 'Acme Widget' AND NOT EXISTS (SELECT * FROM CATALOG C1, SUPPLIERS S1 WHERE P.PID = C1.PID AND C1.SID = S1.SID AND S1.SNAME<>'Acme Widget')
```

The results show two rows:

PNAME
Mobile
Charger

5)Find the sids of suppliers who charge more for some part than the average cost of that part (averaged over all the suppliers who supply that part).

Query: `SELECT DISTINCT C.SID FROM CATALOG C WHERE C.COST > (SELECT AVG (C1.COST) FROM CATALOG C1 WHERE C1.PID=C.PID);`

Server: 127.0.0.1 » Database: supplier » Table: CATALOG

Browse Structure SQL Search Insert Export Import Privileges Operations

Show query box

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 1 (2 total, Query took 0.1253 seconds.)

```
SELECT DISTINCT C.SID FROM CATALOG C WHERE C.COST > ( SELECT AVG (C1.COST) FROM CATALOG C1 WHERE C1.PID = C.PID )
```

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

+ Options

SID
102
104

6)For each part, find the sname of the supplier who charges the most for that part.

Query: `SELECT * FROM SUPPLIERS S, PARTS P, CATALOG C WHERE P.PID=C.PID AND C.SID=S.SID AND COST IN (SELECT MAX(COST) FROM CATALOG C1 WHERE C1.PID=P.PID);`

Server: 127.0.0.1 » Database: supplier » Table: CATALOG

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Show query box

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 5 (6 total, Query took 0.0024 seconds.)

```
SELECT * FROM SUPPLIERS S, PARTS P, CATALOG C WHERE P.PID=C.PID AND C.SID=S.SID AND COST IN (SELECT MAX(COST) FROM CATALOG C1 WHERE C1.PID=P.PID)
```

Show all Number of rows: 25 Filter rows: Search this table Profiling [Edit inline] [Edit] [Explain SQL]

+ Options

sid	sname	address	pid	pname	color	sid	pid	cost
101	Acme Widget	Bangalore	201	Book	Red	101	201	10
101	Acme Widget	Bangalore	204	Mobile	Green	101	204	10
101	Acme Widget	Bangalore	205	Charger	Black	101	205	10
102	Johns	Kolkata	201	Book	Red	102	201	10
102	Johns	Kolkata	202	Pen	Red	102	202	20
104	Reliance	Delhi	203	Pencil	Green	104	203	40

7)Find the sids of suppliers who supply only red parts.

Query: **SELECT * FROM SUPPLIERS S WHERE NOT EXISTS ((SELECT PID FROM CATALOG C WHERE C.SID=S.SID) EXCEPT (SELECT PID FROM PARTS WHERE COLOR='Red'));**

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** supplier
- Table:** SUPPLIERS
- Toolbar:** Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Tracking, Triggers.
- Query Result:** A green message bar at the top says "Showing rows 0 - 0 (1 total, Query took 0.0017 seconds.)". Below it is the executed query: "SELECT * FROM SUPPLIERS S WHERE NOT EXISTS ((SELECT PID FROM CATALOG C WHERE C.SID=S.SID) EXCEPT (SELECT PID FROM PARTS WHERE COLOR='Red'))".
- Table View:** The SUPPLIERS table has columns sid, sname, and address. One row is visible: sid 102, sname Johns, address Kolkata.
- Options:** Show all (unchecked), Number of rows: 25, Filter rows: Search this table.
- Buttons:** + Options, Row selection, Edit, Copy, Delete.

PROGRAM 4 : STUDENT FACULTY DATABASE

Consider the following Student enrolment database for course:

STUDENT (snum: integer, sname: string, major: string, level: string, age:integer)

CLASS (name: string, meets at: time, room: string, fid: integer)

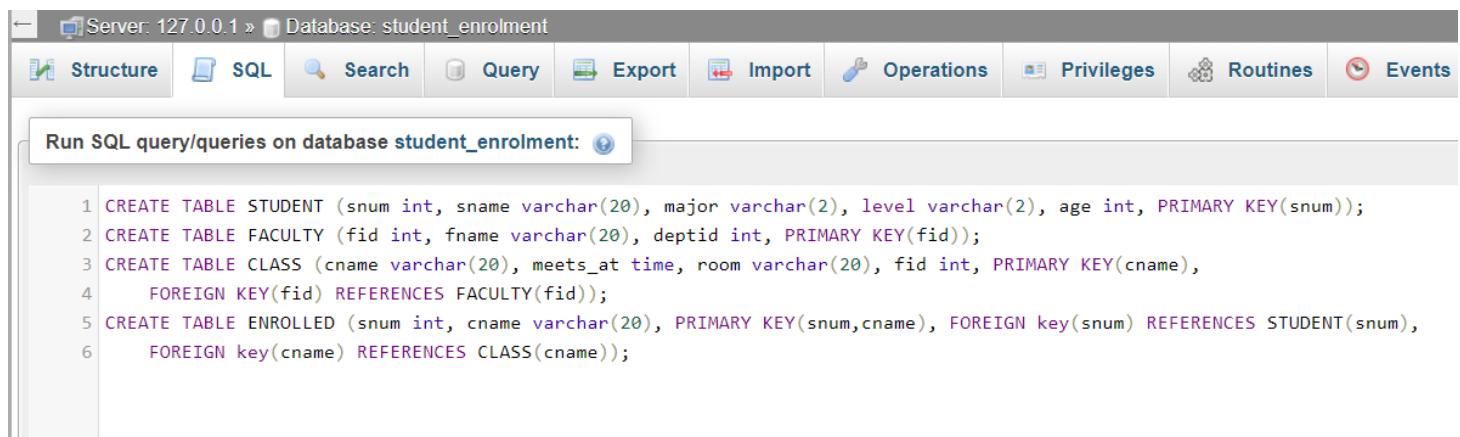
ENROLLED (snum: integer, cname: string)

FACULTY (fid: integer, fname: string, deptid: integer)

The meaning of these relations is straightforward; for example, Enrolled has one record per student-class pair such that the student is enrolled in the class. Level is a two character code with 4 different values (example: Junior: JR etc)

Write the following queries in SQL. No duplicates should be printed in any of the answers.

Creating tables:



```
1 CREATE TABLE STUDENT (snum int, sname varchar(20), major varchar(2), level varchar(2), age int, PRIMARY KEY(snum));
2 CREATE TABLE FACULTY (fid int, fname varchar(20), deptid int, PRIMARY KEY(fid));
3 CREATE TABLE CLASS (cname varchar(20), meets_at time, room varchar(20), fid int, PRIMARY KEY(cname),
4 FOREIGN KEY(fid) REFERENCES FACULTY(fid));
5 CREATE TABLE ENROLLED (snum int, cname varchar(20), PRIMARY KEY(snum,cname), FOREIGN key(snum) REFERENCES STUDENT(snum),
6 FOREIGN key(cname) REFERENCES CLASS(cname));
```

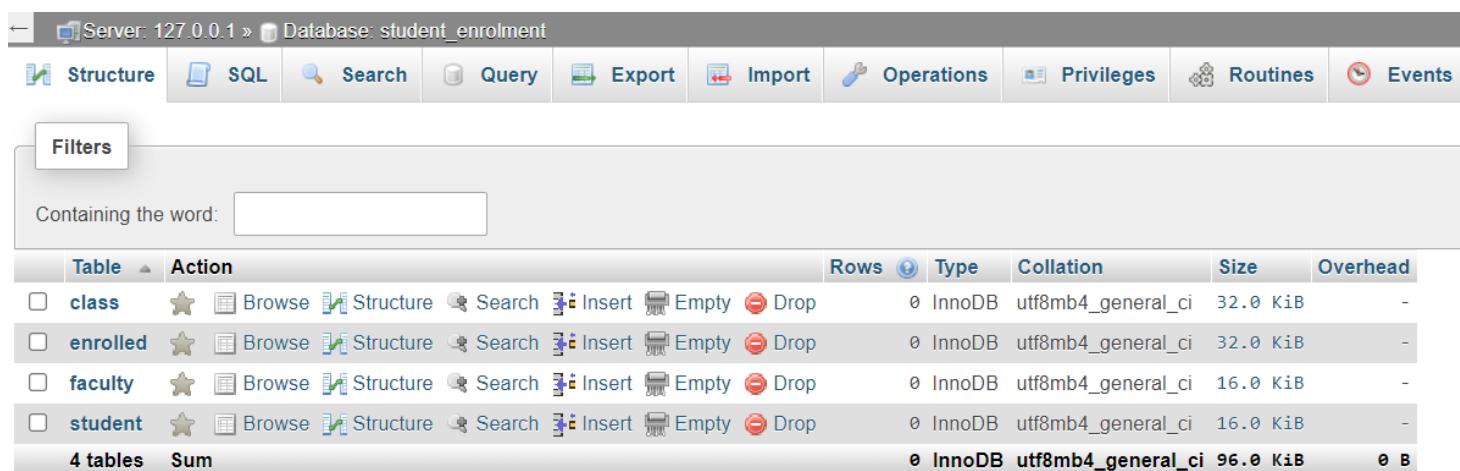
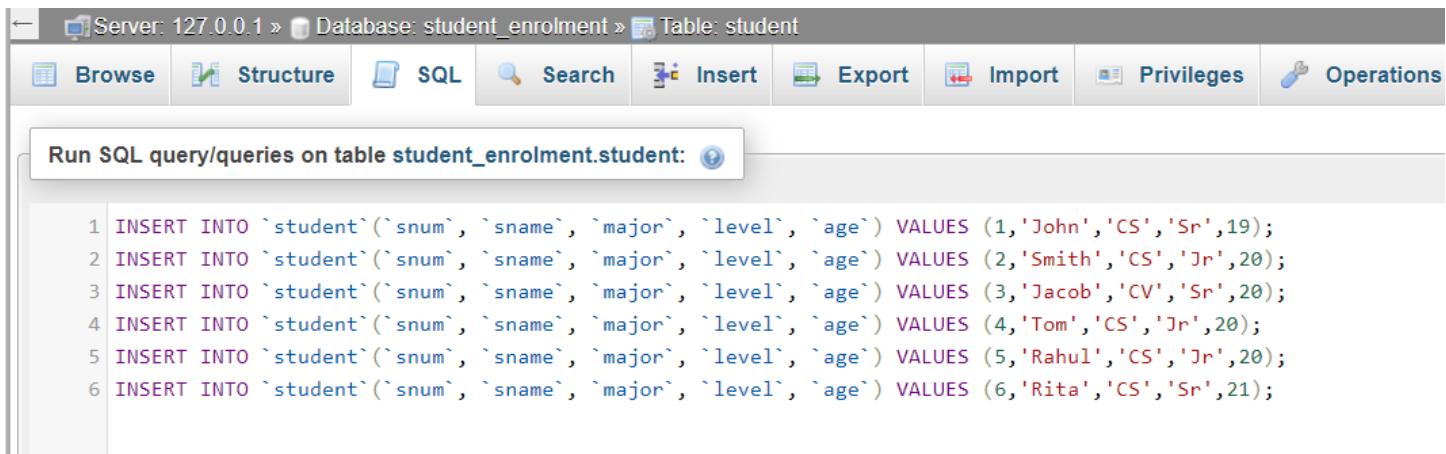


Table	Action	Rows	Type	Collation	Size	Overhead
class		0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
enrolled		0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
faculty		0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
student		0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
4 tables	Sum	0	InnoDB	utf8mb4_general_ci	96.0 KiB	0 B

'STUDENT' table:

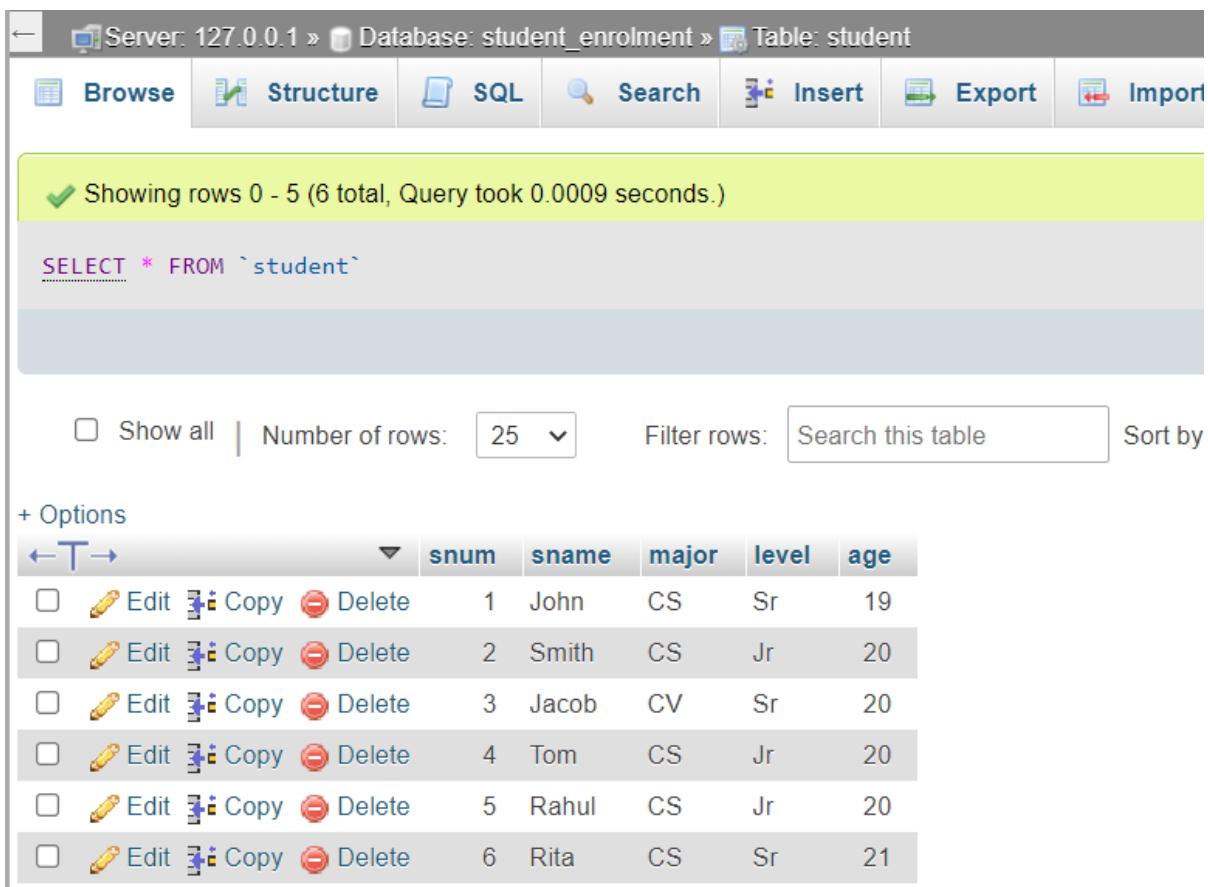


Server: 127.0.0.1 » Database: student_enrolment » Table: student

Browse Structure SQL Search Insert Export Import Privileges Operations

Run SQL query/queries on table student_enrolment.student: [?](#)

```
1 INSERT INTO `student`(`snum`, `sname`, `major`, `level`, `age`) VALUES (1,'John','CS','Sr',19);
2 INSERT INTO `student`(`snum`, `sname`, `major`, `level`, `age`) VALUES (2,'Smith','CS','Jr',20);
3 INSERT INTO `student`(`snum`, `sname`, `major`, `level`, `age`) VALUES (3,'Jacob','CV','Sr',20);
4 INSERT INTO `student`(`snum`, `sname`, `major`, `level`, `age`) VALUES (4,'Tom','CS','Jr',20);
5 INSERT INTO `student`(`snum`, `sname`, `major`, `level`, `age`) VALUES (5,'Rahul','CS','Jr',20);
6 INSERT INTO `student`(`snum`, `sname`, `major`, `level`, `age`) VALUES (6,'Rita','CS','Sr',21);
```



Server: 127.0.0.1 » Database: student_enrolment » Table: student

Browse Structure SQL Search Insert Export Import

Showing rows 0 - 5 (6 total, Query took 0.0009 seconds.)

```
SELECT * FROM `student`
```

Show all Number of rows: 25 Filter rows: Search this table Sort by

+ Options

	snum	sname	major	level	age
<input type="checkbox"/> Edit Copy Delete	1	John	CS	Sr	19
<input type="checkbox"/> Edit Copy Delete	2	Smith	CS	Jr	20
<input type="checkbox"/> Edit Copy Delete	3	Jacob	CV	Sr	20
<input type="checkbox"/> Edit Copy Delete	4	Tom	CS	Jr	20
<input type="checkbox"/> Edit Copy Delete	5	Rahul	CS	Jr	20
<input type="checkbox"/> Edit Copy Delete	6	Rita	CS	Sr	21

'FACULTY' table:

The screenshot shows the MySQL Workbench interface. The title bar indicates the connection is to 'Server: 127.0.0.1', the database is 'student_enrolment', and the current table is 'faculty'. Below the title bar is a toolbar with buttons for 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', and 'Import'. A sub-toolbar below the main toolbar has a button for running SQL queries on the 'student_enrolment.faculty' table. The main area contains five numbered SQL INSERT statements:

```
1 INSERT INTO `faculty`(`fid`, `fname`, `deptid`) VALUES (11, 'Harish', 1000);
2 INSERT INTO `faculty`(`fid`, `fname`, `deptid`) VALUES (12, 'MV', 1000);
3 INSERT INTO `faculty`(`fid`, `fname`, `deptid`) VALUES (13, 'Mira', 1001);
4 INSERT INTO `faculty`(`fid`, `fname`, `deptid`) VALUES (14, 'Shiva', 1002);
5 INSERT INTO `faculty`(`fid`, `fname`, `deptid`) VALUES (15, 'Nupur', 1000);
```

The screenshot shows the MySQL Workbench interface with the same connection details as the first screenshot. The 'Structure' tab is selected in the toolbar. A message bar at the top indicates 'Showing rows 0 - 4 (5 total, Query took 0.0006 seconds.)'. Below the message bar is a SQL query window containing the command: 'SELECT * FROM `faculty`'. At the bottom of the interface are filters for 'Show all', 'Number of rows' (set to 25), and a search bar for 'Filter rows'.

+ Options

	fid	fname	deptid
<input type="checkbox"/>	11	Harish	1000
<input type="checkbox"/>	12	MV	1000
<input type="checkbox"/>	13	Mira	1001
<input type="checkbox"/>	14	Shiva	1002
<input type="checkbox"/>	15	Nupur	1000

'CLASS' table:

Server: 127.0.0.1 » Database: student_enrolment » Table: class

Browse Structure SQL Search Insert Export Import Privileges Operations

Run SQL query/queries on table student_enrolment.class: [?](#)

```
1 INSERT INTO `class`(`cname`, `meets_at`, `room`, `fid`) VALUES ('class1','12/11/15 10:15:16','R1',14);
2 INSERT INTO `class`(`cname`, `meets_at`, `room`, `fid`) VALUES ('class10','12/11/15 10:15:16','R128',14);
3 INSERT INTO `class`(`cname`, `meets_at`, `room`, `fid`) VALUES ('class2','12/11/15 10:15:20','R2',12);
4 INSERT INTO `class`(`cname`, `meets_at`, `room`, `fid`) VALUES ('class3','12/11/15 10:15:25','R3',11);
5 INSERT INTO `class`(`cname`, `meets_at`, `room`, `fid`) VALUES ('class4','12/11/15 20:15:20','R4',14);
6 INSERT INTO `class`(`cname`, `meets_at`, `room`, `fid`) VALUES ('class5','12/11/15 20:15:20','R3',15);
7 INSERT INTO `class`(`cname`, `meets_at`, `room`, `fid`) VALUES ('class6','12/11/15 13:20:20','R2',14);
8 INSERT INTO `class`(`cname`, `meets_at`, `room`, `fid`) VALUES ('class7','12/11/15 10:10:10','R3',14);
```

Server: 127.0.0.1 » Database: student_enrolment » Table: class

Browse Structure SQL Search Insert Export

Showing rows 0 - 7 (8 total, Query took 0.0007 seconds.)

```
SELECT * FROM `class`
```

Show all Number of rows: 25 Filter rows: Search this table

+ Options

	cname	meets_at	room	fid
<input type="checkbox"/>	class1	10:15:16	R1	14
<input type="checkbox"/>	class10	10:15:16	R128	14
<input type="checkbox"/>	class2	10:15:20	R2	12
<input type="checkbox"/>	class3	10:15:25	R3	11
<input type="checkbox"/>	class4	20:15:20	R4	14
<input type="checkbox"/>	class5	20:15:20	R3	15
<input type="checkbox"/>	class6	13:20:20	R2	14
<input type="checkbox"/>	class7	10:10:10	R3	14

'ENROLLED' table:

Server: 127.0.0.1 » Database: student_enrolment » Table: enrolled

Browse Structure SQL Search Insert Export Import

Run SQL query/queries on table student_enrolment.enrolled:

```
1 INSERT INTO `enrolled`(`snum`, `cname`) VALUES (1,'class1');
2 INSERT INTO `enrolled`(`snum`, `cname`) VALUES (2,'class1');
3 INSERT INTO `enrolled`(`snum`, `cname`) VALUES (3,'class3');
4 INSERT INTO `enrolled`(`snum`, `cname`) VALUES (4,'class3');
5 INSERT INTO `enrolled`(`snum`, `cname`) VALUES (5,'class4');
6 INSERT INTO `enrolled`(`snum`, `cname`) VALUES (1,'class5');
7 INSERT INTO `enrolled`(`snum`, `cname`) VALUES (2,'class5');
8 INSERT INTO `enrolled`(`snum`, `cname`) VALUES (3,'class5');
9 INSERT INTO `enrolled`(`snum`, `cname`) VALUES (4,'class5');
10 INSERT INTO `enrolled`(`snum`, `cname`) VALUES (5,'class5');
```

← Server: 127.0.0.1 » Database: student_enrolment » Table: enrolled

Browse Structure SQL Search Insert Export

Showing rows 0 - 9 (10 total, Query took 0.0007 seconds.)

```
SELECT * FROM `enrolled`
```

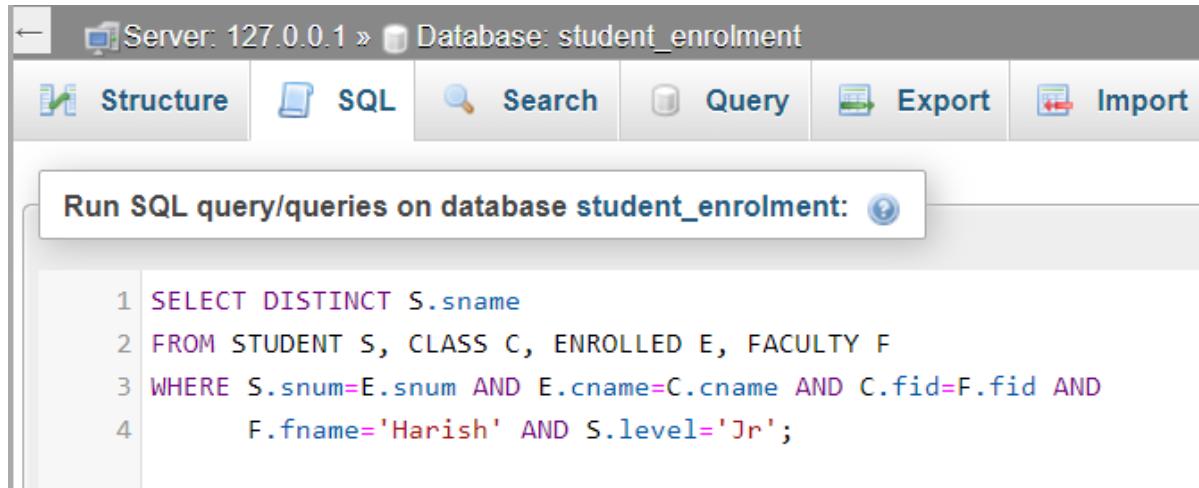
Show all Number of rows: 25 Filter rows: Search this table

+ Options

	snum	cname
<input type="checkbox"/>	1	class1
<input type="checkbox"/>	1	class5
<input type="checkbox"/>	2	class1
<input type="checkbox"/>	2	class5
<input type="checkbox"/>	3	class3
<input type="checkbox"/>	3	class5
<input type="checkbox"/>	4	class3
<input type="checkbox"/>	4	class5
<input type="checkbox"/>	5	class4
<input type="checkbox"/>	5	class5

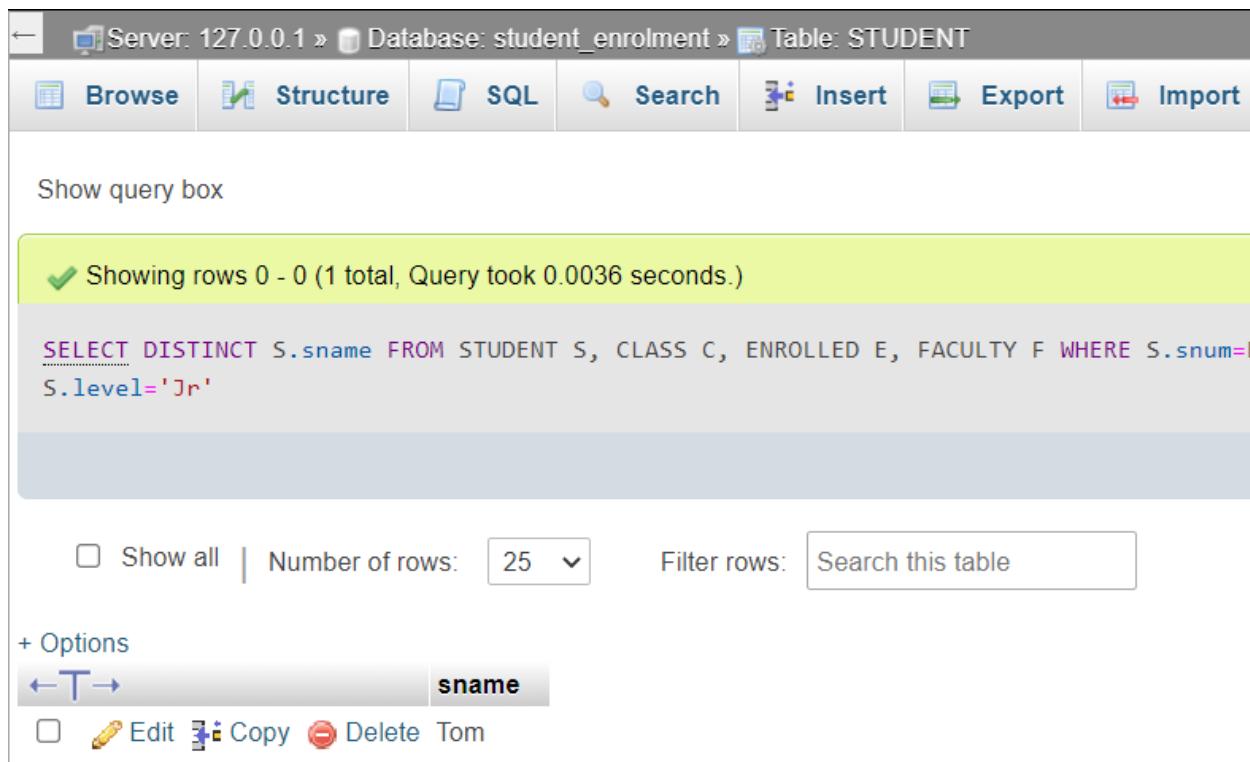
1) Find the names of all Juniors (level = Jr) who are enrolled in a class taught by 'Harish'.

Query: `SELECT DISTINCT S.sname FROM STUDENT S, CLASS C, ENROLLED E, FACULTY F WHERE S.snum = E.snum AND E cname = C cname AND C fid = F fid AND F fname = 'Harish' AND S level = 'Jr';`



The screenshot shows the MySQL Workbench interface. The title bar indicates "Server: 127.0.0.1 » Database: student_enrolment". The tabs at the top are Structure, SQL, Search, Query, Export, and Import. The SQL tab is selected. Below the tabs is a search bar with the placeholder "Run SQL query/queries on database student_enrolment: ?". The main area contains the following SQL code:

```
1 SELECT DISTINCT S.sname
2 FROM STUDENT S, CLASS C, ENROLLED E, FACULTY F
3 WHERE S.snum=E.snum AND E.cname=C cname AND C.fid=F.fid AND
4       F.fname='Harish' AND S.level='Jr';
```



The screenshot shows the MySQL Workbench interface with the title bar "Server: 127.0.0.1 » Database: student_enrolment » Table: STUDENT". The tabs at the top are Browse, Structure, SQL, Search, Insert, Export, and Import. The SQL tab is selected. Below the tabs is a message box with a green checkmark and the text "Showing rows 0 - 0 (1 total, Query took 0.0036 seconds.)". The main area displays the query: `SELECT DISTINCT S.sname FROM STUDENT S, CLASS C, ENROLLED E, FACULTY F WHERE S.snum=E.snum AND S.level='Jr'`. At the bottom, there are options to "Show all" (unchecked), set the "Number of rows" to 25 (selected), and a "Filter rows" input field with the placeholder "Search this table". Below these are "Options" (+ Options) and a toolbar with icons for Edit, Copy, and Delete, and the name "Tom".

2)Find the names of all classes that either meet in room R128 or have five or more Students enrolled.

Query: `SELECT DISTINCT cname FROM class WHERE room='R128' OR cname IN (SELECT E.cname FROM ENROLLED E GROUP BY E.cname HAVING COUNT(*)>=5);`

Server: 127.0.0.1 » Database: student_enrolment » Table: STUDENT

Browse Structure SQL Search Insert Export

Run SQL query/queries on table student_enrolment.STUDENT:

```
1 SELECT DISTINCT cname
2 FROM class
3 WHERE room='R128' OR cname IN (SELECT E.cname
4                                     FROM ENROLLED E GROUP BY E.cname
5                                     HAVING COUNT(*) >= 5);
```

Server: 127.0.0.1 » Database: student_enrolment » Table: class

Browse Structure SQL Search Insert Export Import

Show query box

Showing rows 0 - 1 (2 total, Query took 0.0266 seconds.)

```
SELECT DISTINCT cname FROM class WHERE room='R128' OR cname IN (SELECT E.cname FROM E
```

Show all Number of rows: 25 Filter rows: Search this table Sort by k

+ Options

	cname
<input type="checkbox"/>	class10
<input type="checkbox"/>	class5

← →

Edit Copy Delete

Edit Copy Delete

3)Find the names of all students who are enrolled in two classes that meet at the same time.

Query: SELECT DISTINCT S.sname FROM STUDENT S WHERE S.snum IN (SELECT E1.snum FROM ENROLLED E1, ENROLLED E2, CLASS C1, CLASS C2 WHERE E1.snum = E2.snum AND E1 cname <> E2 cname AND E1 cname = C1 cname AND E2 cname = C2 cname AND C1 meets_at = C2 meets_at);

Server: 127.0.0.1 » Database: student_enrolment » Table: class

Browse Structure SQL Search Insert Export Import Privileges

Run SQL query/queries on table student_enrolment.class:

```
1 SELECT DISTINCT S.sname
2 FROM STUDENT S
3 WHERE S.snum IN (SELECT E1.snum
4                   FROM ENROLLED E1, ENROLLED E2, CLASS C1, CLASS C2
5                   WHERE E1.snum=E2.snum AND E1.cname<>E2.cname AND
6                   E1.cname=C1.cname AND E2.cname=C2.cname AND C1.meets_at=C2.meets_at);|
```

Server: 127.0.0.1 » Database: student_enrolment » Table: STUDENT

Browse Structure SQL Search Insert Export

Show query box

Showing rows 0 - 0 (1 total, Query took 0.1082 seconds.)

```
SELECT DISTINCT S.sname FROM STUDENT S WHERE S.snum IN (SELECT E1.snum F
E1.cname<>E2.cname AND E1.cname=C1.cname AND E2.cname=C2.cname AND C1.me
```

Show all Number of rows: 25 Filter rows: Search this table

+ Options

Edit Copy Delete Rahul

4)Find the names of faculty members who teach in every room in which some class is taught.

Query: `SELECT F.fname, F.fid FROM FACULTY F WHERE F.fid IN (SELECT fid FROM CLASS GROUP BY fid HAVING COUNT(*)=(SELECT COUNT(DISTINCT room) FROM CLASS));`

Server: 127.0.0.1 » Database: student_enrolment » Table: STUDENT

Browse Structure SQL Search Insert Export Import

Run SQL query/queries on table student_enrolment.STUDENT:

```
1 SELECT F.fname, F.fid
2 FROM FACULTY F
3 WHERE F.fid IN (SELECT fid
4                 FROM CLASS
5                 GROUP BY fid
6                 HAVING COUNT(*) = (SELECT COUNT(DISTINCT room)
7                                     FROM CLASS));
```

Server: 127.0.0.1 » Database: student_enrolment » Table: FACULTY

Browse Structure SQL Search Insert Export Import

Show query box

Showing rows 0 - 0 (1 total, Query took 0.1073 seconds.)

```
SELECT F.fname, F.fid FROM FACULTY F WHERE F.fid IN (SELECT fid FROM CLASS GROUP BY
```

Show all Number of rows: 25 Filter rows: Search this table

+ Options

Edit Copy Delete Shiva

fname	fid
Shiva	14

5)Find the names of faculty members for whom the combined enrolment of the courses that they teach is less than five.

Query: `SELECT DISTINCT F.fname FROM FACULTY F WHERE 5 > (SELECT COUNT(E.snum) FROM CLASS C, ENROLLED E WHERE C.cname = E cname AND C.fid = F.fid);`

Server: 127.0.0.1 » Database: student_enrolment » Table: FACULTY

Browse Structure SQL Search Insert Export

Run SQL query/queries on table student_enrolment.FACULTY:

```
1 SELECT DISTINCT F.fname
2 FROM FACULTY F
3 WHERE 5 > (SELECT COUNT(E.snum)
4             FROM CLASS C, ENROLLED E
5             WHERE C.cname=E cname AND C.fid=F.fid);
6
```

Server: 127.0.0.1 » Database: student_enrolment » Table: FACULTY

Browse Structure SQL Search Insert Export

Show query box

Showing rows 0 - 3 (4 total, Query took 0.0020 seconds.)

```
SELECT DISTINCT F.fname FROM FACULTY F WHERE 5 > (SELECT COUNT(E.snum) FF
```

Show all | Number of rows: 25 Filter rows: Search this table

+ Options

← → fname

	Edit	Copy	Delete	fname
<input type="checkbox"/>				Harish
<input type="checkbox"/>				MV
<input type="checkbox"/>				Mira
<input type="checkbox"/>				Shiva

6)Find the names of students who are not enrolled in any class.

Query: `SELECT DISTINCT S.sname FROM STUDENT S WHERE S.snum NOT IN (SELECT E.snum FROM ENROLLED E);`

Server: 127.0.0.1 » Database: student_enrolment

Structure SQL Search Query Export Import

Run SQL query/queries on database student_enrolment: [?](#)

```
1 SELECT DISTINCT S.sname
2 FROM STUDENT S
3 WHERE S.snum NOT IN (SELECT E.snum
4                      FROM ENROLLED E);
```

Server: 127.0.0.1 » Database: student_enrolment » Table: STUDENT

Browse Structure SQL Search Insert Export

Show query box

Showing rows 0 - 0 (1 total, Query took 0.0018 seconds.)

`SELECT DISTINCT S.sname FROM STUDENT S WHERE S.snum NOT IN (SELECT E.snum`

Show all

Number of rows:

25

Filter rows:

Search this table

+ Options



sname

Edit Copy Delete Rita

7) For each age value that appears in Students, find the level value that appears most often. For ex, if there are more FR level students aged 18 than SR, JR, or SO students aged 18, you should print the pair (18, FR).

Query: `SELECT S.age, S.level FROM STUDENT S GROUP BY S.age, S.level HAVING S.level IN (SELECT S1.level FROM STUDENT S1 WHERE S1.age=S.age GROUP BY S1.age, S1.level HAVING COUNT(*) >= ALL (SELECT COUNT(*) FROM STUDENT S2 WHERE S1.age=S2.age GROUP BY S2.level, S2.age)) ORDER BY S.age;`

Server: 127.0.0.1 » Database: student_enrolment

Structure SQL Search Query Export Import Operations

Run SQL query/queries on database student_enrolment:

```
1 SELECT S.age, S.level
2 FROM STUDENT S
3 GROUP BY S.age, S.level
4 HAVING S.level IN (SELECT S1.level
5                 FROM STUDENT S1
6                 WHERE S1.age=S.age
7                 GROUP BY S1.age, S1.level
8                 HAVING COUNT(*) >= ALL (SELECT COUNT(*)
9                     FROM STUDENT S2
10                    WHERE S1.age=S2.age
11                    GROUP BY S2.level, S2.age))
12 ORDER BY S.age;
```

Server: 127.0.0.1 » Database: student_enrolment » Table: STUDENT

Browse Structure SQL Search Insert Export Import

Show query box

Showing rows 0 - 2 (3 total, Query took 0.0049 seconds.) [age: 19... - 21...]

```
SELECT S.age, S.level FROM STUDENT S GROUP BY S.age, S.level HAVING S.level IN (SELECT S1.level
FROM STUDENT S1 WHERE S1.age=S.age GROUP BY S1.age, S1.level HAVING COUNT(*) >= ALL (SELECT COUNT(*) FROM STUDENT S2 WHERE S1.age=S2.age GROUP BY S2.level, S2.age)) ORDER BY S.age;
```

Show all Number of rows: 25 Filter rows: Search this table Sort by k

+ Options

	age	level
<input type="checkbox"/>	19	Sr
<input type="checkbox"/>	20	Jr
<input type="checkbox"/>	21	Sr

PROGRAM 5 : AIRLINE FLIGHT DATABASE

Consider the following database that keeps track of airline flight information:

FLIGHTS (flno: integer, from: string, to: string, distance: integer, departs: time, arrives: time, price: integer)

AIRCRAFT (aid: integer, fname: string, cruisingrange: integer)

CERTIFIED (eid: integer, aid: integer)

EMPLOYEES (eid: integer, fname: string, salary: integer)

Note that the Employees relation describes pilots and other kinds of employees as well. Every pilot is certified for some aircraft and only pilots are certified to fly. Write each of the following queries in SQL.

Creating tables:

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** airline_flight
- Toolbar:** Structure, SQL, Search, Query, Export, Import, Operations, Privileges
- SQL Editor:** Run SQL query/queries on database airline_flight:
The code in the editor is:

```
1 create table flights(flno int, fromplace varchar(15), toplace varchar(15), distance int,
2 departs datetime, arrives datetime, price int, primary key (flno));
3 desc flights;
4 create table aircraft(aid int, fname varchar(15), cruisingrange int, primary key (aid));
5 desc aircraft;
6 create table employees(eid int, fname varchar(15), salary int, primary key (eid));
7 desc employees;
8 create table certified(eid int, aid int, foreign key (eid) references employees(eid), foreign
key (aid) references aircraft(aid));
9 desc certified;
```

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** airline_flight
- Toolbar:** Structure, SQL, Search, Query, Export, Import, Operations, Privileges, Routines, Events
- Filters:** Containing the word: [empty input field]
- Table View:** Shows the structure of four tables:

Table	Action	Rows	Type	Collation	Size	Overhead
aircraft	Browse Structure Search Insert Empty Drop	9	InnoDB	utf8mb4_general_ci	16.0 KiB	-
certified	Browse Structure Search Insert Empty Drop	13	InnoDB	utf8mb4_general_ci	48.0 KiB	-
employees	Browse Structure Search Insert Empty Drop	7	InnoDB	utf8mb4_general_ci	16.0 KiB	-
flights	Browse Structure Search Insert Empty Drop	8	InnoDB	utf8mb4_general_ci	16.0 KiB	-
4 tables	Sum	37	InnoDB	utf8mb4_general_ci	96.0 KiB	0 B

'FLIGHTS' table:

Server: 127.0.0.1 » Database: airline_flight

Structure SQL Search Query Export Import Operations Privileges Routines Events

Run SQL query/queries on database airline_flight:

```

1 insert into flights values(101, 'Bangalore', 'Delhi', 2500, '2005-05-13 07:15:31', '2005-05-13 18:15:31', 5000);
2 insert into flights values(102, 'Bangalore', 'Lucknow', 3000, '2013-05-05 07:15:31', '2013-05-05 11:15:31', 6000);
3 insert into flights values(103, 'Lucknow', 'Delhi', 500, '2013-05-05 12:15:31', '2013-05-05 17:15:31', 3000);
4 insert into flights values(107, 'Bangalore', 'Frankfurt', 8000, '2013-05-05 07:15:31', '2013-05-05 22:15:31', 60000);
5 insert into flights values(104, 'Bangalore', 'Frankfurt', 8500, '2013-05-05 07:15:31', '2013-05-05 23:15:31', 75000);
6 insert into flights values(105, 'Kolkata', 'Delhi', 3400, '2013-05-05 07:15:31', '2013-05-05 09:15:31', 7000);
7 insert into flights values(106, 'Bangalore', 'Kolkata', 1000, '2013-05-05 01:15:30', '2013-05-05 09:20:30', 10000);
8 insert into flights values(108, 'Lucknow', 'Kolkata', 1000, '2013-05-05 11:30:30', '2013-05-05 15:20:30', 10000);
9
10 commit;
11
12 select * from flights;

```

Server: 127.0.0.1 » Database: airline_flight » Table: flights

Browse Structure SQL Search Insert Export Import Privileges Operations

Showing rows 0 - 7 (8 total, Query took 0.0007 seconds.)

SELECT * FROM `flights`

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

+ Options

	flno	fromplace	toplace	distance	departs	arrives	price
<input type="checkbox"/>	101	Bangalore	Delhi	2500	2005-05-13 07:15:31	2005-05-13 18:15:31	5000
<input type="checkbox"/>	102	Bangalore	Lucknow	3000	2013-05-05 07:15:31	2013-05-05 11:15:31	6000
<input type="checkbox"/>	103	Lucknow	Delhi	500	2013-05-05 12:15:31	2013-05-05 17:15:31	3000
<input type="checkbox"/>	104	Bangalore	Frankfurt	8500	2013-05-05 07:15:31	2013-05-05 23:15:31	75000
<input type="checkbox"/>	105	Kolkata	Delhi	3400	2013-05-05 07:15:31	2013-05-05 09:15:31	7000
<input type="checkbox"/>	106	Bangalore	Kolkata	1000	2013-05-05 01:15:30	2013-05-05 09:20:30	10000
<input type="checkbox"/>	107	Bangalore	Frankfurt	8000	2013-05-05 07:15:31	2013-05-05 22:15:31	60000
<input type="checkbox"/>	108	Lucknow	Kolkata	1000	2013-05-05 11:30:30	2013-05-05 15:20:30	10000

'AIRCRAFT' table:

Server: 127.0.0.1 » Database: airline_flight

Structure SQL Search Query Export Import

Run SQL query/queries on database **airline_flight**: [?](#)

```
1 insert into aircraft values(101, '747', 3000);
2 insert into aircraft values(102, 'Boeing', 900);
3 insert into aircraft values(103, '647', 800);
4 insert into aircraft values(104, 'Dreamliner', 10000);
5 insert into aircraft values(105, 'Boeing', 3500);
6 insert into aircraft values(106, '707', 1500);
7 insert into aircraft values(107, 'Dream', 120000);
8 insert into aircraft values(108, '707', 760);
9 insert into aircraft values(109, '747', 1000);
10
11 commit;
12
13 select * from aircraft;
```

Server: 127.0.0.1 » Database: airline_flight » Table: aircraft

Browse Structure SQL Search Insert Export

Showing rows 0 - 8 (9 total, Query took 0.0006 seconds.)

```
SELECT * FROM `aircraft`
```

Show all | Number of rows: 25 Filter rows:

+ Options

	aid	aname	cruisingrange
<input type="checkbox"/>	101	747	3000
<input type="checkbox"/>	102	Boeing	900
<input type="checkbox"/>	103	647	800
<input type="checkbox"/>	104	Dreamliner	10000
<input type="checkbox"/>	105	Boeing	3500
<input type="checkbox"/>	106	707	1500
<input type="checkbox"/>	107	Dream	120000
<input type="checkbox"/>	108	707	760
<input type="checkbox"/>	109	747	1000

'EMPLOYEES' table:

Server: 127.0.0.1 » Database: airline_flight

Structure SQL Search Query Export Import

Run SQL query/queries on database airline_flight: [?](#)

```
1 insert into employees values(701, 'A', 50000);
2 insert into employees values(702, 'B', 100000);
3 insert into employees values(703, 'C', 150000);
4 insert into employees values(704, 'D', 90000);
5 insert into employees values(705, 'E', 40000);
6 insert into employees values(706, 'F', 60000);
7 insert into employees values(707, 'G', 90000);
8
9 commit;
10
11 select * from employees;
```

Server: 127.0.0.1 » Database: airline_flight » Table: employees

Browse Structure SQL Search Insert

Showing rows 0 - 6 (7 total, Query took 0.0006 seconds.)

```
SELECT * FROM `employees`
```

Show all Number of rows: 25 Filter rows:

+ Options

	<input type="button" value="T"/>	<input type="button" value="L"/>	<input type="button" value="R"/>	eid	ename	salary
<input type="checkbox"/>	Edit	Copy	Delete	701	A	50000
<input type="checkbox"/>	Edit	Copy	Delete	702	B	100000
<input type="checkbox"/>	Edit	Copy	Delete	703	C	150000
<input type="checkbox"/>	Edit	Copy	Delete	704	D	90000
<input type="checkbox"/>	Edit	Copy	Delete	705	E	40000
<input type="checkbox"/>	Edit	Copy	Delete	706	F	60000
<input type="checkbox"/>	Edit	Copy	Delete	707	G	90000

'CERTIFIED' table:

Server: 127.0.0.1 » Database: airline_flight » Table: certified

Browse Structure SQL Search Insert

Run SQL query/queries on table airline_flight.certified:

```
1 insert into certified values(701, 101);
2 insert into certified values(701, 102);
3 insert into certified values(701, 106);
4 insert into certified values(701, 105);
5 insert into certified values(702, 104);
6 insert into certified values(703, 104);
7 insert into certified values(704, 104);
8 insert into certified values(702, 107);
9 insert into certified values(703, 107);
10 insert into certified values(704, 107);
11 insert into certified values(702, 101);
12 insert into certified values(702, 108);
13 insert into certified values(701, 109);
14
15 commit;
16
17 select * from certified;
```

Server: 127.0.0.1 » Database: airline_flight » Table: certified

Browse Structure SQL Search Insert

Current selection does not contain a unique column. Grid edit, checkboxes, etc. will not work correctly.

Showing rows 0 - 12 (13 total, Query took 0.0005 seconds.)

```
SELECT * FROM `certified`
```

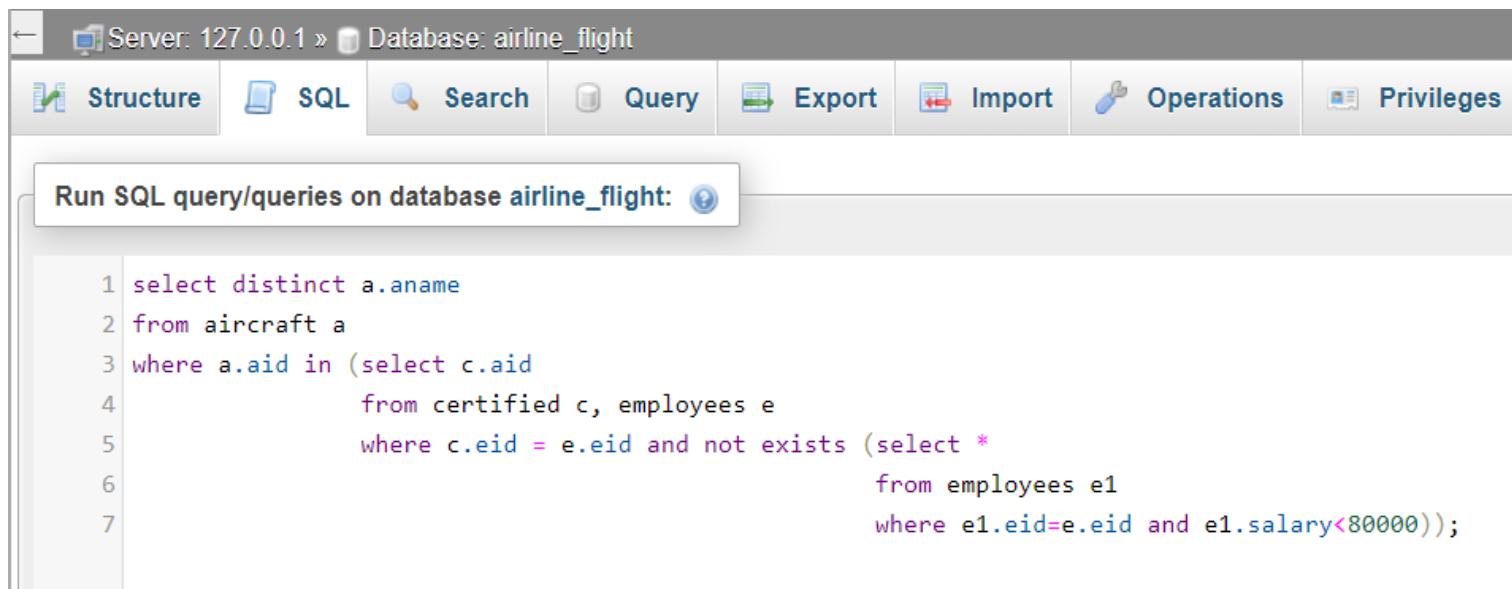
Show all | Number of rows: 25 Filter rows:

+ Options

eid	aid
701	101
701	102
701	106
701	105
702	104
703	104
704	104
702	107
703	107
704	107
702	101
702	108
701	109

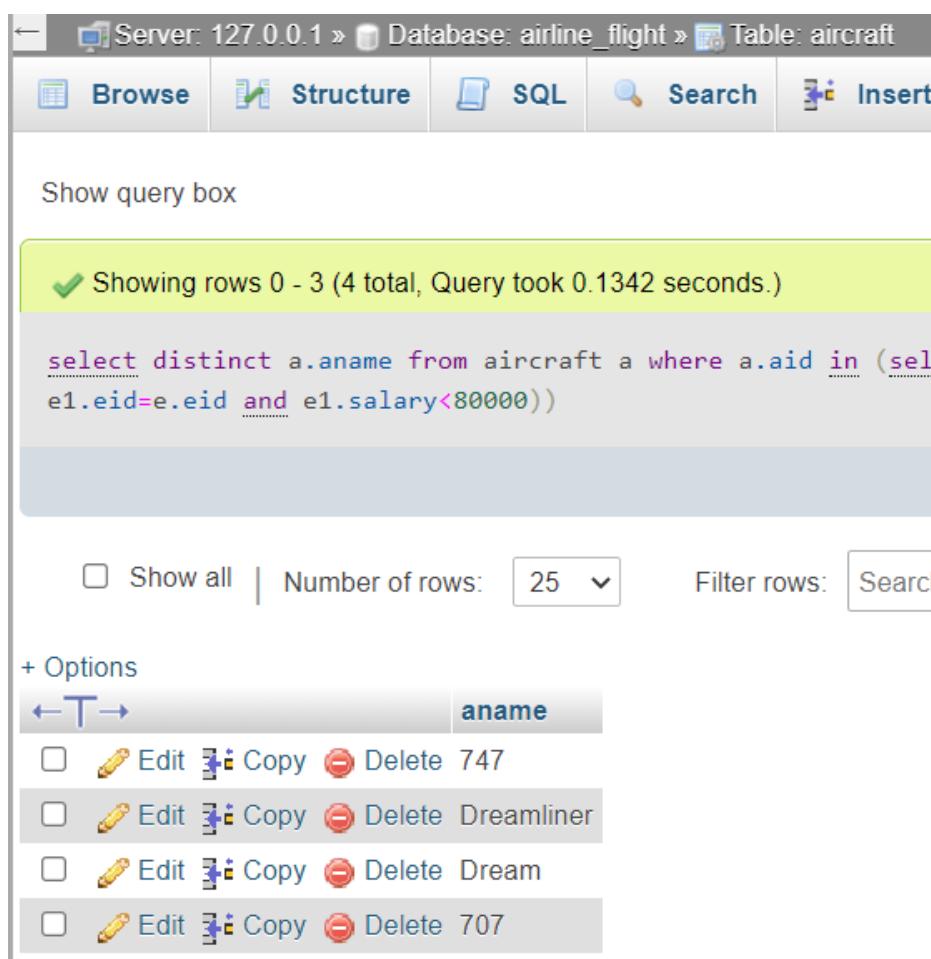
1) Find the names of aircraft such that all pilots certified to operate them have salaries more than Rs.80,000.

Query: select distinct a.aname from aircraft a where a.aid in (select c.aid from certified c, employees e where c.eid = e.eid and not exists (select * from employees e1 where e1.eid=e.eid and e1.salary<80000));



The screenshot shows the MySQL Workbench interface. The title bar indicates "Server: 127.0.0.1 » Database: airline_flight". The tabs at the top include "Structure", "SQL" (which is selected), "Search", "Query", "Export", "Import", "Operations", and "Privileges". Below the tabs is a search bar with the placeholder "Run SQL query/queries on database airline_flight: ?". The main area contains the following SQL code:

```
1 select distinct a.aname
2 from aircraft a
3 where a.aid in (select c.aid
4                   from certified c, employees e
5                   where c.eid = e.eid and not exists (select *
6                                         from employees e1
7                                         where e1.eid=e.eid and e1.salary<80000));
```



The screenshot shows the MySQL Workbench interface with the "Table" tab selected for the "aircraft" table. The title bar indicates "Server: 127.0.0.1 » Database: airline_flight » Table: aircraft". The tabs at the top are "Browse" (selected), "Structure", "SQL", "Search", and "Insert". Below the tabs, there is a message "Show query box". The main area displays the query results:

Showing rows 0 - 3 (4 total, Query took 0.1342 seconds.)

```
select distinct a.aname from aircraft a where a.aid in (select c.aid
e1.eid=e.eid and e1.salary<80000))
```

Below the results, there are options to "Show all" (unchecked), "Number of rows: 25", "Filter rows: Search", and a "+ Options" button. The results table has columns "T" and "aname". The data rows are:| T | aname |
| --- | --- |
| | Edit Copy Delete 747 |
| | Edit Copy Delete Dreamliner |
| | Edit Copy Delete Dream |
| | Edit Copy Delete 707 |

2) For each pilot who is certified for more than three aircrafts, find the eid and the maximum cruising range of the aircraft for which she or he is certified.

Query: select max(a.cruisingrange), c.eid from certified c, aircraft a where c.aid = a.aid group by c.eid having count(c.eid)>3;

Server: 127.0.0.1 » Database: airline_flight

Structure SQL Search Query Export

Run SQL query/queries on database airline_flight:

```
1 select max(a.cruisingrange), c.eid
2 from certified c, aircraft a
3 where c.aid = a.aid
4 group by c.eid
5 having count(c.eid)>3;
```

Server: 127.0.0.1 » Database: airline_flight » Table: certified

Browse Structure SQL Search Insert

Show query box

⚠ Current selection does not contain a unique column. Grid edit, checkboxes

✓ Showing rows 0 - 1 (2 total, Query took 0.0024 seconds.)

```
select max(a.cruisingrange), c.eid from certified c, aircraf
```

Show all | Number of rows: 25 Filter rows:

+ Options

max(a.cruisingrange)	eid
3500	701
120000	702

3) Find the names of pilots whose salary is less than the price of the cheapest route from Bengaluru to Frankfurt.

Query: select ename from employees where salary < (select min(price) from flights where fromplace='Bangalore' and toplace='Frankfurt');

The screenshot shows the MySQL Workbench interface with the SQL tab selected. The query window contains the following SQL code:

```
1 select ename
2 from employees
3 where salary < (select min(price)
4                   from flights
5                   where fromplace='Bangalore' and toplace='Frankfurt');
```

The screenshot shows the MySQL Workbench interface with the Structure tab selected for the employees table. The table structure is displayed below:

ename	job	mgr	hiredate	sal	comm	deptno
A	SA	10	1981-06-17	1600	0	5
E	CLERK	10	1981-06-17	300	0	5

Show query box

The screenshot shows the MySQL Workbench interface with the SQL tab selected. The results of the query execution are displayed in a green box:

Showing rows 0 - 1 (2 total, Query took 0.1026 seconds.)

```
select ename from employees where salary < (select min(price)
```

Below the results, there are options to show all rows, filter rows, and search. The results table shows two rows:

ename
A
E

+ Options

← → ename

Edit Copy Delete A

Edit Copy Delete E

4) For all aircraft with cruising range over 1000 Kms, find the name of the aircraft and the average salary of all pilots certified for this aircraft.

Query: select avg(e.salary), c.aid from certified c, employees e where c.aid in (select aid from aircraft where cruisingrange>1000) and e.eid = c.eid group by c.aid;

Server: 127.0.0.1 » Database: airline_flight

Structure SQL Search Query Export

Run SQL query/queries on database airline_flight:

```
1 select avg(e.salary), c.aid
2 from certified c, employees e
3 where c.aid in (select aid
4                 from aircraft
5                 where cruisingrange>1000)
6                 and e.eid = c.eid
7 group by c.aid;
```

Server: 127.0.0.1 » Database: airline_flight » Table: certified

Browse Structure SQL Search Insert

Show query box

⚠ Current selection does not contain a unique column. Grid edit, checkb

✓ Showing rows 0 - 4 (5 total, Query took 0.0441 seconds.)

select avg(e.salary), c.aid from certified c, employees e w

Show all | Number of rows: 25 ▾ Filter rows: Search

+ Options

avg(e.salary)	aid
75000.0000	101
113333.3333	104
50000.0000	105
50000.0000	106
113333.3333	107

5) Find the names of pilots certified for some Boeing aircraft.

Query: select ename from employees where eid in (select eid from certified where aid in (select aid from aircraft where fname = 'Boeing'));

Server: 127.0.0.1 » Database: airline_flight

Structure SQL Search Query Export Import

Run SQL query/queries on database airline_flight:

```
1 select ename
2 from employees
3 where eid in (select eid
4                 from certified
5                 where aid in (select aid
6                               from aircraft
7                               where fname = 'Boeing'));
```

Server: 127.0.0.1 » Database: airline_flight » Table: employees

Browse Structure SQL Search Insert

Show query box

Showing rows 0 - 0 (1 total, Query took 0.0017 seconds.)

select ename from employees where eid in (select eid from ce

Show all

Number of rows:

25

Filter rows:

Search

+ Options



ename



Edit Copy Delete A

6) Find the aids of all aircraft that can be used on routes from Bengaluru to Delhi.

Query: select fname from aircraft where cruisingrange > any (select distance from flights where fromplace='Bangalore' and toplace='Delhi');

The screenshot shows the MySQL Workbench interface with the SQL tab selected. The query window contains the following SQL code:

```
1 select fname
2 from aircraft
3 where cruisingrange > any (select distance
4                               from flights
5                               where fromplace='Bangalore' and toplace='Delhi');
```

The screenshot shows the MySQL Workbench interface with the results of the query execution. The message bar at the top says "Showing rows 0 - 3 (4 total, Query took 0.1056 seconds.)". The results table shows the following data:

fname
747
Dreamliner
Boeing
Dream

Show query box

The screenshot shows the MySQL Workbench interface with the results of the query execution in a modal dialog. The message bar at the top says "Showing rows 0 - 3 (4 total, Query took 0.1056 seconds.)". The results table shows the following data:

fname
747
Dreamliner
Boeing
Dream

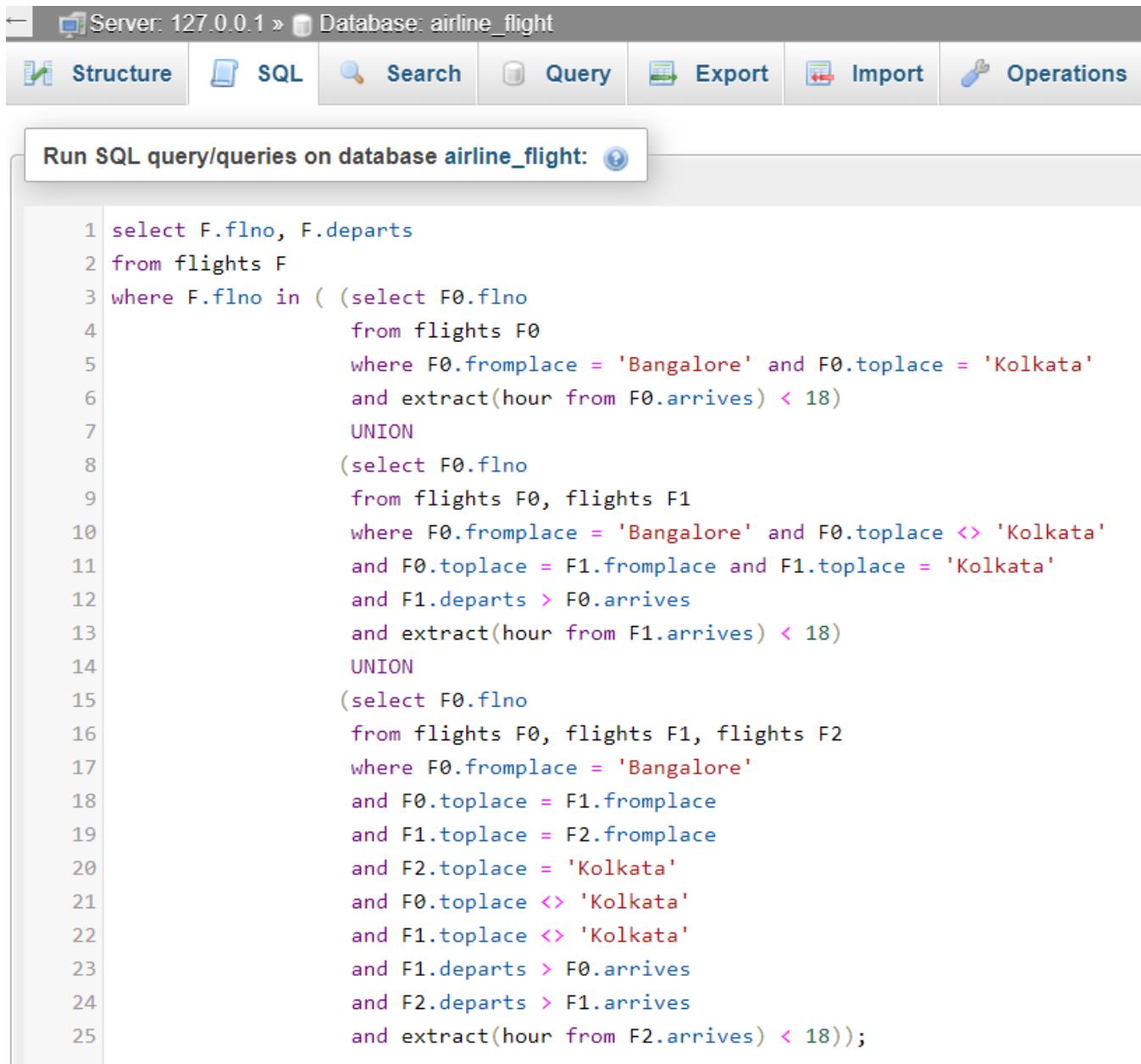
Show all | Number of rows: 25 Filter rows:

+ Options

← T →		fname
<input type="checkbox"/>		Edit
<input type="checkbox"/>		Copy
<input type="checkbox"/>		Delete
747		
<input type="checkbox"/>		Edit
<input type="checkbox"/>		Copy
<input type="checkbox"/>		Delete
Dreamliner		
<input type="checkbox"/>		Edit
<input type="checkbox"/>		Copy
<input type="checkbox"/>		Delete
Boeing		
<input type="checkbox"/>		Edit
<input type="checkbox"/>		Copy
<input type="checkbox"/>		Delete
Dream		

7) A customer wants to travel from Bangalore to Kolkata with no more than two changes of flight. List the choice of departure times from Bangalore if the customer wants to arrive in Kolkata by 6 p.m.

Query: SELECT F.flno, F.deperts FROM flights F WHERE F.flno IN ((SELECT F0.flno FROM flights F0 WHERE F0.fromplace = 'Bangalore' AND F0.toplace = 'Kolkata' AND extract(hour from F0.arrives) < 18) UNION (SELECT F0.flno FROM flights F0, flights F1 WHERE F0.fromplace = 'Bangalore' AND F0.toplace <> 'Kolkata' AND F0.toplace = F1.fromplace AND F1.toplace = 'Kolkata' AND F1.deperts > F0.arrives AND extract(hour from F1.arrives) < 18) UNION (SELECT F0.flno FROM flights F0, flights F1, flights F2 WHERE F0.fromplace = 'Bangalore' AND F0.toplace = F1.fromplace AND F1.toplace = F2.fromplace AND F2.toplace = 'Kolkata' AND F0.toplace <> 'Kolkata' AND F1.toplace <> 'Kolkata' AND F1.deperts > F0.arrives AND F2.deperts > F1.arrives AND extract(hour from F2.arrives) < 18));



The screenshot shows the MySQL Workbench interface. The top bar displays 'Server: 127.0.0.1 » Database: airline_flight'. Below the bar are tabs for 'Structure', 'SQL', 'Search', 'Query', 'Export', 'Import', and 'Operations'. The main area is a code editor with the following SQL query:

```
1 select F.flno, F.deperts
2 from flights F
3 where F.flno in ( (select F0.flno
4                     from flights F0
5                     where F0.fromplace = 'Bangalore' and F0.toplace = 'Kolkata'
6                     and extract(hour from F0.arrives) < 18)
7                     UNION
8                     (select F0.flno
9                     from flights F0, flights F1
10                    where F0.fromplace = 'Bangalore' and F0.toplace <> 'Kolkata'
11                    and F0.toplace = F1.fromplace and F1.toplace = 'Kolkata'
12                    and F1.deperts > F0.arrives
13                    and extract(hour from F1.arrives) < 18)
14                     UNION
15                     (select F0.flno
16                     from flights F0, flights F1, flights F2
17                     where F0.fromplace = 'Bangalore'
18                     and F0.toplace = F1.fromplace
19                     and F1.toplace = F2.fromplace
20                     and F2.toplace = 'Kolkata'
21                     and F0.toplace <> 'Kolkata'
22                     and F1.toplace <> 'Kolkata'
23                     and F1.deperts > F0.arrives
24                     and F2.deperts > F1.arrives
25                     and extract(hour from F2.arrives) < 18));
```

Server: 127.0.0.1 » Database: airline_flight » Table: flights

Browse Structure SQL Search Insert Export Import

Show query box

Showing rows 0 - 1 (2 total, Query took 0.0119 seconds.)

```
SELECT F.flno, F.deperts FROM flights F WHERE F.flno IN ( ( SELECT F0.flno FROM flights F0, flights F1 WHERE F0.flno < 18 ) UNION ( SELECT F0.flno FROM flights F0, flights F1 WHERE F0.flno > F1.flno AND F1.toplace = 'Kolkata' AND F1.deperts > F0.deperts AND extract(hour from F1.deperts) - extract(hour from F0.deperts) > 12 ) ) WHERE F0.fromplace = 'Bangalore' AND F0.toplace = F1.fromplace AND F1.toplace = F2.fromplace AND F1.deperts > F0.deperts AND F2.deperts > F1.deperts AND extract(hour from F2.deperts) - extract(hour from F1.deperts) > 12 ) )
```

Show all

Number of rows:

25

Filter rows:

Search this table

Sort by

+ Options

flno	departs
102	2013-05-05 07:15:31
106	2013-05-05 01:15:30