Assignment Programme : Bachelor of Information Technology (BIT)

Subject : Database Management System

Medium : English

Deadline : 15th November, 2023

Chapter 01

Question no 01

1. A DBMS, or Database Management System, is a software system that allows users to efficiently store, retrieve, manage, and manipulate data in a structured way. It serves as an intermediary between the user, applications, and the actual data stored in the database.

2. Advantages of a DBMS include:

- Data Integrity: Data accuracy and consistency are guaranteed by DBMS.
- Data Security: Permissions and access control safeguard data.
- Data Independence: Application changes have no effect on data storage changes.
- Concurrent Access: Data can be accessed concurrently by several people.
- Data Recovery: Options for backup and restoration provide protection against data loss.
- Query Language: A language for reporting and querying data is provided by DBMS.

3. Comparison between traditional File-based system and Database approach:

- A file-based system keeps data in discrete files without any central structure, whereas a database system organizes data into tables with relationships.
- Database systems provide effective querying, data integrity, and security, while file-based systems can lack these qualities.
- File-based solutions are typically less scalable and more prone to errors when compared to databases.

4. Major components of a Database Manager include:

- Data Definition Language (DDL): Allows defining the database structure.
- Data Manipulation Language (DML): Enables adding, modifying, and querying data.
- Query Processor: Handles user queries and retrieves data.
- Storage Manager: Manages data storage and retrieval.
- Transaction Manager: Ensures data consistency in transactions.
- Security and Authorization: Controls access to the database.

5. A database administrator often oversees both the applications that access the data and the data itself (DBA). They are responsible for maintaining backups, setting data access rules, assuring data security, and maximizing database performance.

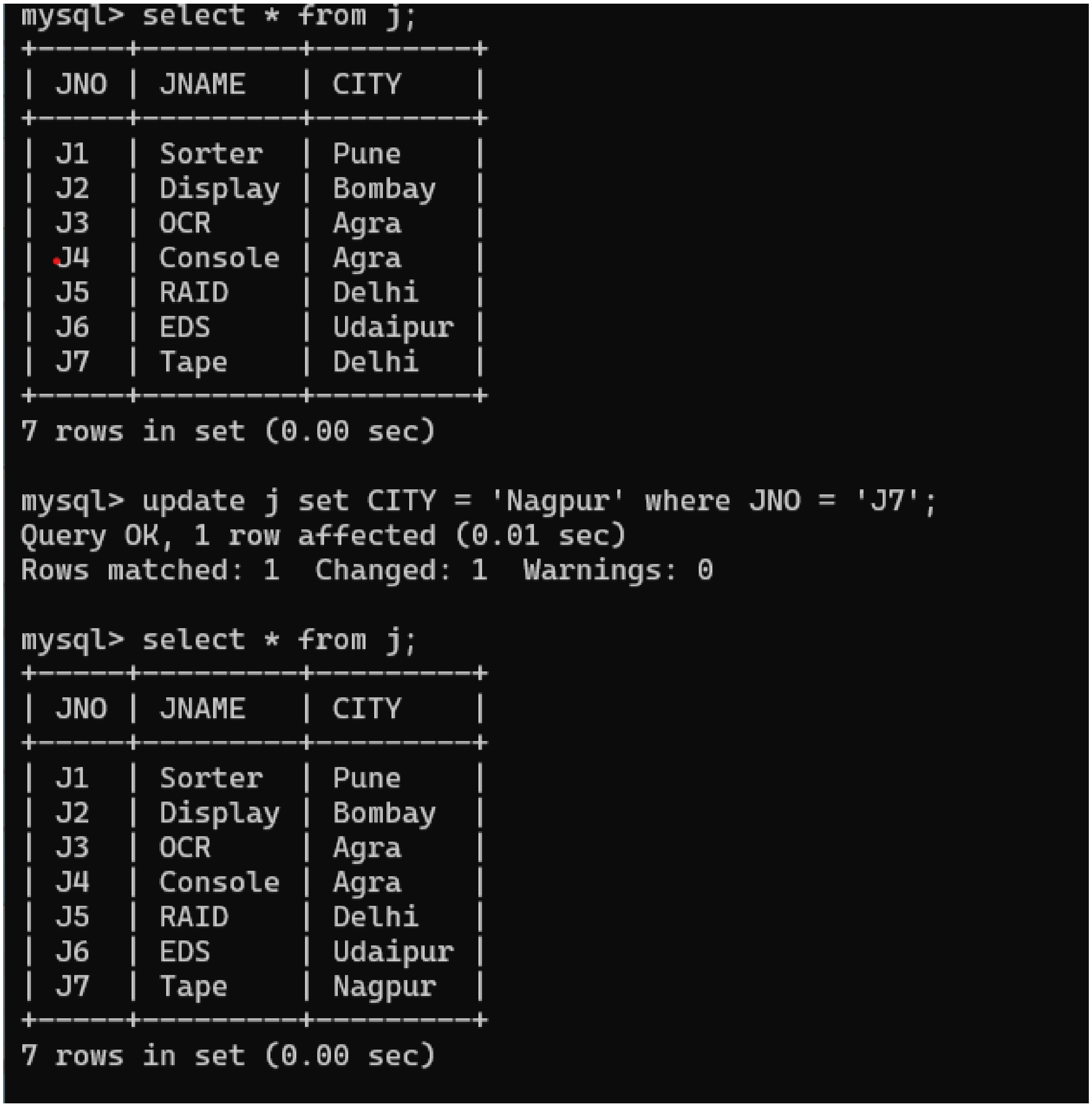
6. State whether the following statements are True or False:

- True: Where and how data are arranged in physical data storage is specified by the external schema.
- True: The intellectual and physical components of data representation are kept apart by a schema.
- True: For certain users, the conceptual schema specifies one or more database views.
- True: A database is an assortment of data intended for use by various users.
- True: The way that data are kept in a database ensures that they are unaffected by the applications or users who use them.
- True: Redundancy in database use can be minimized.
- False: A database's contents can be shared.
- False: It is not feasible to implement security limitations within a database.
- True: Data integrity may be preserved in a database.
- True: In order to prevent modifications to the schema at one level from affecting the other levels, the three levels of the schema internal, conceptual, and external must be independent of one another.

Chapter 02

Question 01

a) update j set CITY = 'Nagpur' where JNO = 'J7';

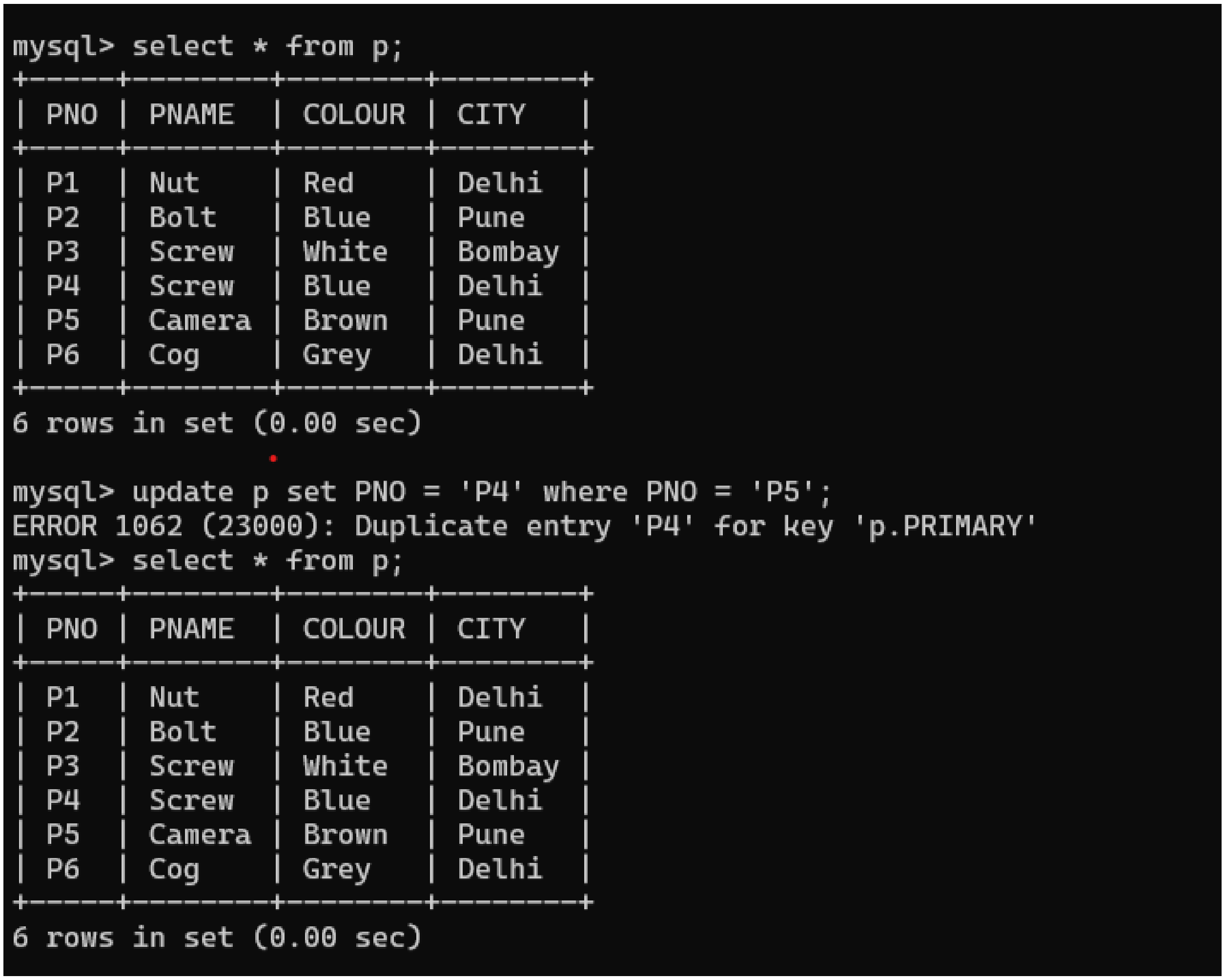


```
mysql> select * from j;
+-----+---------+---------+
| JNO | JNAME   | CITY    |
+-----+---------+---------+
| J1  | Sorter  | Pune    |
| J2  | Display | Bombay  |
| J3  | OCR     | Agra    |
| J4  | Console | Agra    |
| J5  | RAID    | Delhi   |
| J6  | EDS     | Udaipur |
| J7  | Tape    | Delhi   |
+-----+---------+---------+
7 rows in set (0.00 sec)

mysql> update j set CITY = 'Nagpur' where JNO = 'J7';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from j;
+-----+---------+---------+
| JNO | JNAME   | CITY    |
+-----+---------+---------+
| J1  | Sorter  | Pune    |
| J2  | Display | Bombay  |
| J3  | OCR     | Agra    |
| J4  | Console | Agra    |
| J5  | RAID    | Delhi   |
| J6  | EDS     | Udaipur |
| J7  | Tape    | Nagpur  |
+-----+---------+---------+
7 rows in set (0.00 sec)
```

b.) mysql> update p set PNO = 'P4' where PNO = 'P5';

ERROR 1062 (23000): Duplicate entry 'P4' for key 'p.PRIMARY'

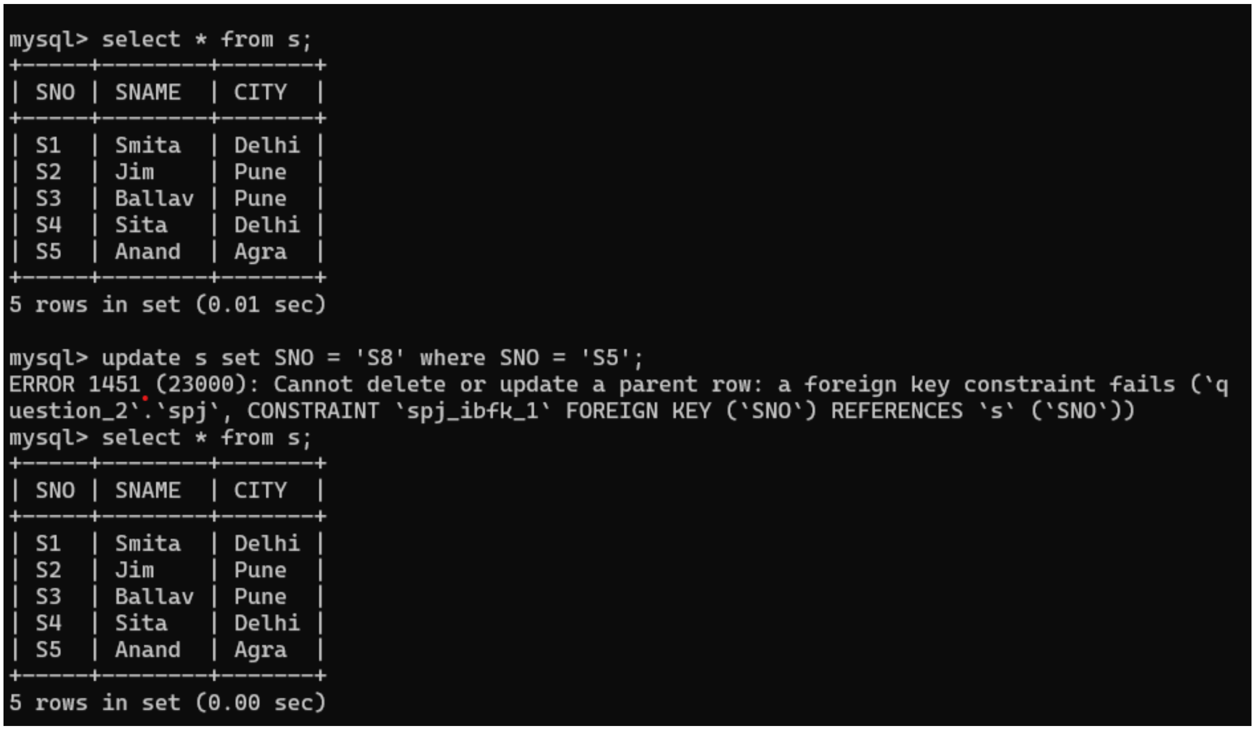
** we can’t add duplicate value in primary key column.



```
mysql> select * from p;
+-----+--------+--------+--------+
| PNO | PNAME  | COLOUR | CITY   |
+-----+--------+--------+--------+
| P1  | Nut    | Red    | Delhi  |
| P2  | Bolt   | Blue   | Pune   |
| P3  | Screw  | White  | Bombay |
| P4  | Screw  | Blue   | Delhi  |
| P5  | Camera | Brown  | Pune   |
| P6  | Cog    | Grey   | Delhi  |
+-----+--------+--------+--------+
6 rows in set (0.00 sec)

mysql> update p set PNO = 'P4' where PNO = 'P5';
ERROR 1062 (23000): Duplicate entry 'P4' for key 'p.PRIMARY'
mysql> select * from p;
+-----+--------+--------+--------+
| PNO | PNAME  | COLOUR | CITY   |
+-----+--------+--------+--------+
| P1  | Nut    | Red    | Delhi  |
| P2  | Bolt   | Blue   | Pune   |
| P3  | Screw  | White  | Bombay |
| P4  | Screw  | Blue   | Delhi  |
| P5  | Camera | Brown  | Pune   |
| P6  | Cog    | Grey   | Delhi  |
+-----+--------+--------+--------+
6 rows in set (0.00 sec)
```

c.) mysql> update s set SNO = 'S8' where SNO = 'S5';

ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails (`question_2`.`spj`, CONSTRAINT `spj_ibfk_1` FOREIGN KEY (`SNO`) REFERENCES `s` (`SNO`))



```
mysql> select * from s;
+-----+--------+-------+
| SNO | SNAME  | CITY  |
+-----+--------+-------+
| S1  | Smita  | Delhi |
| S2  | Jim    | Pune  |
| S3  | Ballav | Pune  |
| S4  | Sita   | Delhi |
| S5  | Anand  | Agra  |
+-----+--------+-------+
5 rows in set (0.01 sec)

mysql> update s set SNO = 'S8' where SNO = 'S5';
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails (`q
uestion_2`.`spj`, CONSTRAINT `spj_ibfk_1` FOREIGN KEY (`SNO`) REFERENCES `s` (`SNO`))
mysql> select * from s;
+-----+--------+-------+
| SNO | SNAME  | CITY  |
+-----+--------+-------+
| S1  | Smita  | Delhi |
| S2  | Jim    | Pune  |
| S3  | Ballav | Pune  |
| S4  | Sita   | Delhi |
| S5  | Anand  | Agra  |
+-----+--------+-------+
5 rows in set (0.00 sec)
```

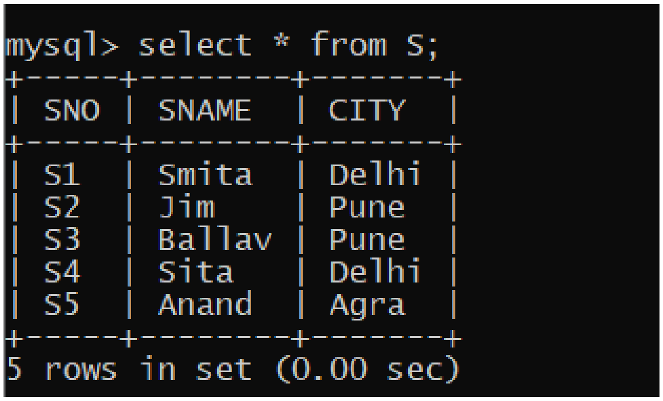
d.)

we edit SPJ table as

mysql> create table SPJ (SNO varchar(3),PNO varchar(3),JNO varchar(3),QUANTITY text(25),foreign key(JNO) references J(JNO) ON DELETE CASCADE,foreign key (PNO) references P(PNO) ON DELETE CASCADE,foreign key (SNO) references S(SNO) ON DELETE CASCADE);

Query OK, 0 rows affected (0.05 sec)

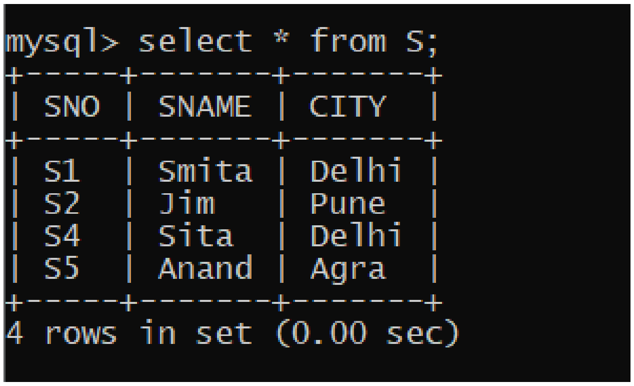
then



```
mysql> select * from S;
+-----+--------+-------+
| SNO | SNAME  | CITY  |
+-----+--------+-------+
| S1  | Smita  | Delhi |
| S2  | Jim    | Pune  |
| S3  | Ballav | Pune  |
| S4  | Sita   | Delhi |
| S5  | Anand  | Agra  |
+-----+--------+-------+
5 rows in set (0.00 sec)
```

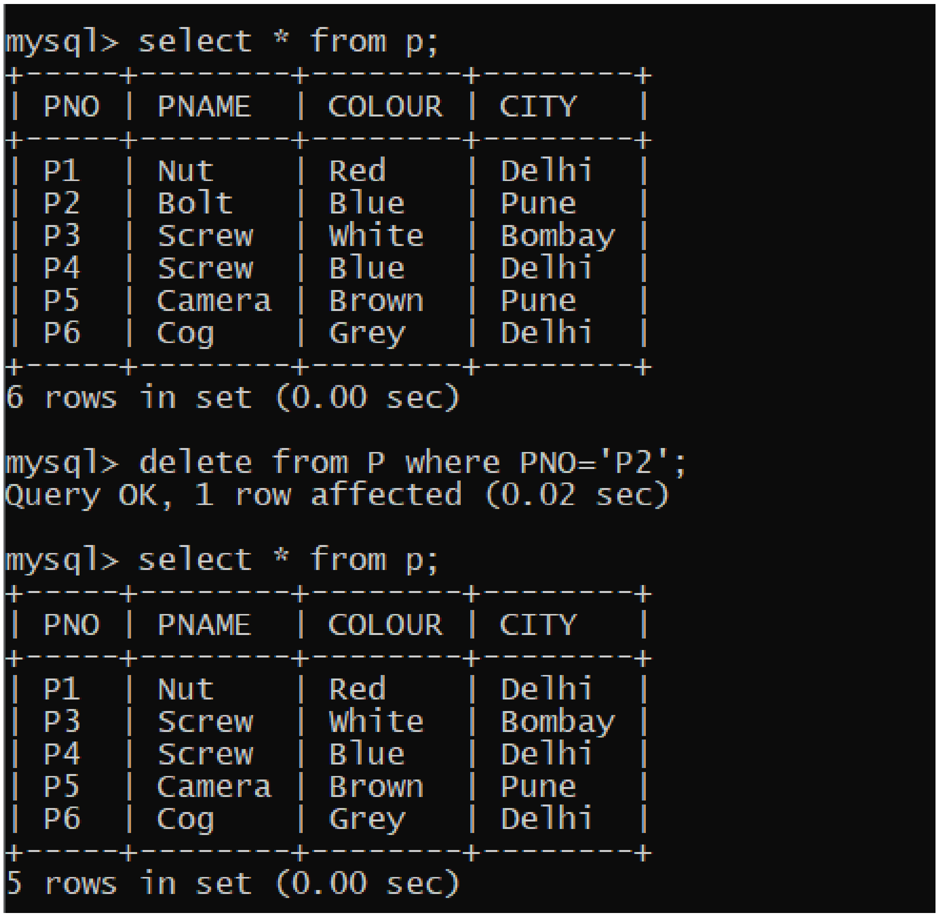
mysql> delete from S where SNO='S3';

Query OK, 1 row affected (0.02 sec)



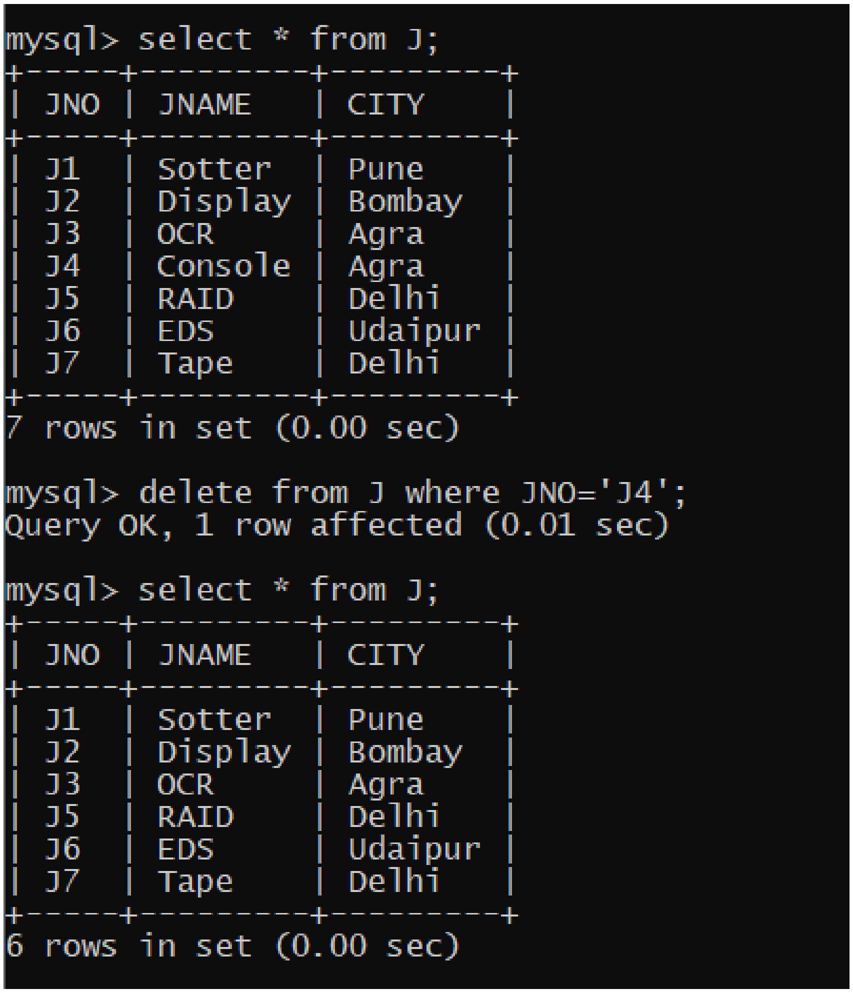
```
mysql> select * from S;
+-----+-------+-------+
| SNO | SNAME | CITY  |
+-----+-------+-------+
| S1  | Smita | Delhi |
| S2  | Jim   | Pune  |
| S4  | Sita  | Delhi |
| S5  | Anand | Agra  |
+-----+-------+-------+
4 rows in set (0.00 sec)
```

e.)



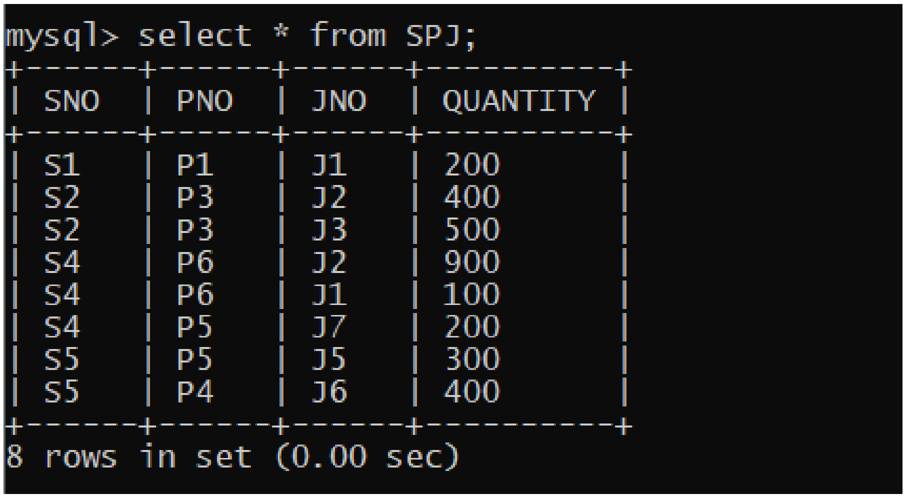
```
mysql> select * from p;
+-----+--------+--------+--------+
| PNO | PNAME  | COLOUR | CITY   |
+-----+--------+--------+--------+
| P1  | Nut    | Red    | Delhi  |
| P2  | Bolt   | Blue   | Pune   |
| P3  | Screw  | White  | Bombay |
| P4  | Screw  | Blue   | Delhi  |
| P5  | Camera | Brown  | Pune   |
| P6  | Cog    | Grey   | Delhi  |
+-----+--------+--------+--------+
6 rows in set (0.00 sec)

mysql> delete from P where PNO='P2';
Query OK, 1 row affected (0.02 sec)

mysql> select * from p;
+-----+--------+--------+--------+
| PNO | PNAME  | COLOUR | CITY   |
+-----+--------+--------+--------+
| P1  | Nut    | Red    | Delhi  |
| P3  | Screw  | White  | Bombay |
| P4  | Screw  | Blue   | Delhi  |
| P5  | Camera | Brown  | Pune   |
| P6  | Cog    | Grey   | Delhi  |
+-----+--------+--------+--------+
5 rows in set (0.00 sec)
```

f.)

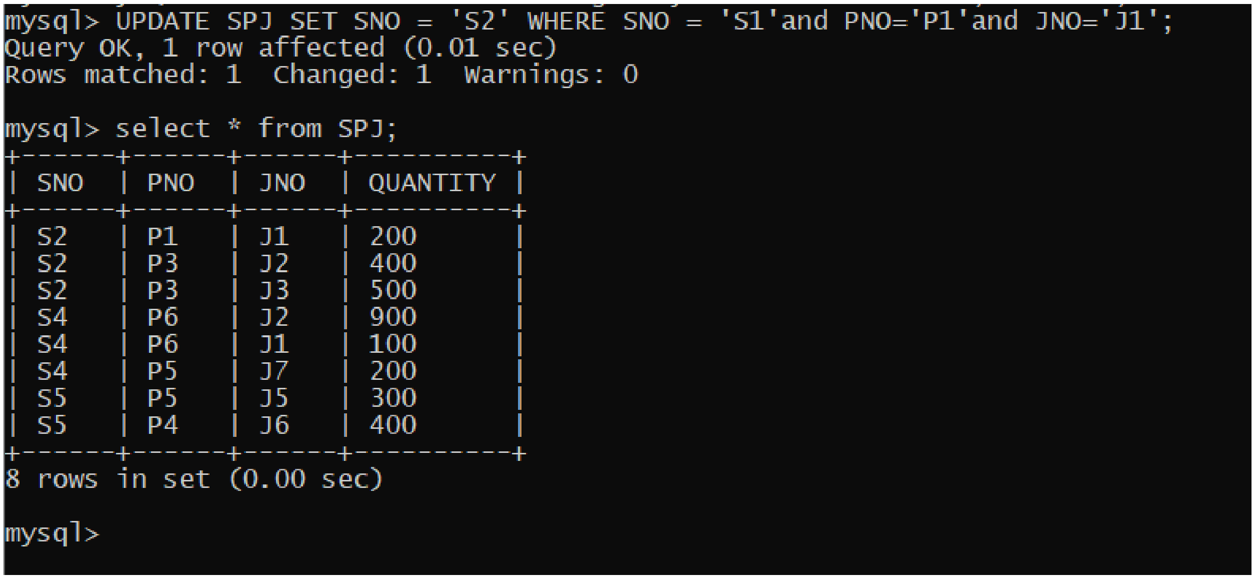


```
mysql> select * from J;
+-----+---------+---------+
| JNO | JNAME   | CITY    |
+-----+---------+---------+
| J1  | Sotter  | Pune    |
| J2  | Display | Bombay  |
| J3  | OCR     | Agra    |
| J4  | Console | Agra    |
| J5  | RAID    | Delhi   |
| J6  | EDS     | Udaipur |
| J7  | Tape    | Delhi   |
+-----+---------+---------+
7 rows in set (0.00 sec)

mysql> delete from J where JNO='J4';
Query OK, 1 row affected (0.01 sec)

mysql> select * from J;
+-----+---------+---------+
| JNO | JNAME   | CITY    |
+-----+---------+---------+
| J1  | Sotter  | Pune    |
| J2  | Display | Bombay  |
| J3  | OCR     | Agra    |
| J5  | RAID    | Delhi   |
| J6  | EDS     | Udaipur |
| J7  | Tape    | Delhi   |
+-----+---------+---------+
6 rows in set (0.00 sec)
```

g.)

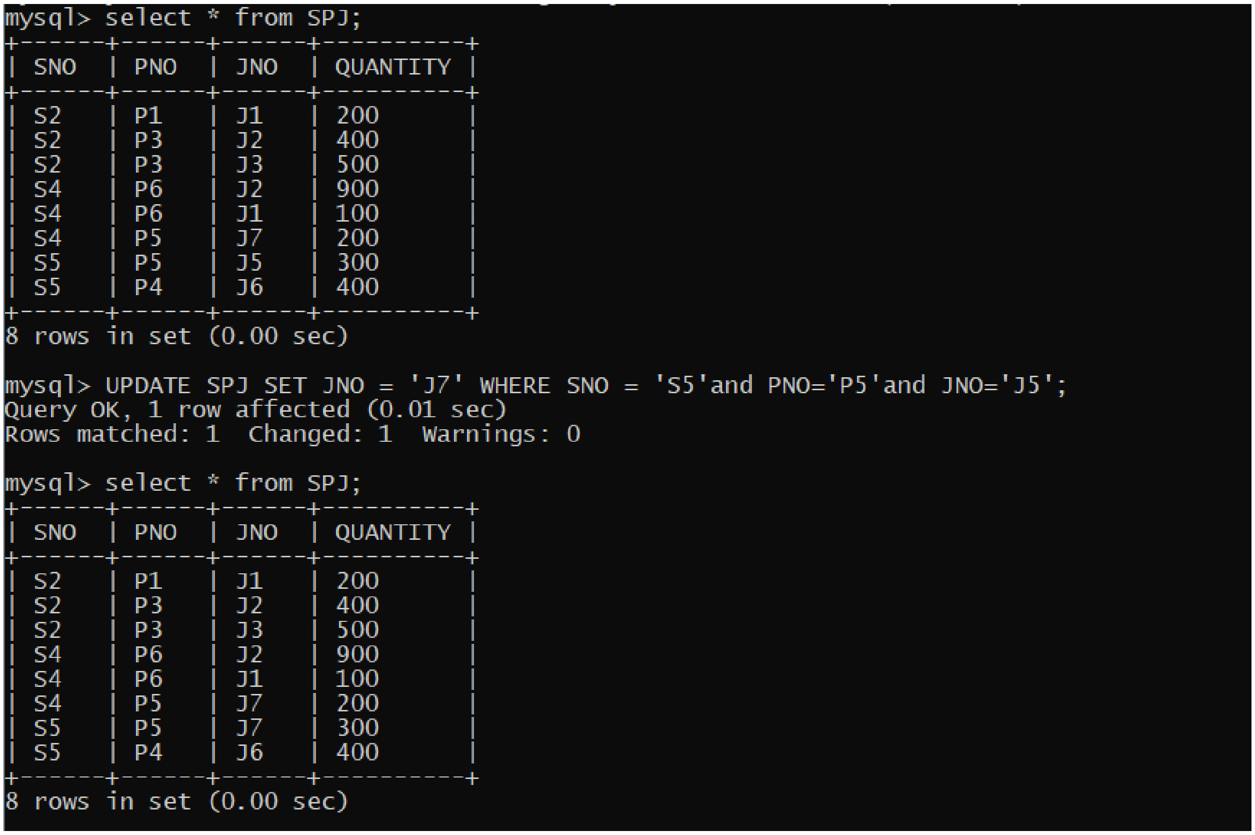


```
mysql> select * from SPJ;
+------+------+------+----------+
| SNO  | PNO  | JNO  | QUANTITY |
+------+------+------+----------+
| S1   | P1   | J1   | 200      |
| S2   | P3   | J2   | 400      |
| S2   | P3   | J3   | 500      |
| S4   | P6   | J2   | 900      |
| S4   | P6   | J1   | 100      |
| S4   | P5   | J7   | 200      |
| S5   | P5   | J5   | 300      |
| S5   | P4   | J6   | 400      |
+------+------+------+----------+
8 rows in set (0.00 sec)
```



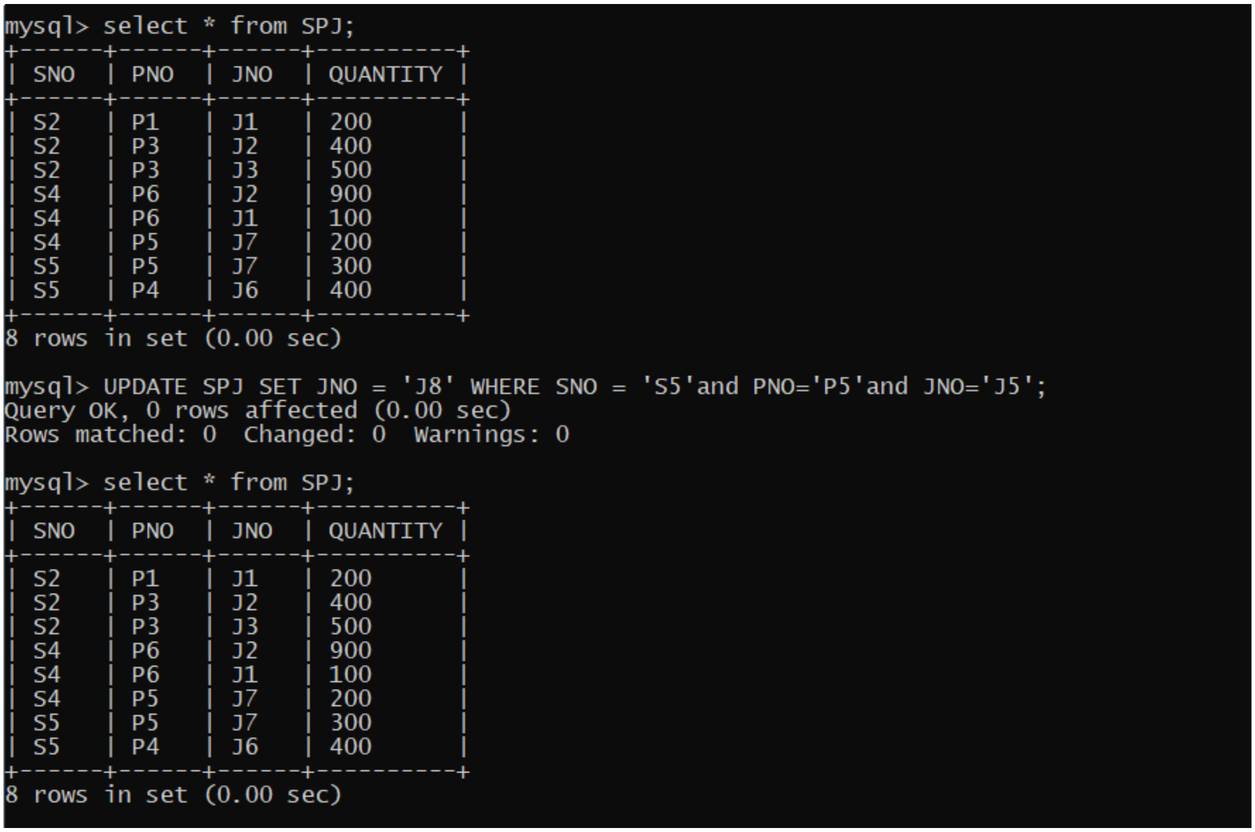
```
mysql> UPDATE SPJ SET SNO = 'S2' WHERE SNO = 'S1'and PNO='P1'and JNO='J1';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from SPJ;
+------+------+------+----------+
| SNO  | PNO  | JNO  | QUANTITY |
+------+------+------+----------+
| S2   | P1   | J1   | 200      |
| S2   | P3   | J2   | 400      |
| S2   | P3   | J3   | 500      |
| S4   | P6   | J2   | 900      |
| S4   | P6   | J1   | 100      |
| S4   | P5   | J7   | 200      |
| S5   | P5   | J5   | 300      |
| S5   | P4   | J6   | 400      |
+------+------+------+----------+
8 rows in set (0.00 sec)

mysql>
```

h.)



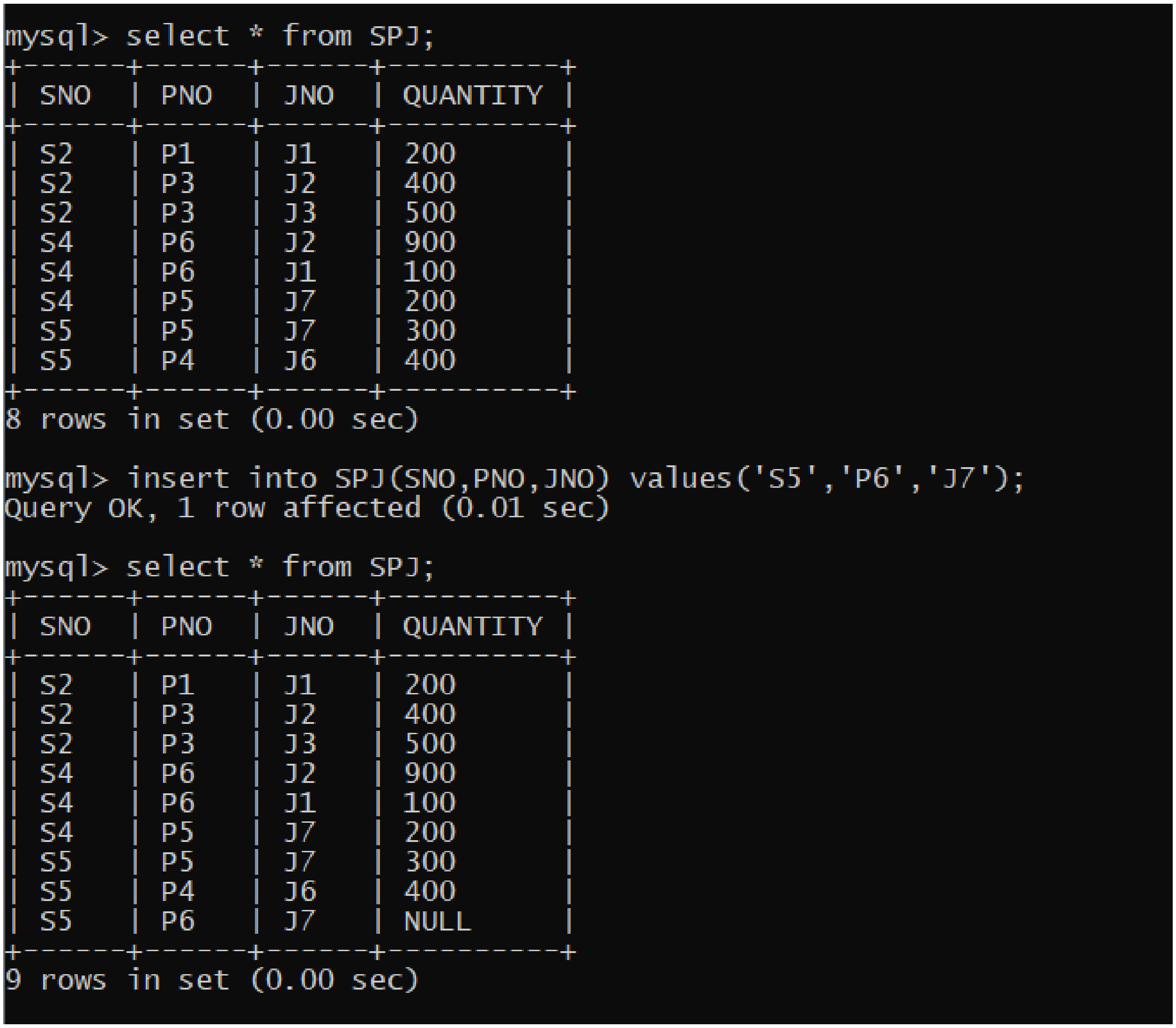
```
mysql> select * from SPJ;
+------+------+------+----------+
| SNO  | PNO  | JNO  | QUANTITY |
+------+------+------+----------+
| S2   | P1   | J1   | 200      |
| S2   | P3   | J2   | 400      |
| S2   | P3   | J3   | 500      |
| S4   | P6   | J2   | 900      |
| S4   | P6   | J1   | 100      |
| S4   | P5   | J7   | 200      |
| S5   | P5   | J5   | 300      |
| S5   | P4   | J6   | 400      |
+------+------+------+----------+
8 rows in set (0.00 sec)

mysql> UPDATE SPJ SET JNO = 'J7' WHERE SNO = 'S5'and PNO='P5'and JNO='J5';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from SPJ;
+------+------+------+----------+
| SNO  | PNO  | JNO  | QUANTITY |
+------+------+------+----------+
| S2   | P1   | J1   | 200      |
| S2   | P3   | J2   | 400      |
| S2   | P3   | J3   | 500      |
| S4   | P6   | J2   | 900      |
| S4   | P6   | J1   | 100      |
| S4   | P5   | J7   | 200      |
| S5   | P5   | J7   | 300      |
| S5   | P4   | J6   | 400      |
+------+------+------+----------+
8 rows in set (0.00 sec)
```

i.)



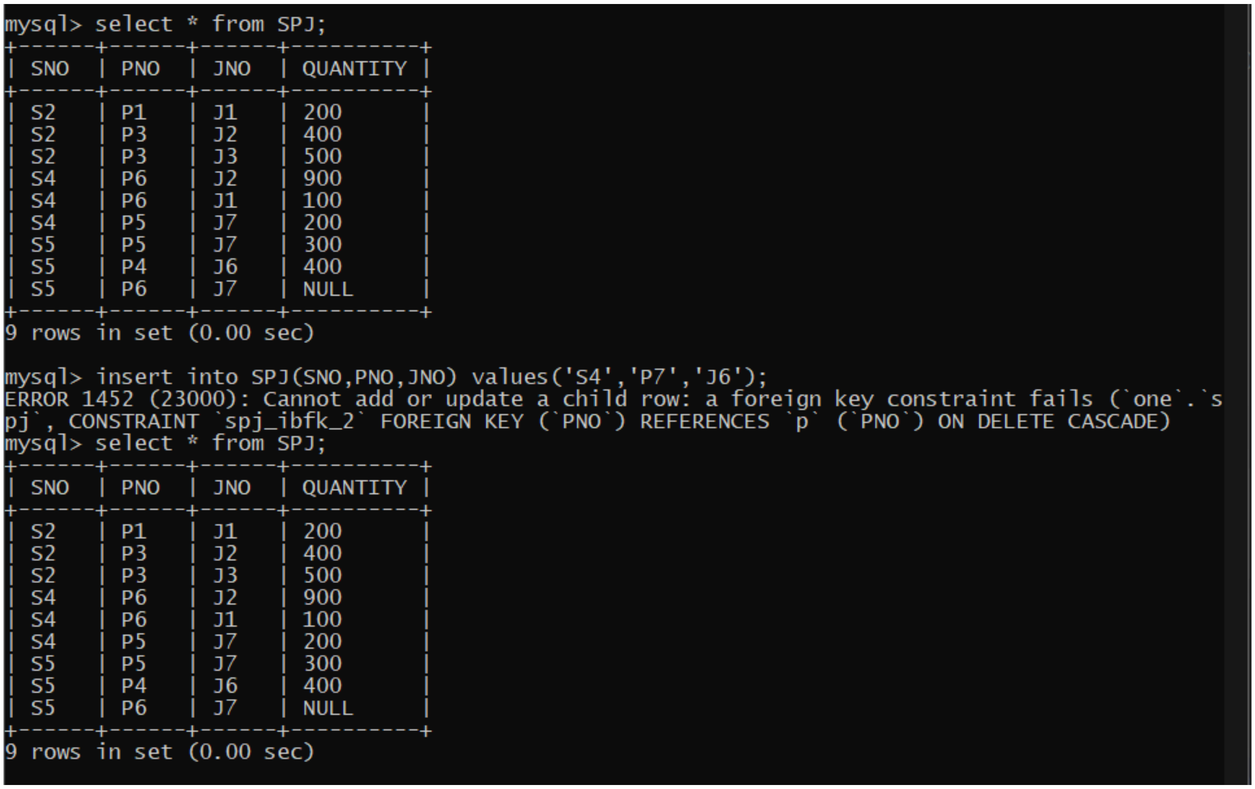
```
mysql> select * from SPJ;
+------+------+------+----------+
| SNO  | PNO  | JNO  | QUANTITY |
+------+------+------+----------+
| S2   | P1   | J1   | 200      |
| S2   | P3   | J2   | 400      |
| S2   | P3   | J3   | 500      |
| S4   | P6   | J2   | 900      |
| S4   | P6   | J1   | 100      |
| S4   | P5   | J7   | 200      |
| S5   | P5   | J7   | 300      |
| S5   | P4   | J6   | 400      |
+------+------+------+----------+
8 rows in set (0.00 sec)

mysql> UPDATE SPJ SET JNO = 'J8' WHERE SNO = 'S5'and PNO='P5'and JNO='J5';
Query OK, 0 rows affected (0.00 sec)
Rows matched: 0  Changed: 0  Warnings: 0

mysql> select * from SPJ;
+------+------+------+----------+
| SNO  | PNO  | JNO  | QUANTITY |
+------+------+------+----------+
| S2   | P1   | J1   | 200      |
| S2   | P3   | J2   | 400      |
| S2   | P3   | J3   | 500      |
| S4   | P6   | J2   | 900      |
| S4   | P6   | J1   | 100      |
| S4   | P5   | J7   | 200      |
| S5   | P5   | J7   | 300      |
| S5   | P4   | J6   | 400      |
+------+------+------+----------+
8 rows in set (0.00 sec)
```

J.)



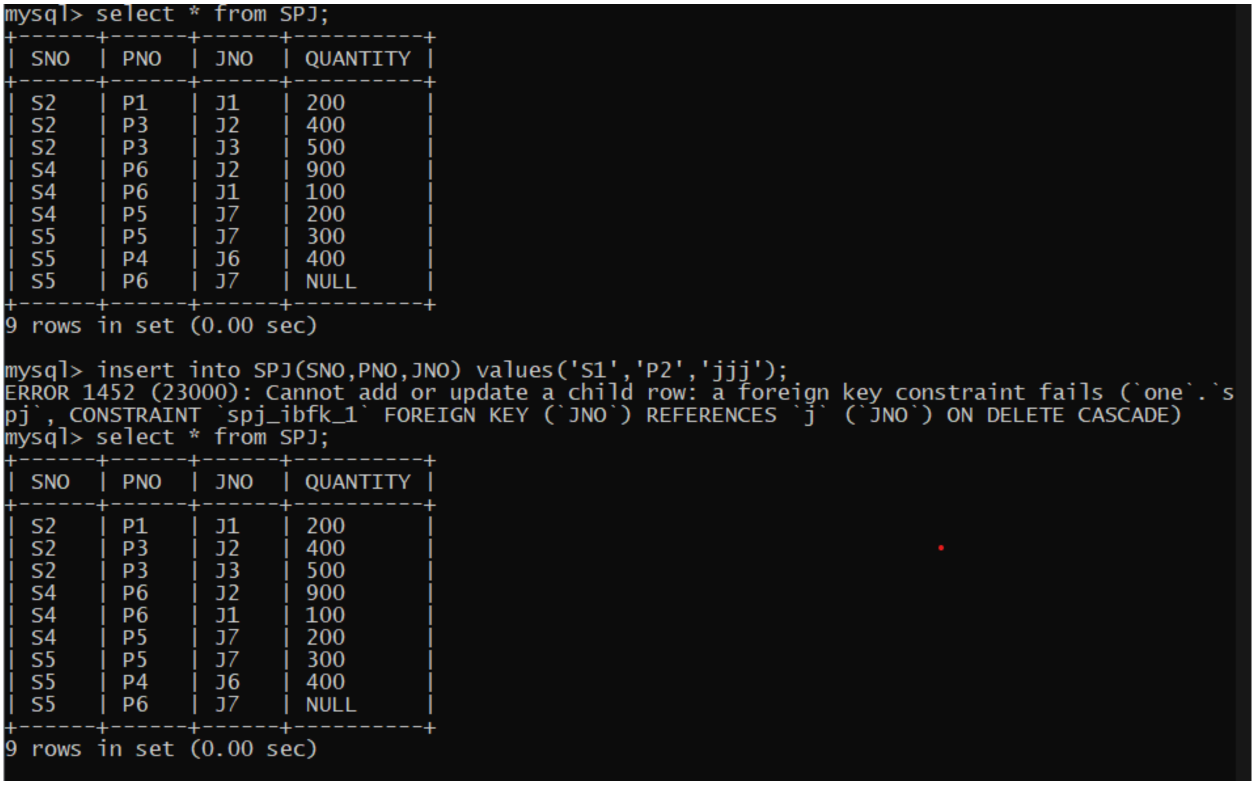
```
mysql> select * from SPJ;
+------+------+------+----------+
| SNO  | PNO  | JNO  | QUANTITY |
+------+------+------+----------+
| S2   | P1   | J1   | 200      |
| S2   | P3   | J2   | 400      |
| S2   | P3   | J3   | 500      |
| S4   | P6   | J2   | 900      |
| S4   | P6   | J1   | 100      |
| S4   | P5   | J7   | 200      |
| S5   | P5   | J7   | 300      |
| S5   | P4   | J6   | 400      |
+------+------+------+----------+
8 rows in set (0.00 sec)

mysql> insert into SPJ(SNO,PNO,JNO) values('S5','P6','J7');
Query OK, 1 row affected (0.01 sec)

mysql> select * from SPJ;
+------+------+------+----------+
| SNO  | PNO  | JNO  | QUANTITY |
+------+------+------+----------+
| S2   | P1   | J1   | 200      |
| S2   | P3   | J2   | 400      |
| S2   | P3   | J3   | 500      |
| S4   | P6   | J2   | 900      |
| S4   | P6   | J1   | 100      |
| S4   | P5   | J7   | 200      |
| S5   | P5   | J7   | 300      |
| S5   | P4   | J6   | 400      |
| S5   | P6   | J7   | NULL     |
+------+------+------+----------+
9 rows in set (0.00 sec)
```

k.)



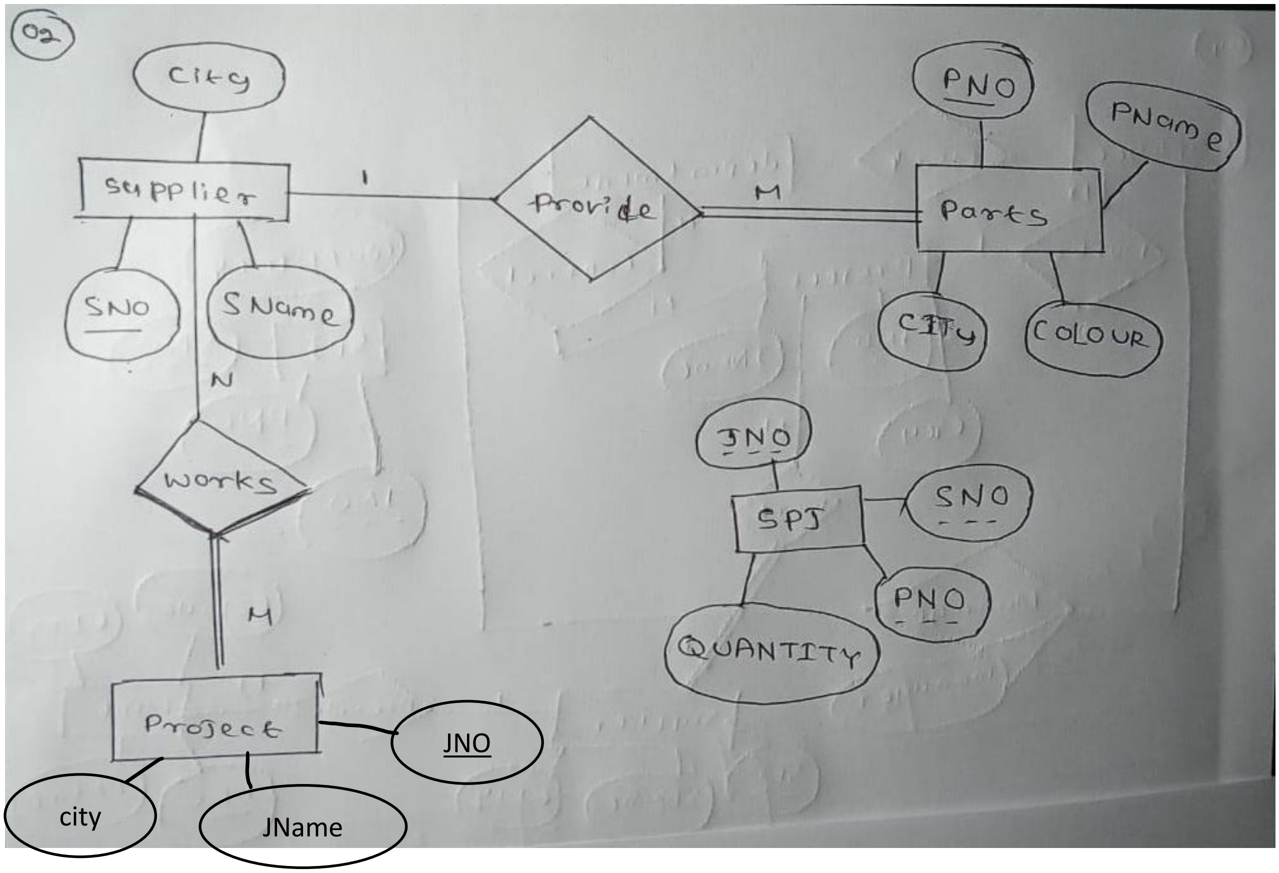
```
mysql> select * from SPJ;
+------+------+------+----------+
| SNO  | PNO  | JNO  | QUANTITY |
+------+------+------+----------+
| S2   | P1   | J1   | 200      |
| S2   | P3   | J2   | 400      |
| S2   | P3   | J3   | 500      |
| S4   | P6   | J2   | 900      |
| S4   | P6   | J1   | 100      |
| S4   | P5   | J7   | 200      |
| S5   | P5   | J7   | 300      |
| S5   | P4   | J6   | 400      |
| S5   | P6   | J7   | NULL     |
+------+------+------+----------+
9 rows in set (0.00 sec)

mysql> insert into SPJ(SNO,PNO,JNO) values('S4','P7','J6');
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`one`.`s
pj`, CONSTRAINT `spj_ibfk_2` FOREIGN KEY (`PNO`) REFERENCES `p` (`PNO`) ON DELETE CASCADE)
mysql> select * from SPJ;
+------+------+------+----------+
| SNO  | PNO  | JNO  | QUANTITY |
+------+------+------+----------+
| S2   | P1   | J1   | 200      |
| S2   | P3   | J2   | 400      |
| S2   | P3   | J3   | 500      |
| S4   | P6   | J2   | 900      |
| S4   | P6   | J1   | 100      |
| S4   | P5   | J7   | 200      |
| S5   | P5   | J7   | 300      |
| S5   | P4   | J6   | 400      |
| S5   | P6   | J7   | NULL     |
+------+------+------+----------+
9 rows in set (0.00 sec)
```

1.)



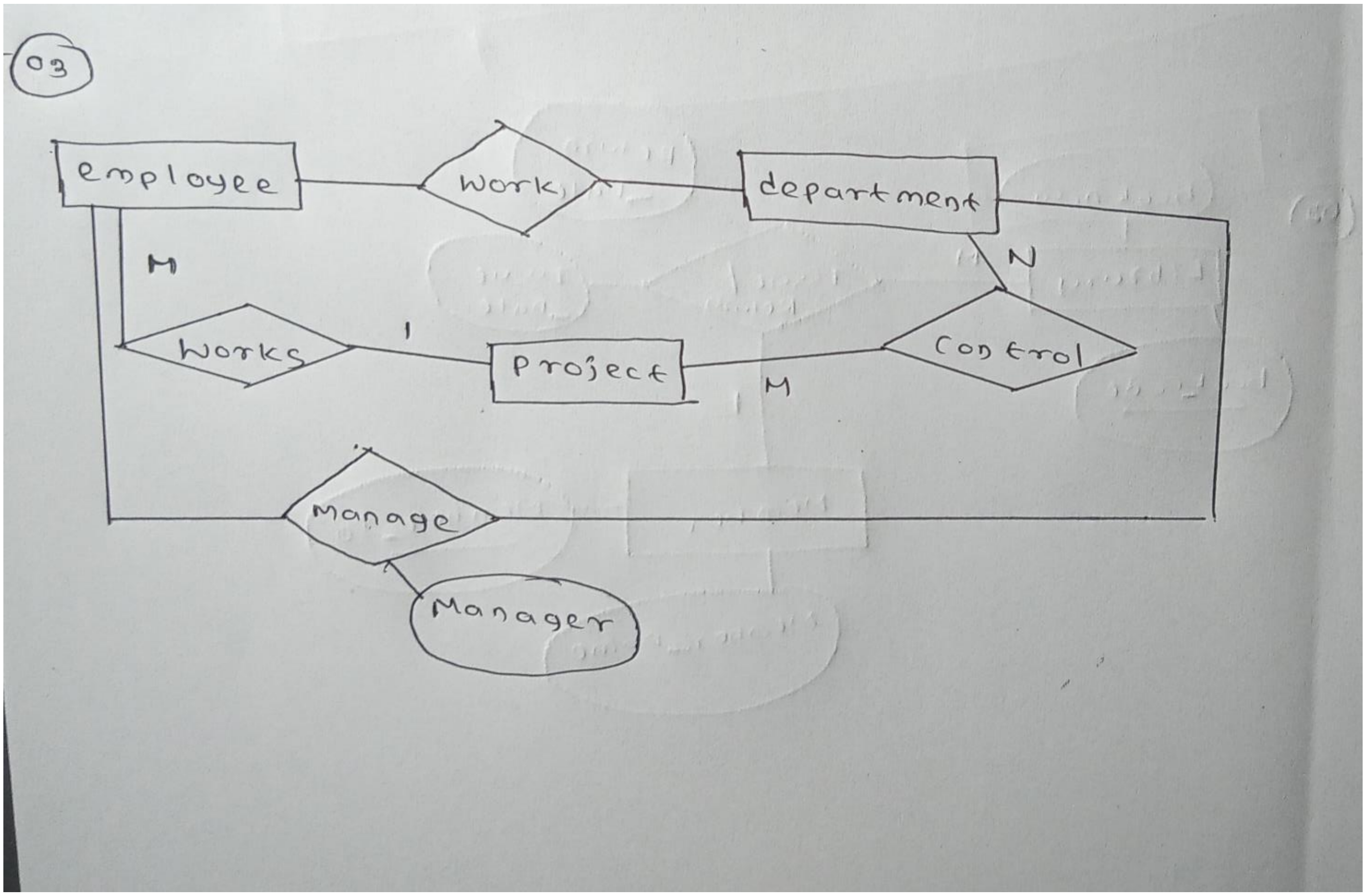
```
mysql> select * from SPJ;
+------+------+------+----------+
| SNO  | PNO  | JNO  | QUANTITY |
+------+------+------+----------+
| S2   | P1   | J1   | 200      |
| S2   | P3   | J2   | 400      |
| S2   | P3   | J3   | 500      |
| S4   | P6   | J2   | 900      |
| S4   | P6   | J1   | 100      |
| S4   | P5   | J7   | 200      |
| S5   | P5   | J7   | 300      |
| S5   | P4   | J6   | 400      |
| S5   | P6   | J7   | NULL     |
+------+------+------+----------+
9 rows in set (0.00 sec)

mysql> insert into SPJ(SNO,PNO,JNO) values('S1','P2','jjj');
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`one`.`s
pj`, CONSTRAINT `spj_ibfk_1` FOREIGN KEY (`JNO`) REFERENCES `j` (`JNO`) ON DELETE CASCADE)
mysql> select * from SPJ;
+------+------+------+----------+
| SNO  | PNO  | JNO  | QUANTITY |
+------+------+------+----------+
| S2   | P1   | J1   | 200      |
| S2   | P3   | J2   | 400      |
| S2   | P3   | J3   | 500      |
| S4   | P6   | J2   | 900      |
| S4   | P6   | J1   | 100      |
| S4   | P5   | J7   | 200      |
| S5   | P5   | J7   | 300      |
| S5   | P4   | J6   | 400      |
| S5   | P6   | J7   | NULL     |
+------+------+------+----------+
9 rows in set (0.00 sec)
```

Question 02

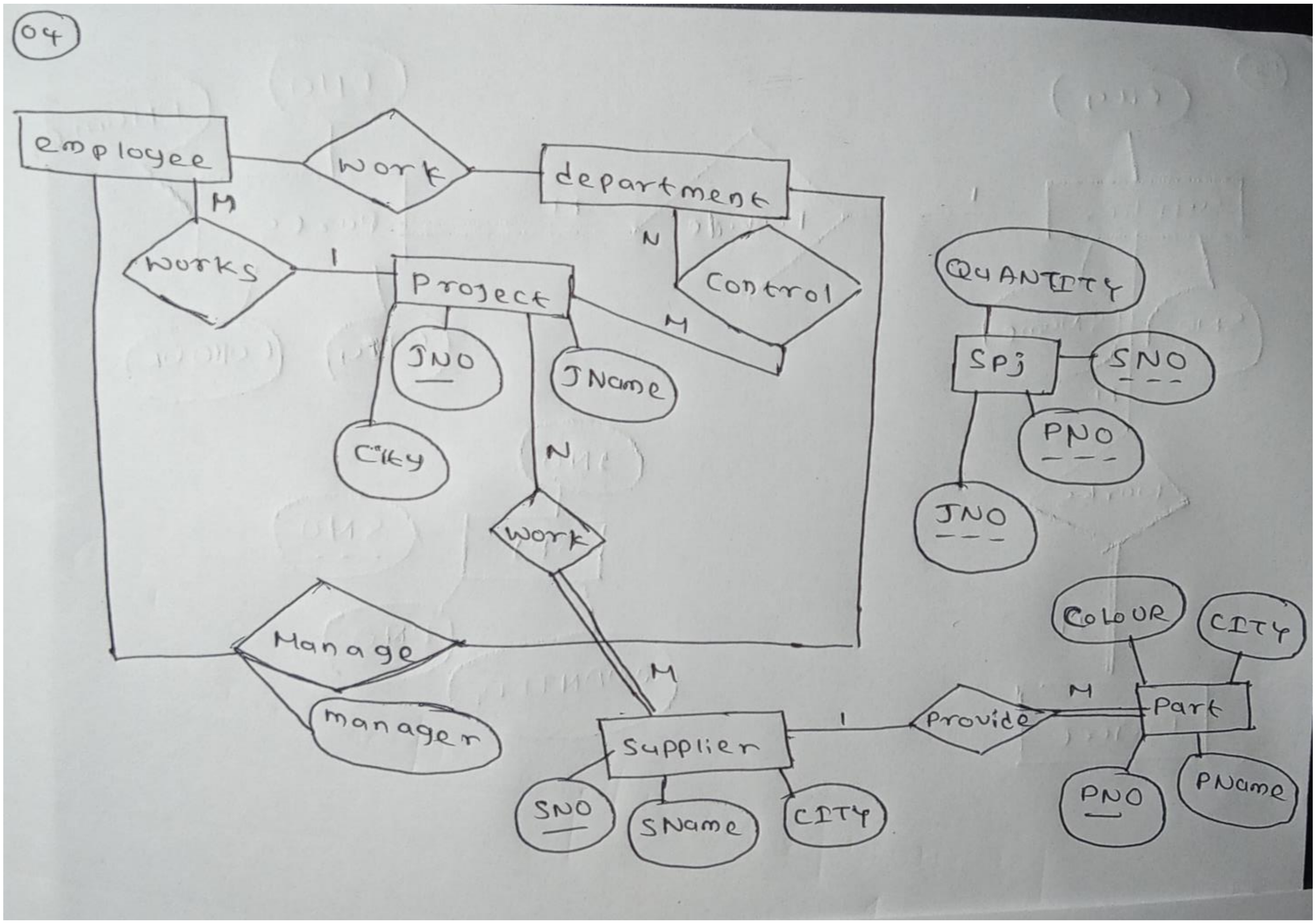


We assume that one of supplier can provide many numbers of parts and supplier must be works on several project as well as many numbers of project can be covered by several number of suppliers.

Question 03


03
employee
work
department
M
N
works
1
project
control
M
Manage
Manager

Question 04


04
employee
work
department
M
N
works
1
Project
Control
M
JNO
JName
City
N
work
Manage
manager
M
Supplier
1
Provide
M
Part
SNO
SName
CITY
QUANTITY
Spj
SNO
PNO
JNO
COLOUR
CITY
PNO
PName

Chapter 03

Question 01

a.)

candidate keys

- SNO , PNO , PROJNO from SUP_PAR_PROJ

Primary Keys

- SNO from SUPPLIERS
- PNO from PARTS
- PROJNO from PROJECTS

b.)

entity integrity constraints

- PNO ,SNO, PROJNO from respectively PROJECT,SUPPLIER,PROJECT tables.
- In here Quantity is domain constraint

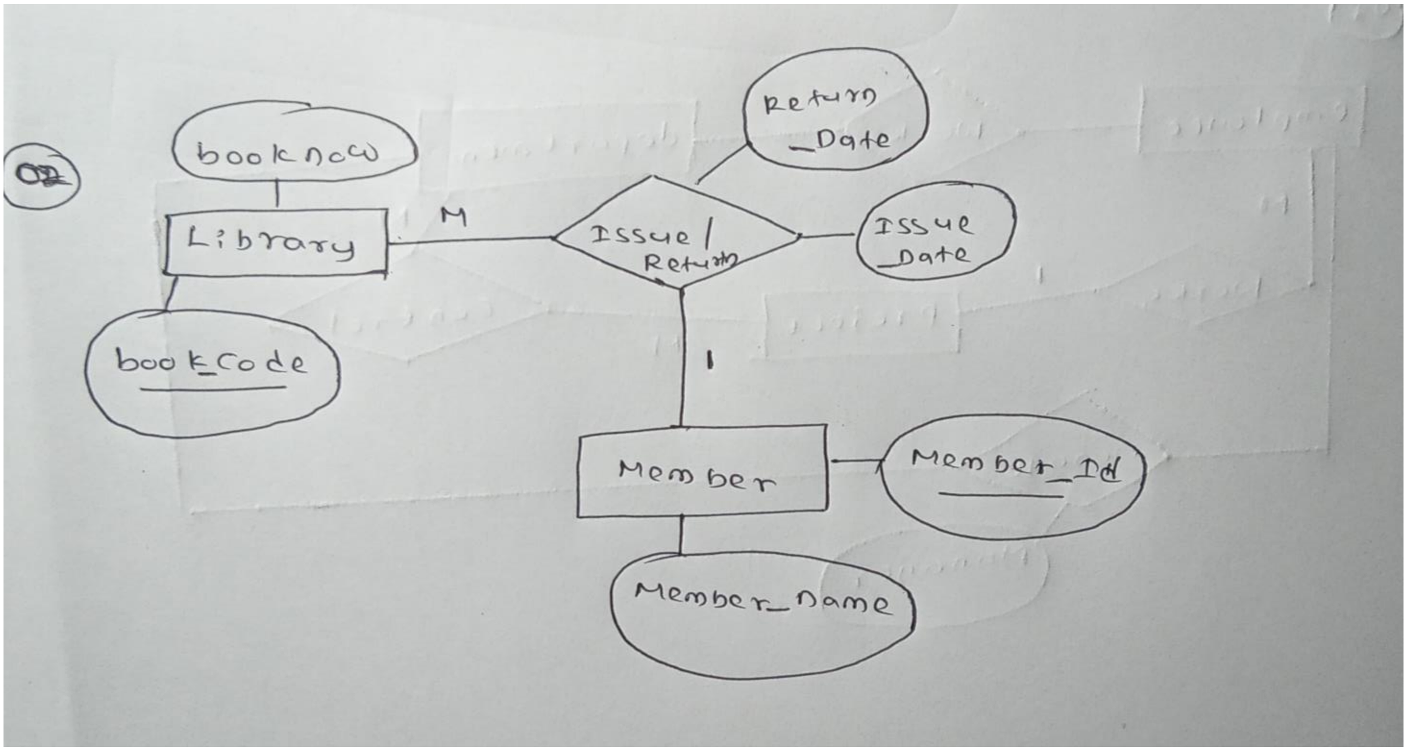
c.)

SUP_PAR_PROJ (SNO) reference SUPPLIER(SNO) where referential integrity constraint as foreign keys when we used it defined relationship between two tables.

d.)

It would prevent from assigning a supply to a parts an project for SUP_PAR_PROJ doesn't exist.

Question 02



Question 03

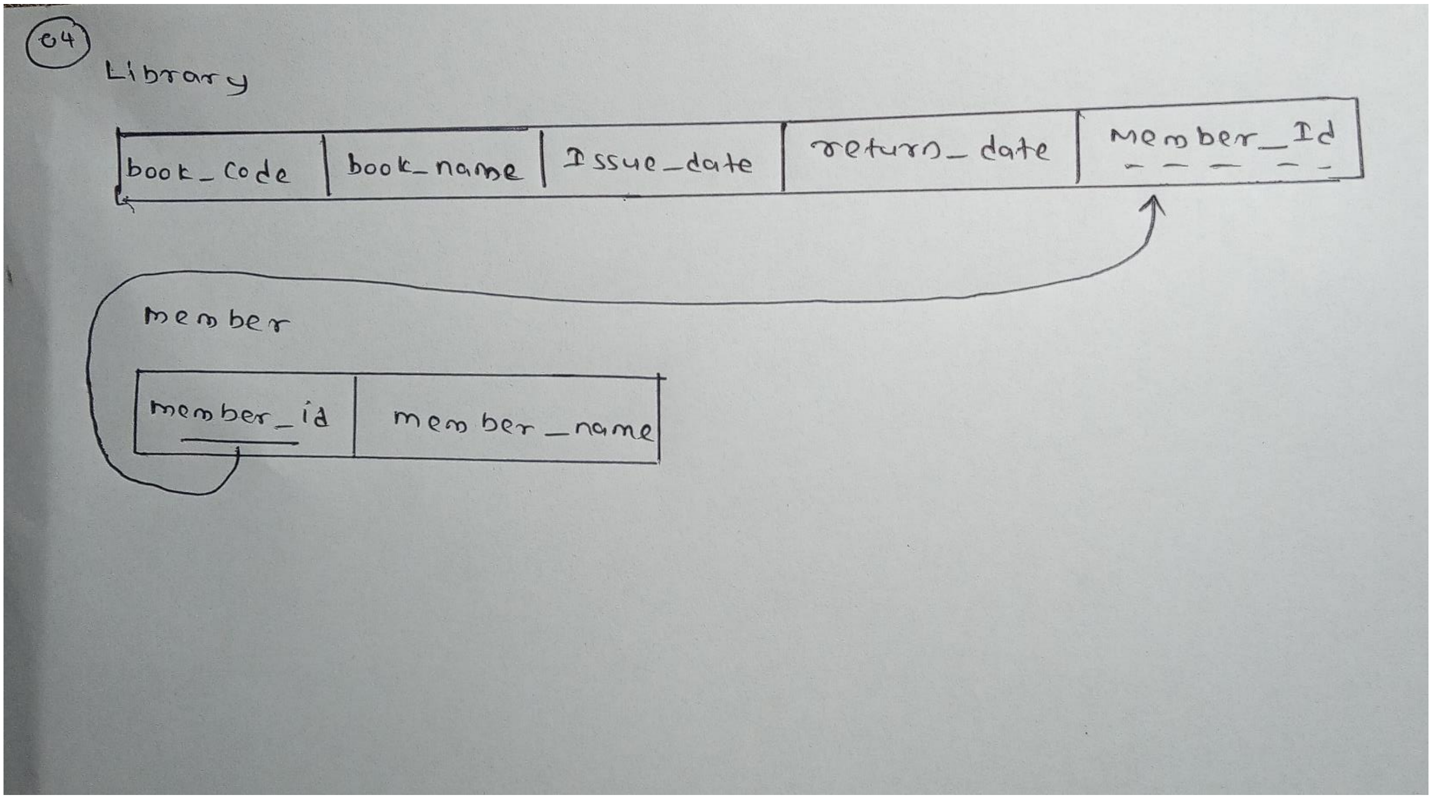
Functional dependence

- From member_id to member_name
- From member_id to issue_date and return_date

Domain constraints

- Issue_date and return_date

Question 04



04

Library

book_code	book_name	Issue_date	return_date	Member_Id

member

member_id	member_name