LAB 1:

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)

i. Create the above tables by properly specifying the primary keys and the foreign keys.

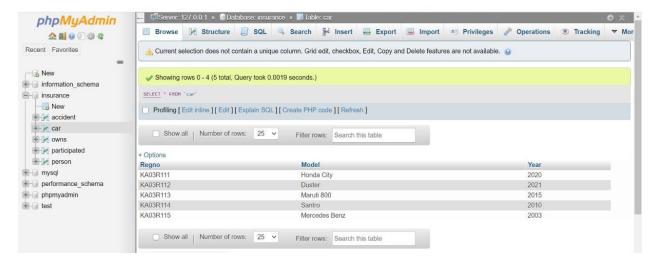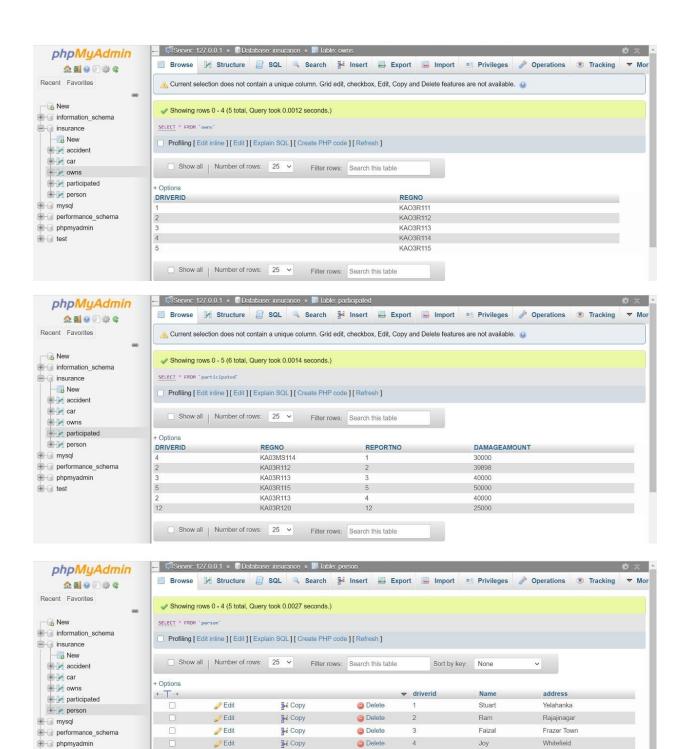ii. Enter at least five tuples for each relation.

iii.Demonstrate how you

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

25000.

b. Add a new accident to the database.

iv. Find the total number of people who owned cars that involved in accidents in 2008.

v. Find the number of accidents in which cars belonging to a specific model were involved.

Recent   Favorites

New
- information_schema
- insurance
  - New
  - accident
  - car
  - owns
  - participated
  - person
- mysql
- performance_schema
- phpmyadmin
- test

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

✓ Showing rows 0 - 4 (5 total, Query took 0.0012 seconds.)

`SELECT * FROM `owns``

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

DRIVERID	REGNO
1	KAO3R111
2	KAO3R112
3	KAO3R113
4	KAO3R114
5	KAO3R115

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

---

Recent   Favorites

New
- information_schema
- insurance
  - New
  - accident
  - car
  - owns
  - participated
  - person
- mysql
- performance_schema
- phpmyadmin
- test

⚠ 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 0.0014 seconds.)

`SELECT * FROM `participated``

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

DRIVERID	REGNO	REPORTNO	DAMAGEAMOUNT
4	KA03MS114	1	30000
2	KA03R112	2	39898
3	KA03R113	3	40000
5	KA03R115	5	50000
2	KA03R113	4	40000
12	KA03R120	12	25000

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

---

Recent   Favorites

New
- information_schema
- insurance
  - New
  - accident
  - car
  - owns
  - participated
  - person
- mysql
- performance_schema
- phpmyadmin
- test

✓ Showing rows 0 - 4 (5 total, Query took 0.0027 seconds.)

`SELECT * FROM `person``

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

				driverid	Name	address
☐	Edit	Copy	Delete	1	Stuart	Yelahanka
☐	Edit	Copy	Delete	2	Ram	Rajajinagar
☐	Edit	Copy	Delete	3	Faizal	Frazer Town
☐	Edit	Copy	Delete	4	Joy	Whitefield
☐	Edit	Copy	Delete	5	Rajesh	Jakkur

↑ ☐ Check all   With selected: Edit   Copy   Delete   Export

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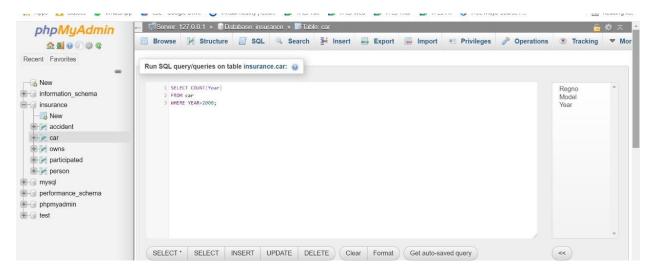
iii. a.

iii. b.

iv.

v.

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)

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

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

BORROWER (customer-name: String, loan-number: int)

i. Create the above tables by properly specifying the primary keys and the foreign keys.

ii. Enter at least five tuples for each relation.

iii. Find all the customers who have at least two accounts at the Main branch.

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

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

ACCOUNTNUMBER	BRANCHNAME	BALANCE
123	SAHAKARNAGAR	3000
124	JUBILEE HILLS	40000
125	BANSHANKARI	35000
126	CHAMRAJPET	20000
127	BANDRA	600
1	SBI_Chamrajpet	2000
2	SBI_ResidencyRoad	5000
3	SBI_ShivajiRoad	6000
4	SBI_ParliamentRoad	9000
5	SBI_JantarMantar	8000
6	SBI_ShivajiRoad	4000
8	SBI_ResidencyRoad	4000
9	SBI_ParliamentRoad	3000
10	SBI_ResidencyRoad	5000
11	SBI_JantarMantar	2000

+ Options

CUSTOMERNAME	LOANNUMBER
SUNIDHI	1
SHREYA	2
SONU	3
ALKA	4
UDIT	5
Avinash	1
Dinesh	2
Nikil	4
Ravi	5
Avinash	8
Nikil	9
Dinesh	10
Nikil	11

BRANCHNAME	BRANCHCITY	ASSETS
SAHAKARNAGAR	BENGALURU	300000
JUBILEE HILLS	HYDERABAD	400000
BANSHANKARI	BENGALURU	350000
CHAMRAJPET	HYDERABAD	200000
BANDRA	MUMBAI	600000
SBI_Chamrajpet	Bangalore	50000
SBI_ResidencyRoad	Bangalore	10000
SBI_ShivajiRoad	Bombay	20000
SBI_ParliamentRoad	Delhi	10000
SBI_JantarMantar	Delhi	20000

CUSTOMERNAME	CUSTOMERSTREET	CUSTOMERCITY
SRISHTI	SAHAKARNAGAR	BENGALURU
SHALINI	JUBILEE HILLS	HYDERABAD
GEORGE	BANSHANKARI	BENGALURU
RAJNI	CHAMRAJPET	HYDERABAD
MOHAN	BANDRA	MUMBAI
Avinash	Bull_Temple_Road	Bangalore
Dinesh	Bannerghatta_Road	Bangalore
Mohan	NationalCollege_Road	Bangalore
Nikil	Akbar_Road	Delhi
Ravi	Prithviraj_Road	Delhi

LOANNUMBER	BRANCHNAME	AMOUNT
1	SAHAKARNAGAR	2000
2	JUBILEE HILLS	800
3	BANSHANKARI	5000
4	CHAMRAJPET	3000
5	BANDRA	400
1	SBI_Chamrajpet	1000
2	SBI_ResidencyRoad	2000
3	SBI_ShivajiRoad	3000
4	SBI_ParliamentRoad	4000
5	SBI_Jantarmantar	5000

iii.

```
1  select C.customername
2  from depositor C
3  where exists (
4  select D.customername, count(D.customername)
5  from borrower D, accounts BA
6  where
7  D.LOANNUMBER = BA.ACCOUNTNUMBER AND
8  C.customername = D.customername AND
9  BA.branchname = 'SBI_ResidencyRoad'
10 group by D.customername
11 having count(D.customername)>=2);
```

```
select C.customername from depositor C where exists ( select D.customername, count(D.customername) from borrower D, accounts BA where
D.LOANNUMBER = BA.ACCOUNTNUMBER AND C.customername = D.customername AND BA.branchname = 'SBI_ResidencyRoad' group by D.customername
having count(D.customername)>=2)
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options
customername
Dinesh

iv.

Run SQL query/queries on table banking.accounts: ⊙

```
1 select BC.customername
2 from depositor BC
3 where not exists (
4 select BRANCHNAME from branch
5 where branchcity='Delhi');
```

```
select BC.customername from depositor BC where not exists ( select BRANCHNAME from branch where branchcity='Delhi')
```

v.

```
1 delete from accounts
2 where branchname IN (
3 select branchname
4 from branch
5 where branchcity='Bombay'
6 );
```

+ Options

ACCOUNTNUMBER	BRANCHNAME	BALANCE
123	SAHAKARNAGAR	3000
124	JUBILEE HILLS	40000
125	BANSHANKARI	35000
126	CHAMRAJPET	20000
127	BANDRA	600
1	SBI_Chamrajpet	2000
2	SBI_ResidencyRoad	5000
4	SBI_ParliamentRoad	9000
5	SBI_JantarMantar	8000
8	SBI_ResidencyRoad	4000
9	SBI_ParliamentRoad	3000
10	SBI_ResidencyRoad	5000
11	SBI_JantarMantar	2000

PROGRAM 3: SUPPLIER DATABASE

Consider the following schema:

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:

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

• Find the snames of suppliers who supply every part.

• Find the snames of suppliers who supply every red part.

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

• 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).

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

+ Options

SID	SNAME	ADDRESS
10001	Acme Widget	Bangalore
10002	Johns	Kolkata
10003	Vimal	Mumbai
10004	Reliance	Delhi

+ Options

PID	PNAME	COLOUR
20001	Book	Red
20002	Pen	Red
20003	Pencil	Green
20004	Mobile	Green
20005	Charger	Black

SID	PID	COST
10001	20001	10
10001	20002	10
10001	20003	30
10001	20004	10
10001	20005	10
10002	20001	10
10002	20002	20
10003	20003	30
10004	20003	40

- Find the pnames of parts for which there is some supplier:

QUERY: SELECT PNAME FROM PARTS,CATALOG WHERE CATALOG.PID=PARTS.PID AND CATALOG.SID IS NOT NULL;

✔ Showing rows 0 - 8 (9 total, Query took 0.0124 seconds.)

SELECT PNAME FROM PARTS,CATALOG WHERE CATALOG.PID=PARTS.PID AND CATALOG.SID IS NOT NULL

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

PNAME

Book

Pen

Pencil

Mobile

Charger

Book

Pen

Pencil

Pencil

- Find the snames of suppliers who supply every part:
  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));

✔ Showing rows 0 - 0 (1 total, Query took 0.0150 seconds.)

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))

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

SNAME

Acme Widget

- 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.COLOUR = 'Red') EXCEPT (SELECT C.PID FROM CATALOG C, PARTS P WHERE C.SID = S.SID AND C.PID = P.PID AND P.COLOUR = 'Red'));

- Find the pnames of parts supplied by Acme Widget Suppliers and by no one else.
  QUERY: 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');

- 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);

- For each part, find the sname of the supplier who charges the most for that part.
  QUERY: SELECT P.PID, S.SNAME FROM PARTS P, SUPPLIERS S, CATALOG C WHERE C.PID = P.PID
  AND C.SID = S.SID AND C.COST = (SELECT MAX (C1.COST) FROM CATALOG C1 WHERE C1.PID =
  P.PID);

Showing rows 0 - 5 (6 total, Query took 0.0035 seconds.)

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

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

PID	SNAME
20001	Acme Widget
20004	Acme Widget
20005	Acme Widget
20001	Johns
20002	Johns
20003	Reliance

PROGRAM 4. STUDENT FACULTY DATABASE

Consider the following database for student enrolment 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.

i. Find the names of all Juniors (level = JR) who are enrolled in a class taught by

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

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

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

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

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

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

← → C   ⓘ localhost/phpmyadmin/index.php?route=/database/structure&server=1&db=lp4_student

phpMyAdmin

Recent Favorites

- New
- information_schema
- lp3_supplier
- lp4_student
  - New
  - class
  - enrolled
  - faculty
  - student
- mysql
- performance_schema
- phpmyadmin
- test

← Server: 127.0.0.1 » Database: lp4_student

| Structure | SQL | Search | Query | Export | Import | Operations | Privileges | Routines | Events | Triggers | Tracking | Designer | Central columns |

Filters

Containing the word: [   ]

	Table ▲	Action							Rows	Type	Collation	Size	Overhead
☐	class	⭐	📋 Browse	🔧 Structure	🔍 Search	➕ Insert	🗑 Empty	⊘ Drop	8	InnoDB	utf8mb4_general_ci	32.0 KiB	-
☐	enrolled	⭐	📋 Browse	🔧 Structure	🔍 Search	➕ Insert	🗑 Empty	⊘ Drop	10	InnoDB	utf8mb4_general_ci	48.0 KiB	-
☐	faculty	⭐	📋 Browse	🔧 Structure	🔍 Search	➕ Insert	🗑 Empty	⊘ Drop	5	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	student	⭐	📋 Browse	🔧 Structure	🔍 Search	➕ Insert	🗑 Empty	⊘ Drop	6	InnoDB	utf8mb4_general_ci	16.0 KiB	-
	4 tables	Sum							29	InnoDB	utf8mb4_general_ci	112.0 KiB	0 B

↑ ☐ Check all   With selected: [▼]

🖨 Print 📖 Data dictionary

Create table

Name: [   ]   Number of columns: [4]

Go

Console

🔍 Search   31°C Partly sunny   ENG   16:36   23-05-2021

phpMyAdmin

Recent Favorites

- New
- information_schema
- lp3_supplier
- lp4_student
  - New
  - class
  - enrolled
  - faculty
  - student
- mysql
- performance_schema
- phpmyadmin
- test

Server: 127.0.0.1 » Database: lp4_student » Table: student

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

✓ Showing rows 0 - 5 (6 total, Query took 0.0006 seconds.)

SELECT * FROM `student`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

				snum	sname	major	lvl	age
☐	Edit	Copy	Delete	1	John	CS	Sr	19
☐	Edit	Copy	Delete	2	Smith	CS	Jr	20
☐	Edit	Copy	Delete	3	Jacob	CV	Sr	20
☐	Edit	Copy	Delete	4	Tom	CS	Jr	20
☐	Edit	Copy	Delete	5	Rahul	CS	Jr	20
☐	Edit	Copy	Delete	6	Rita	CS	Sr	21

↑ ☐ Check all With selected: Edit Copy Delete Export

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

Query results operations

Print Copy to clipboard Export Display chart Create view

Bookmark this SQL query

Label: [   ] ☐ Let every user access this bookmark

Bookmark this SQL query

Console

localhost/phpmyadmin/index.php?route=/sql&server=1&db=lp4_student&table=faculty&pos=0

phpMyAdmin

Recent Favorites

- New
- information_schema
- lp3_supplier
- lp4_student
  - New
  - class
  - enrolled
  - faculty
  - student
- mysql
- performance_schema
- phpmyadmin
- test

Server: 127.0.0.1 » Database: lp4_student » Table: faculty

| Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations | Tracking | Triggers |

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

SELECT * FROM `faculty`

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

				▼ fid	fname	deptid
☐	✎ Edit	⌷ Copy	⊖ Delete	11	Harish	1000
☐	✎ Edit	⌷ Copy	⊖ Delete	12	MV	1000
☐	✎ Edit	⌷ Copy	⊖ Delete	13	Mira	1001
☐	✎ Edit	⌷ Copy	⊖ Delete	14	Shiva	1002
☐	✎ Edit	⌷ Copy	⊖ Delete	15	Nupur	1000

↑ ☐ Check all   With selected: ✎ Edit ⌷ Copy ⊖ Delete ⎘ Export

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

Query results operations

🖨 Print   ⌷ Copy to clipboard   ⎘ Export   📊 Display chart   Create view

🔖 Bookmark this SQL query

Label: [   ]   ☐ Let every user access this bookmark

Bookmark this SQL query

localhost/phpmyadmin/index.php?route=/database/structure&server=1&db=performance_schema

Search O ... 31°C Partly sunny ENG 16:37 23-05-2021

phpMyAdmin

Recent Favorites

- New
- information_schema
- lp3_supplier
- lp4_student
  - New
  - class
  - enrolled
  - faculty
  - student
- mysql
- performance_schema
- phpmyadmin
- test

Server: 127.0.0.1 » Database: lp4_student » Table: enrolled

| Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations | Tracking | Triggers |

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

✔ Showing rows 0 - 9 (10 total, Query took 0.0006 seconds.)

SELECT * FROM `enrolled`

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

snum	cname
1	class1
2	class1
3	class3
4	class3
5	class4
1	class5
2	class5
3	class5
4	class5
5	class5

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

Query results operations

🖨 Print   ⧉ Copy to clipboard   📄 Export   📊 Display chart   📋 Create view

Bookmark this SQL query

Label: [   ]   ☐ Let every user access this bookmark

Bookmark this SQL query

Console

phpMyAdmin

Recent Favorites

- New
- information_schema
- lp3_supplier
- lp4_student
  - New
  - class
  - enrolled
  - faculty
  - student
- mysql
- performance_schema
- phpmyadmin
- test

Server: 127.0.0.1 » Database: lp4_student » Table: class

| Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations | Tracking | Triggers |

✔ Showing rows 0 - 7 (8 total, Query took 0.0006 seconds.)

SELECT * FROM `class`

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

				▼ cname	meets_at	room	fid
☐	🖉 Edit	Copy	⊘ Delete	class1	2012-11-15 10:15:16	R1	14
☐	🖉 Edit	Copy	⊘ Delete	class10	2012-11-15 10:15:16	R128	14
☐	🖉 Edit	Copy	⊘ Delete	class2	2012-11-15 10:15:20	R2	12
☐	🖉 Edit	Copy	⊘ Delete	class3	2012-11-15 10:15:25	R3	11
☐	🖉 Edit	Copy	⊘ Delete	class4	2012-11-15 20:15:20	R4	14
☐	🖉 Edit	Copy	⊘ Delete	class5	2012-11-15 20:15:20	R3	15
☐	🖉 Edit	Copy	⊘ Delete	class6	2012-11-15 13:20:20	R2	14
☐	🖉 Edit	Copy	⊘ Delete	class7	2012-11-15 10:10:10	R3	14

↑ ☐ Check all With selected: 🖉 Edit Copy ⊘ Delete Export

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

Query results operations

🖨 Print Copy to clipboard Export Display chart Create view

Bookmark this SQL query

Label: _____ ☐ Let every user access this bookmark

Bookmark this SQL query

Console

PROGRAM 5. AIRLINE FLIGHT DATABASE

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

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

CERTIFIED (eid: integer, aid: integer)

EMPLOYEE (eid: integer, ename: 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.

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

ii. 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.

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

Frankfurt.

iv. 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.

v. Find the names of pilots certified for some Boeing aircraft.

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

vii. A customer wants to travel from Madison to New York with no more than two changes of flight. List the

choice of departure times from Madison if the customer wants to arrive in New York by 6 p.m.

viii. Print the name and salary of every non-pilot whose salary is more than the average salary for pilots.

phpMyAdmin

Recent | Favorites

- New
- banking
- flights
  - New
  - aircraft
  - certified
  - employee
  - flights
- information_schema
- mysql
- performance_schema
- phpmyadmin
- project_work
- student_faculty
- supplier
- test

Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations | Tracking | Triggers

Showing rows 0 - 5 (6 total, Query took 0.0007 seconds.)

SELECT * FROM `flights`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

			flno	from_	to_	distance	departs	arrives	price
Edit	Copy	Delete	101	Bangalore	Delhi	2500	2005-05-13 07:15:31	2005-05-13 17:15:31	5000
Edit	Copy	Delete	102	Bangalore	Lucknow	3000	2005-05-13 07:15:31	2005-05-13 11:15:31	6000
Edit	Copy	Delete	103	Lucknow	Delhi	500	2005-05-13 12:15:31	2005-05-13 17:15:31	3000
Edit	Copy	Delete	104	Bangalore	Frankfurt	8500	2005-05-13 07:15:31	2005-05-13 23:15:31	75000
Edit	Copy	Delete	105	Kolkata	Delhi	3400	2005-05-13 07:15:31	2005-05-13 09:15:31	7000
Edit	Copy	Delete	107	Bangalore	Frankfurt	8000	2005-05-13 07:15:31	2005-05-13 22:15:31	60000

Check all | With selected: Edit | Copy | Delete | Export

phpMyAdmin

Recent | Favorites

- New
- banking
- flights
  - New
  - aircraft
  - certified
  - employee
  - flights
- information_schema
- mysql
- performance_schema
- phpmyadmin
- project_work
- student_faculty
- supplier
- test

Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations | Tracking | Triggers

Showing rows 0 - 6 (7 total, Query took 0.0012 seconds.)

SELECT * FROM `employee`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

			eid	ename	salary
Edit	Copy	Delete	701	A	50000
Edit	Copy	Delete	702	B	100000
Edit	Copy	Delete	703	C	150000
Edit	Copy	Delete	704	D	90000
Edit	Copy	Delete	705	E	40000
Edit	Copy	Delete	706	F	60000
Edit	Copy	Delete	707	G	90000

Check all | With selected: Edit | Copy | Delete | Export

phpMyAdmin

Recent | Favorites

- New
- banking
- flights
  - New
  - aircraft
  - certified
  - employee
  - flights
- information_schema
- mysql
- performance_schema
- phpmyadmin
- project_work
- student_faculty
- supplier
- test

Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations | Tracking | Triggers

SELECT * FROM `certified`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

			eid	aid
Edit	Copy	Delete	701	101
Edit	Copy	Delete	701	102
Edit	Copy	Delete	701	105
Edit	Copy	Delete	701	106
Edit	Copy	Delete	702	101
Edit	Copy	Delete	702	104
Edit	Copy	Delete	702	107
Edit	Copy	Delete	703	104
Edit	Copy	Delete	703	105
Edit	Copy	Delete	703	107
Edit	Copy	Delete	704	104
Edit	Copy	Delete	704	105
Edit	Copy	Delete	704	107
Edit	Copy	Delete	705	103

Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations | Tracking | Triggers

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

SELECT * FROM `aircraft`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

				aid	aname	cruisingrange
☐	✎ Edit	⊞ Copy	⊝ Delete	101	747	3000
☐	✎ Edit	⊞ Copy	⊝ Delete	102	Boeing	900
☐	✎ Edit	⊞ Copy	⊝ Delete	103	647	800
☐	✎ Edit	⊞ Copy	⊝ Delete	104	Dreamliner	10000
☐	✎ Edit	⊞ Copy	⊝ Delete	105	Boeing	3500
☐	✎ Edit	⊞ Copy	⊝ Delete	106	707	1500
☐	✎ Edit	⊞ Copy	⊝ Delete	107	Dream	120000

↑   ☐ Check all   With selected: ✎ Edit   ⊞ Copy   ⊝ Delete   Export

1)

Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations | Tracking | Triggers

Show query box

Showing rows 0 - 3 (4 total, Query took 0.0420 seconds.)

select distinct a.aname from aircraft a , employee e , certified c where a.aid = c.aid and e.eid = c.eid and e.salary >= 80000

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

				aname
☐	✎ Edit	⊞ Copy	⊝ Delete	747
☐	✎ Edit	⊞ Copy	⊝ Delete	Dreamliner
☐	✎ Edit	⊞ Copy	⊝ Delete	Boeing
☐	✎ Edit	⊞ Copy	⊝ Delete	Dream

↑   ☐ Check all   With selected: ✎ Edit   ⊞ Copy   ⊝ Delete   Export

2)

Showing rows 0 - 0 (1 total, Query took 0.1144 seconds.)

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

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

				eid	max(a.cruisingrange)
☐	✎ Edit	⊞ Copy	⊝ Delete	701	3500

↑   ☐ Check all   With selected: ✎ Edit   ⊞ Copy   ⊝ Delete   Export

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

3)

✅ Showing rows 0 - 1 (2 total, Query took 0.0026 seconds.)

```sql
select e.ename from employee e where e.salary < (select min(price) from flights where from_ = 'Bangalore' and to_ = 'Frankfurt')
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options
←T→				ename
☐	🖉 Edit	📋 Copy	⊝ Delete	A
☐	🖉 Edit	📋 Copy	⊝ Delete	E

↥ ☐ Check all   With selected: 🖉 Edit 📋 Copy ⊝ Delete 🖷 Export

4)

✅ Showing rows 0 - 4 (5 total, Query took 0.0396 seconds.)

```sql
select a.aname , avg(e.salary) from aircraft a , employee e , certified c where a.aid = c.aid and e.eid = c.eid and a.cruisingrange > 1000 group by a.aname
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options
←T→				aname	avg(e.salary)
☐	🖉 Edit	📋 Copy	⊝ Delete	707	50000.0000
☐	🖉 Edit	📋 Copy	⊝ Delete	747	75000.0000
☐	🖉 Edit	📋 Copy	⊝ Delete	Boeing	96666.6667
☐	🖉 Edit	📋 Copy	⊝ Delete	Dream	113333.3333
☐	🖉 Edit	📋 Copy	⊝ Delete	Dreamliner	113333.3333

↥ ☐ Check all   With selected: 🖉 Edit 📋 Copy ⊝ Delete 🖷 Export

5)

✅ Showing rows 0 - 2 (3 total, Query took 0.0043 seconds.)

```sql
select distinct e.ename from employee e , aircraft a , certified c where a.aid = c.aid and e.eid = c.eid and a.aname = 'Boeing'
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options
←T→				ename
☐	🖉 Edit	📋 Copy	⊝ Delete	A
☐	🖉 Edit	📋 Copy	⊝ Delete	C
☐	🖉 Edit	📋 Copy	⊝ Delete	D

↥ ☐ Check all   With selected: 🖉 Edit 📋 Copy ⊝ Delete 🖷 Export

6)

✅ Showing rows 0 - 1 (2 total, Query took 0.0047 seconds.)

```sql
select a.aid from aircraft a where a.cruisingrange >= (select min(distance) from flights where from_ = 'Bangalore' and to_ = 'Frankfurt')
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options
←T→				aid
☐	🖉 Edit	📋 Copy	⊝ Delete	104
☐	🖉 Edit	📋 Copy	⊝ Delete	107

↥ ☐ Check all   With selected: 🖉 Edit 📋 Copy ⊝ Delete 🖷 Export

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

7)

```
select F.departs from Flights F where F.flno IN ( ( select F0.flno from Flights F0 where F0.from_ = 'Bangalore' and F0.to_ = 'Delhi' and
extract(hour from F0.arrives) < 18 ) union ( select F0.flno from Flights F0, Flights F1 where F0.from_ = 'Bangalore' AND F0.to_ <> 'Delhi' and
F0.to_ = F1.from_ AND F1.to_ = 'Delhi' and F1.departs > F0.arrives and extract(hour from F1.arrives) < 18) union ( select F0.flno from Flights
F0, Flights F1, Flights F2 where F0.from_ = 'Bangalore' and F0.to_ = F1.from_ and F1.to_ = F2.from_ and F2.to_ = 'Delhi' and F0.to_ <> 'Delhi'
and F1.to_ <> 'Delhi' and F1.departs > F0.arrives and F2.departs > F1.arrives and extract(hour from F2.arrives) < 18))
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

←T→				departs
☐	🖉 Edit	⌷ Copy	⊖ Delete	2005-05-13 07:15:31
☐	🖉 Edit	⌷ Copy	⊖ Delete	2005-05-13 07:15:31

↥ ☐ Check all   With selected: 🖉 Edit ⌷ Copy ⊖ Delete 🖳 Export

8)

Show query box

```
select e.ename , e.salary from employee e where e.salary > (select avg(distinct e1.salary) from employee e1 , certified c where e1.eid = c.eid)
and e.eid not in (select distinct eid from certified)
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

←T→				ename	salary
☐	🖉 Edit	⌷ Copy	⊖ Delete	G	90000

↥ ☐ Check all   With selected: 🖉 Edit ⌷ Copy ⊖ Delete 🖳 Export