

Lab Record

PROGRAM 6. ORDER PROCESSING DATABASE

Consider the following relations for an Order Processing database application in a company.

CUSTOMER (CUST #: int, cname: String, city: String)

ORDER (order #: int, odate: date, cust #: int, ord-Amt: int)

ITEM (item #: int, unit-price: int)

ORDER-ITEM (order #: int, item #: int, qty: int)

WAREHOUSE (warehouse #: int, city: String)

SHIPMENT (order #: int, warehouse #: int, ship-date: date)

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

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> customer	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/> item	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/> order1	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
<input type="checkbox"/> orderitem	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 KiB	-
<input type="checkbox"/> shipment	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 KiB	-
<input type="checkbox"/> warehouse	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
6 tables	Sum	0	InnoDB	utf8mb4_general_ci	176.0 KiB	0 B

2. Enter at least five tuples for each relation.

✓ 6 rows inserted. (Query took 0.1476 seconds.)

```
INSERT INTO `customer`(`cust_id`, `cname`, `city`) VALUES (1,'karan','Bangalore'), (2,'varun','Bangalore'), (3,'Shashank','Delhi'), (4,'Jon','Delhi'), (5,'Ria','Mumbai'), (6,'Balaji','Chennai')
```

[\[Edit inline\]](#) [\[Edit\]](#) [\[Create P](#)

✓ 10 rows inserted. (Query took 0.1438 seconds.)

```
INSERT INTO `item`(`iid`, `price`) VALUES (1,1200), (2,500), (3,250), (4,700), (5,900), (6,600), (7,1200), (8,600), (9,1000), (10,100)
```

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

✓ 10 rows inserted. (Query took 0.1377 seconds.)

```
INSERT INTO `orders`(`oid`, `odate`, `cid`, `oamt`) VALUES (1,'2021-03-05',1,2000), (2,'2021-03-08',1,1600), (3,'2021-03-10',3,1500), (4,'2021-03-09',2,2200), (5,'2021-03-06',3,3000), (6,'2021-03-08',4,2500), (7,'2021-03-12',4,4000), (8,'2021-03-06',5,3000), (9,'2021-03-11',1,1500), (10,'2021-03-12',5,3000)
```

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

✓ 16 rows inserted. (Query took 0.1315 seconds.)

```
INSERT INTO `orderitem`(`oid`, `iid`, `qty`) VALUES (1,9,2),(2,9,1),(2,8,1),(3,2,3),(4,7,1),(4,3,4),(5,10,10),(6,2,3),(6,3,2),(7,9,4),(8,1,1),(8,7,1),(8,8,1),(9,8,1),(9,5,1),(10,2,6)
```

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

✓ 6 rows inserted. (Query took 0.1420 seconds.)

```
INSERT INTO `warehouse`(`war`, `city`) VALUES (1,'Mumbai'), (2,'Bangalore'), (3,'Chennai'), (4,'Delhi'), (5,'Kolkata'), (6,'Pune')
```

✓ 10 rows inserted. (Query took 0.0826 seconds.)

```
INSERT INTO `shipment`(`oid`, `war`, `sdate`) VALUES (1,1,'2021-03-10'),(2,3,'2021-03-11'),(3,6,'2021-03-21'),(4,2,'2021-03-10'),(5,2,'2021-03-17'),(6,4,'2021-03-10'),(7,5,'2021-03-22'),(8,6,'2021-03-07'),(9,5,'2021-03-16'),(10,1,'2021-03-13')
```

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

3. Produce a listing: CUSTNAME, #oforders, AVG_ORDER_AMT, where the middle column is the total numbers of orders by the customer and the last column is the average order amount for that customer.

✓ Showing rows 0 - 4 (5 total, Query took 0.0033 seconds.) [cname: JON... - VARUN...]

```
SELECT c.cname, COUNT(o.oid) AS oforders, AVG(o.oamt) FROM customer c,orders o WHERE c.cid=o.cid group by c.cname order BY c.cname ASC
```

☐ Profiling [\[Edit\]](#)

☐ Show all | Number of rows: 25 | Filter rows:

+ Options

cname	oforders	AVG(o.oamt)
Jon	2	3250.0000
karan	3	1700.0000
Ria	2	3000.0000
Shashank	2	2250.0000
varun	1	2200.0000

4. List the order# for orders that were shipped from all warehouses that the company has in a specific city.

Showing rows 0 - 0 (1 total, Query took 0.0057 seconds.)

```
SELECT distinct oid FROM `shipment` s1 WHERE not EXISTS ((select war from warehouse WHERE city='Bangalore') EXCEPT (SELECT war FROM shipment s2 WHERE s1.oid=s2.oid))
```

☐ Profiling [\[Edit inline\]](#) [\[Edit\]](#) [\[Explain SQL\]](#) [\[Create PHP code\]](#) [\[Refresh\]](#)

☐ Show all | Number of rows: 25 | Filter rows:

+ Options

oid
4

Showing rows 0 - 0 (1 total, Query took 0.0057 seconds.)

5. Demonstrate how you delete item# 10 from the ITEM table and make that field null in the ORDER_ITEM table.

+ Options		
oid	iid	qty
1	9	2
2	9	1
2	8	1
3	2	3
4	7	1
4	3	4
5	10	10
6	2	3
6	3	4
7	9	4
8	1	1
8	7	1
8	8	1
9	8	1
9	NULL	1
10	2	6
5	9	2

oid	iid	qty
1	9	2
2	9	1
2	8	1
3	2	3
4	7	1
4	3	4
5	9	2
5	10	10
6	2	3
6	3	4
7	9	4
8	1	1
8	7	1
8	8	1
9	8	1
9	5	1
10	2	6

PROGRAM 7. BOOK DEALER DATABASE

The following tables are maintained by a book dealer:

AUTHOR(author-id: int, name: String, city: String, country: String)

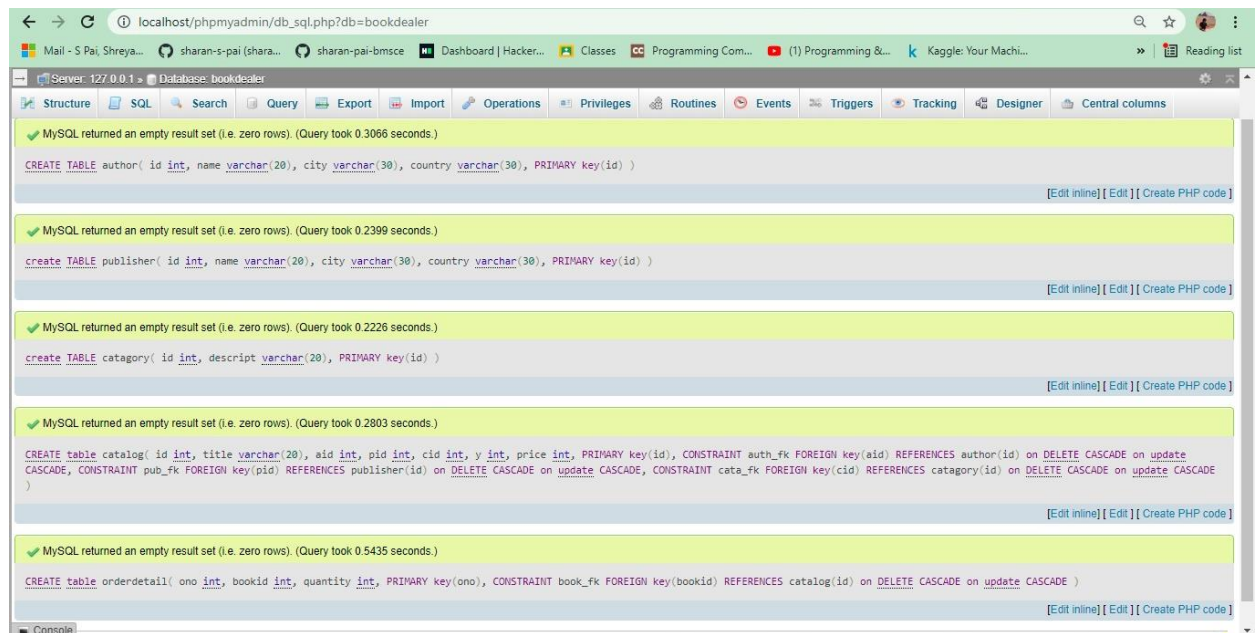
PUBLISHER(publisher-id: int, name: String, city: String, country: String)

CATALOG (book-id: int, title: String, author-id: int, publisher-id: int, category-id: int, year: int, price: int)

CATEGORY(category-id: int, description: String)

ORDER-DETAILS(order-no: int, book-id: int, quantity: int)

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



```
CREATE TABLE author( id int, name varchar(20), city varchar(30), country varchar(30), PRIMARY key(id) )

create TABLE publisher( id int, name varchar(20), city varchar(30), country varchar(30), PRIMARY key(id) )

create TABLE category( id int, descript varchar(20), PRIMARY key(id) )

CREATE table catalog( id int, title varchar(20), aid int, pid int, cid int, y int, price int, PRIMARY key(id), CONSTRAINT auth_fk FOREIGN key(aid) REFERENCES author(id) on DELETE CASCADE on update CASCADE, CONSTRAINT pub_fk FOREIGN key(pid) REFERENCES publisher(id) on DELETE CASCADE on update CASCADE, CONSTRAINT cata_fk FOREIGN key(cid) REFERENCES category(id) on DELETE CASCADE on update CASCADE )

CREATE table orderdetail( ono int, bookid int, quantity int, PRIMARY key(ono), CONSTRAINT book_fk FOREIGN key(bookid) REFERENCES catalog(id) on DELETE CASCADE on update CASCADE )
```

2. Enter at least five tuples for each relation.



```
INSERT INTO `author`(`id`, `name`, `city`, `country`) VALUES (1,'George RR Martin','London','Britain'), (2,'Veronica Roth','New York','USA'), (3,'Cassandra Clare','California','USA'), (4,'Sanjaya Baru','New Delhi','India'), (5,'JK Rowling','London','Britain')

INSERT INTO `publisher`(`id`, `name`, `city`, `country`) VALUES (1,'Bantam Spectra','London','UK'), (2,'Bloombury Publication','London','UK'), (3,'Penguin Publication','New Delhi','India'), (4,'Scholastic','New York','USA'), (5,'Pearson','California','USA')
```

Warning: #1265 Data truncated for column 'name' at row 2

Show query box

✓ 5 rows inserted. (Query took 0.1340 seconds.)

```
INSERT INTO `category`(`id`, `descript`) VALUES (1,'Sci-Fic'), (2,'Fiction'), (3,'Fantasy'), (4,'Documentary'), (5,'Auto biography')
```

Show query box

✓ 7 rows inserted. (Query took 0.1400 seconds.)

```
INSERT INTO `catalog`(`id`, `title`, `aid`, `pid`, `cid`, `y`, `price`) VALUES (1,'Harry Potter Order of Phenoix',5,2,2,2010,600), (2,'Game Of Thrones',1,1,1,1997,800), (3,'The Accidental PM',4,3,5,2014,400), (4,'Divergent',2,4,2,2010,500), (5,'Insurgent',2,4,2,2012,500), (6,'The Red Wedding',1,1,1,2004,900), (7,'Harry Potter Goblet of Fire',5,2,2,2007,400)
```

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

⚠ Warning: #1265 Data truncated for column 'title' at row 1

⚠ Warning: #1265 Data truncated for column 'title' at row 7

Show query box

✓ 5 rows inserted. (Query took 0.0976 seconds.)

```
INSERT INTO `orderdetail`(`ono`, `bookid`, `quantity`) VALUES (1,1,4), (2,7,6), (3,3,10), (4,4,2), (5,2,14)
```

- Give the details of the authors who have 2 or more books in the catalog and the price of the books in the catalog and the year of publication is after 2000.

✓ Showing rows 0 - 1 (2 total, Query took 0.0414 seconds.)

```
select * from author WHERE id IN (SELECT aid from catalog WHERE y>2000 group by aid having count(*)>=2)
```

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

+ Options

	id	name	city	country
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	Veronica Roth	New York	USA
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	JK Rowling	London	Britain

- Find the author of the book which has maximum sales.

✓ Showing rows 0 - 1 (2 total, Query took 0.0041 seconds.)

```
SELECT * FROM `author` WHERE id in ( SELECT aid FROM catalog c WHERE id in (SELECT bookid FROM orderdetail WHERE quantity = (SELECT MAX(quantity) FROM orderdetail)))
```

☐ Profiling [\[Edit inline\]](#) [\[Edit \]](#) [\[Explain \]](#)

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

+ Options

	id	name	city	country
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	George RR Martin	London	Britain
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	Sanjaya Baru	New Delhi	India

5. Demonstrate how you increase the price of books published by a specific publisher by 10%.

✓ Showing rows 0 - 6 (7 total, Query took 0.0009 seconds.)

```
SELECT price from catalog
```

☐ Show all | Number of rows: 25

+ Options

	price
<input type="checkbox"/> Edit Copy Delete	600
<input type="checkbox"/> Edit Copy Delete	800
<input type="checkbox"/> Edit Copy Delete	400
<input type="checkbox"/> Edit Copy Delete	500
<input type="checkbox"/> Edit Copy Delete	500
<input type="checkbox"/> Edit Copy Delete	900
<input type="checkbox"/> Edit Copy Delete	400

✓ 7 rows affected. (Query took 0.0622 seconds.)

```
UPDATE catalog SET price=1.1*price
```

✓ Showing rows 0 - 6 (7 total, Query took 0.0009 seconds)

```
SELECT price from catalog
```

☐ Show all | Number of rows: 25 Filter r

+ Options

	price
<input type="checkbox"/> Edit Copy Delete	660
<input type="checkbox"/> Edit Copy Delete	880
<input type="checkbox"/> Edit Copy Delete	440
<input type="checkbox"/> Edit Copy Delete	550
<input type="checkbox"/> Edit Copy Delete	550
<input type="checkbox"/> Edit Copy Delete	990
<input type="checkbox"/> Edit Copy Delete	440

PROGRAM 8. STUDENT ENROLLMENT DATABASE

Consider the following database of student enrollment in courses and books adopted for each course.

STUDENT (regno: String, name: String, major: String, bdate: date)

COURSE (course #: int, cname: String, dept: String)

ENROLL (regno: String, cname: String, sem: int, marks: int)

BOOK_ADOPTION (course #: int, sem: int, book-ISBN: int)

TEXT(book-ISBN:int, book-title:String, publisher:String, author:String)

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

Show query box:

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.1863 seconds.)

```
CREATE TABLE TextBook( isbn int, title varchar(20), pub varchar(20), auth varchar(20), PRIMARY key(isbn) )
```

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

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.2995 seconds.)

```
CREATE TABLE Book( cid int, sem int, isbn int, constraint c_fk FOREIGN key(cid) REFERENCES course(cid) ON DELETE CASCADE ON UPDATE CASCADE, constraint isbn_fk FOREIGN key(isbn) REFERENCES TextBook(isbn) ON DELETE CASCADE ON UPDATE CASCADE )
```

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

Table	Action	Rows	Type	Collation	Size	Overhead
book	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 KiB	-
course	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
enroll	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 KiB	-
student	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
textbook	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
5 tables	Sum	0	InnoDB	utf8mb4_general_ci	160.0 KiB	0 B

↑ ☐ Check all With selected: ▼

2. Enter at least five tuples for each relation.

✓ 6 rows inserted. (Query took 0.1174 seconds.)

```
INSERT INTO `student`(`regno`, `name`, `major`, `bdate`) VALUES ('1','Robert','SR','2001-04-28'), ('2','John','JR','2004-05-28'), ('3','Sheldon','SR','2000-12-25'), ('4','Lennard','INTR','2003-06-18'), ('5','Howard','JR','2005-01-02'), ('6','Penny','JR','2004-08-20')
```

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

✓ 6 rows inserted. (Query took 0.1297 seconds.)

```
INSERT INTO `course`(`cid`, `cname`, `dept`) VALUES (1,'ADA','CSE'), (2,'DBMS','CSE'), (3,'EED','Mech'), (4,'EME','Mech'), (5,'MPMC','ECE'), (6,'DAE','ECE')
```

✓ 7 rows inserted. (Query took 0.1474 seconds.)

```
INSERT INTO `enroll`(`regno`, `cname`, `sem`, `marks`) VALUES ('1','ADA',4,72), ('1','MPMC',3,35), ('2','EED',2,39), ('3','ADA',4,99), ('3','DAE',4,100), ('3','DBMS',4,98), ('4','DBMS',4,89)
```

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

6 rows inserted. (Query took 0.1717 seconds.)

```
INSERT INTO `textbook` (`isbn`, `title`, `pub`, `auth`) VALUES (1,'Algorithm Design','Pearson','Levlin'), (2,'Database System','Pearson','Elmasri'), (3,'Engg Drawing','Scholastic','Antonio'), (4,'Machine Interior','Bloomberg','John'), (5,'Electric Circuits','Pearson','Varun'), (6,'8086 Microprocessor','Bloomberg','Burchandi')
```

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

6 rows inserted. (Query took 0.1743 seconds.)

```
INSERT INTO `book` (`cid`, `sem`, `isbn`) VALUES ('2',3,7), ('2',4,2), ('1',3,1), ('1',4,1), ('3',3,3), ('4',1,4)
```

3. Demonstrate how you add a new text book to the database and make this book be adopted by some department.

Show query box

1 row inserted. (Query took 0.1768 seconds.)

```
insert INTO textbook VALUES (8,'Analysis of Complex Algo','Pearson','Krishnamurthy')
```

Warning: #1265 Data truncated for column 'title' at row 1

1 row inserted. (Query took 0.0400 seconds.)

```
INSERT into book VALUES (1,4,8)
```

4. Produce a list of text books (include Course #, Book-ISBN, Book-title) in the alphabetical order for courses offered by the 'CS' department that use more than two books.

Showing rows 0 - 4 (5 total, Query took 0.0078 seconds.) [cname: ADA... - DBMS...]

```
SELECT DISTINCT c.cid,c.cname,b.sem,b.isbn,t.title FROM course c,book b,textbook t WHERE c.cid=b.cid AND b.isbn=t.isbn AND c.dept='CSE' AND c.cid IN (SELECT cid FROM book group by cid having COUNT(isbn)>=2) ORDER BY c.cname
```

☐ Profiling [\[Edit inline\]](#) [\[Edit\]](#) [\[Explain SQL\]](#) [\[Create PHP code\]](#) [\[Refresh\]](#)

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

+ Options

cid	cname	sem	isbn	title
1	ADA	4	1	Algorithm Design
1	ADA	3	1	Algorithm Design
1	ADA	4	8	Analysis of Complex
2	DBMS	4	2	Database System
2	DBMS	3	7	DBMS Design

5. List any department that has all its adopted books published by a specific publisher.

Showing rows 0 - 0 (1 total, Query took 0.0083 seconds.)

```
SELECT DISTINCT c.dept FROM course c WHERE c.dept IN (SELECT c.dept FROM course c,book b,textbook t WHERE c.cid=b.cid AND t.isbn=b.isbn AND t.pub='Pearson') AND c.dept NOT IN (SELECT c.dept FROM course c,book b,textbook t WHERE c.cid=b.cid AND t.isbn=b.isbn AND t.pub<>'Pearson')
```

☐ Profiling [\[Edit inline\]](#) [\[Edit\]](#) [\[Explain SQL\]](#) [\[Create PHP code\]](#) [\[Refresh\]](#)

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

+ Options

dept
CSE

PROGRAM 9: MOVIE DATABASE

Consider the schema for Movie Database:

ACTOR(Act_id, Act_Name, Act_Gender)

DIRECTOR(Dir_id, Dir_Name, Dir_Phone)

MOVIES(Mov_id, Mov_Title, Mov_Year, Mov_Lang, Dir_id)

MOVIE_CAST(Act_id, Mov_id, Role)

RATING(Mov_id, Rev_Stars)

MySQL returned an empty result set (i.e. zero rows). (Query took 0.2805 seconds.)

```
create table moviecast( actid int, movid int, role varchar(30), CONSTRAINT ac_fk FOREIGN key(actid) REFERENCES actor(aid) on update cascade on DELETE CASCADE, CONSTRAINT movie_fk FOREIGN key(movid) REFERENCES movie(mid) on update cascade on DELETE CASCADE )
```

MySQL returned an empty result set (i.e. zero rows). (Query took 0.2074 seconds.)

```
CREATE table review( mid int, stars int(2), CONSTRAINT mov_fk FOREIGN key(mid) REFERENCES movie(mid) on update cascade on DELETE CASCADE )
```

Table	Action	Rows	Type	Collation	Size	Overhead
actor	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
director	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
movie	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
moviecast	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 KiB	-
review	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
5 tables	Sum	0	InnoDB	utf8mb4_general_ci	144.0 KiB	0 B

1. List the titles of all movies directed by 'Hitchcock'.

Showing rows 0 - 2 (3 total, Query took 0.0038 seconds.)

```
SELECT title from movie WHERE did = (SELECT did FROM director WHERE dname='Stephan SpillBerg')
```

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

+ Options

	title
<input type="checkbox"/> Edit Copy Delete	Avataar
<input type="checkbox"/> Edit Copy Delete	Tintin
<input type="checkbox"/> Edit Copy Delete	The Titanic

2. Find the movie names where one or more actors acted in two or more movies.

✓ Showing rows 0 - 4 (5 total, Query took 0.0222 seconds.)

```
SELECT title from movie m, moviecast c WHERE m.mid=c.movid AND actid IN (SELECT actid from moviecast GROUP BY actid having count(actid)>1)
```

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

+ Options

title
The Titanic
Inception
The Shutter Island
Avengers: Endgame
The Pulp Fiction

3. List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).

✓ Showing rows 0 - 3 (4 total, Query took 0.0192 seconds.)

```
SELECT DISTINCT a.aname, m.myear FROM ((actor a JOIN moviecast mc ON a.aid=mc.actid) JOIN movie m ON m.mid=mc.movid) WHERE m.myear NOT BETWEEN 2000 and 2015
```

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

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

+ Options

aname	myear
Leonardo Dicaprio	1997
Sam L Jackson	2019
Sam L Jackson	1996
Scarlet Johhanson	2019

4. Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.

✓ Showing rows 0 - 6 (7 total, Query took 0.0046 seconds.) [title: AVATAAR... - TINTIN...]

```
SELECT title, MAX(stars) FROM movie m, review r WHERE m.mid=r.mid GROUP by m.title having MAX(stars)>0 ORDER BY m.title
```

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

+ Options

title	MAX(stars)
Avataar	5
Avengers: Endgame	5
Green Lantern	2.5
Inception	5
The Shutter Island	5
The Titanic	4.5
Tintin	4.5

5. Update rating of all movies directed by 'Steven Spielberg' to 5.

✓ 3 rows affected. (Query took 0.0736 seconds.)

```
UPDATE review SET stars=5 WHERE mid IN (SELECT mid FROM movie WHERE did=(SELECT did FROM director WHERE dname='Stephan Spillberg'))
```

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

✓ Showing rows 0 - 10 (11 total, Query took 0.0014 seconds.)

```
SELECT * FROM review
```

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

+ Options

mid	stars
1	5
2	5
3	2.5
4	5
5	5
6	4
1	5
5	5
6	5
7	5

PROGRAM 10:COLLEGE DATABASE

Consider the schema for College Database:

STUDENT(USN, SName, Address, Phone, Gender)

SEMSEC(SSID, Sem, Sec)

CLASS(USN, SSID)

SUBJECT(Subcode, Title, Sem, Credits)

IAMARKS(USN, Subcode, SSID, Test1, Test2, Test3, FinalIA)

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.3415 seconds.)

```
CREATE TABLE student( usn varchar(10), sname varchar(20), address varchar(20), phone int(9), gender char, PRIMARY key(usn) )
```

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.2197 seconds.)

```
CREATE TABLE semsec( ssid int, sem int, sec char, PRIMARY key(ssid) )
```

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.2309 seconds.)

```
CREATE TABLE subject( subcode varchar(10), title varchar(20), sem int, credits int, PRIMARY key(subcode) )
```

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.3757 seconds.)

```
CREATE TABLE class( usn varchar(10), ssid int, constraint usn_fk FOREIGN key(usn) REFERENCES student(usn) ON DELETE CASCADE ON UPDATE CASCADE, constraint ssid_fk FOREIGN key(ssid) REFERENCES semsec(ssid) ON DELETE CASCADE ON UPDATE CASCADE )
```

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

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.2417 seconds.)

```
CREATE table iamarks( usn varchar(10), subcode varchar(10), ssid int, test1 int, test2 int, test3 int, final int, constraint usn_fk_2 FOREIGN key(usn) REFERENCES student(usn) ON UPDATE CASCADE ON DELETE cascade, constraint sub_fk FOREIGN key(subcode) REFERENCES subject(subcode) ON UPDATE CASCADE ON DELETE cascade, constraint ssid_fk_2 FOREIGN key(ssid) REFERENCES semsec(ssid) ON DELETE CASCADE ON UPDATE CASCADE )
```

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

1. List all the student details studying in fourth semester 'C' section.

✓ Showing rows 0 - 0 (1 total, Query took 0.0028 seconds.)

```
SELECT * from student WHERE usn in (SELECT usn from class WHERE ssid in (SELECT ssid from semsec WHERE sec='A' AND sem=4))
```

☐ Show all | Number of rows: 25 | Filter rows:

+ Options

	usn	sname	address	phone	gender
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1xy19cs001	aaa	qwerty yuiop	123456789	M

2. Compute the total number of male and female students in each semester and in each section.

✓ Showing rows 0 - 4 (5 total, Query took 0.0043 seconds.)

```
SELECT ss.sem,s.gender,count(*) from student s,semsec ss,class c WHERE s.usn=c.usn AND ss.ssid=c.ssid group by s.gender,ss.sem,ss.sec
```

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

+ Options

sem	gender	count(*)
1	F	1
3	F	1
1	M	1
4	M	1
5	M	1

3. Create a view of Test1 marks of student USN '1BI15CS101' in all subjects.

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.2097 seconds.)

```
CREATE VIEW test1 as SELECT s.sname,ia.test1 FROM student s,iamarks ia WHERE ia.usn=s.usn AND s.usn='1xy19cs002'
```

Server: 127.0.0.1 Database: college View: test1

Browse Structure SQL Search Insert Export Privileges

⚠ Current selection does not contain a unique column. Grid edit, Edit, Copy and Delete features may result in data loss.

✓ Showing rows 0 - 1 (2 total, Query took 0.0010 seconds.)

```
SELECT * FROM `test1`
```

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

+ Options

	sname	test1
<input type="checkbox"/> Edit Copy Delete	abc	36
<input type="checkbox"/> Edit Copy Delete	abc	39