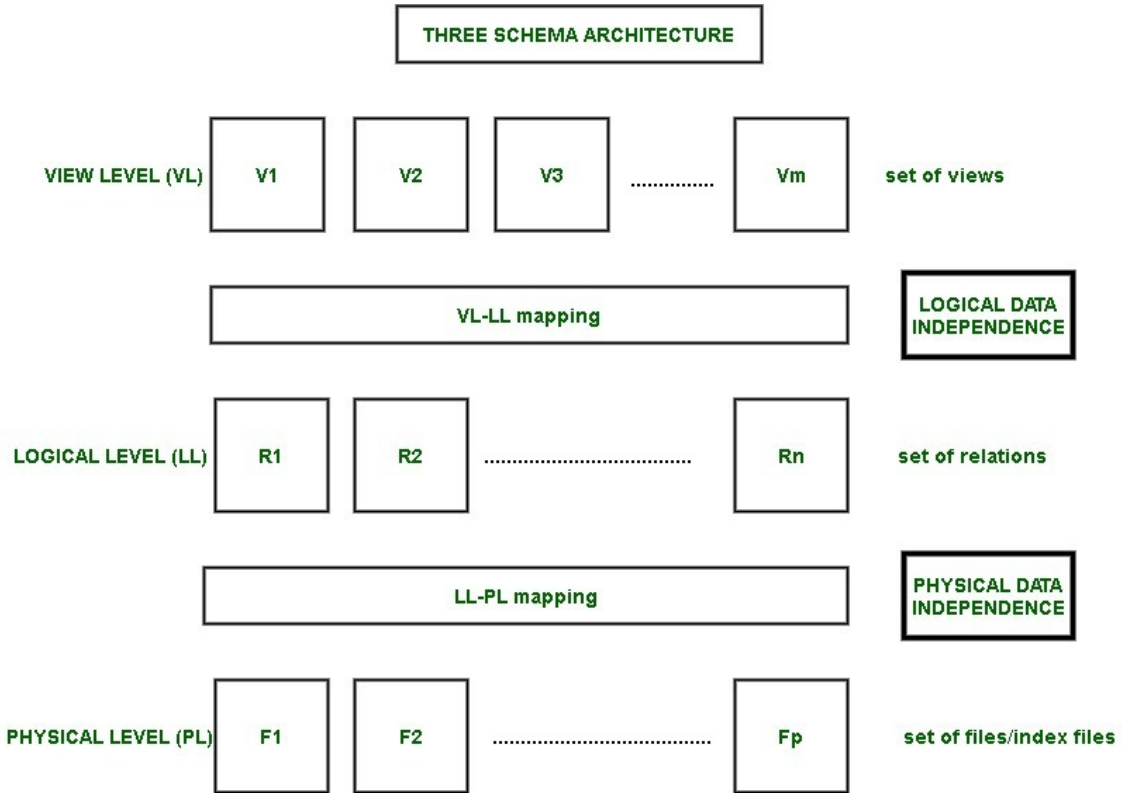
Course Code: KCS 501 **Roll No.:**
Course Name: DBMS **Time: 1.15 Hrs.**
Maximum Marks: 30
Instructions:

1. **Attempt All sections.**
2. **If require any missing data, then choose suitably.**

<table>
<tr><th>Q. No.</th><th>Question</th></tr>
<tr><td>1</td><td>Attempt ANY ONE part from the following</td></tr>
<tr><td>a)</td><td>List the advantages and disadvantages of file system over the database approach?</td></tr>
<tr><td></td><td>
<table>
<tr><th>Parameter</th><th>File System</th><th>DBMS</th></tr>
<tr><td>Definition</td><td>File system is a software that manages and organizes the files in a storage medium within a computer.</td><td>DBMS is a software for managing the database.</td></tr>
<tr><td>Data Redundancy</td><td>Redundant data can be present in a file system.</td><td>In DBMS there is no redundant data.</td></tr>
<tr><td>Backup and Recovery</td><td>It doesn’t provide backup and recovery of data if it is lost.</td><td>It provides backup and recovery of data even if it is lost.</td></tr>
<tr><td>Query Processing</td><td>There is no efficient query processing in file system.</td><td>Efficient query processing is there in DBMS.</td></tr>
<tr><td>Data Consistency</td><td>There is less data consistency in file system.</td><td>There is more data consistency because of the process of normalization.</td></tr>
<tr><td>Complexity</td><td>It is less complex as compared to DBMS.</td><td>It has more complexity in handling as compared to file system.</td></tr>
<tr><td>Security</td><td>File systems provide less security in comparison to DBMS.</td><td>DBMS has more security mechanisms as compared to file system.</td></tr>
<tr><td>Cost</td><td>It is less expensive than DBMS.</td><td>It has a comparatively higher cost than a file system.</td></tr>
</table>
</td></tr>
<tr><td>b)</td><td>Compare and Contrast between physical and logical data independence with example?</td></tr>
<tr><td></td><td>Data Independence, which means that upper levels are unaffected by changes at lower levels.

There are two types of data independence:
- Physical Data Independence
- Conceptual Data Independence</td></tr>
</table>

- **Physical Data Independence:** Any change in the physical location of tables and indexes should not affect the conceptual level or external view of data. This data independence is easy to achieve and implemented by most of the DBMS today.
- **Conceptual Data Independence:** The data at conceptual level schema and external level schema must be independent. This means a change in conceptual schema should not affect external schema. e.g., Adding or deleting attributes of a table should not affect the user's view of the table.



2 **Attempt ANY ONE part from the following**

a)

Given an instance of the STUDENTS relation as shown below:

StudentId	StudentName	StudentEmail	Student Age	CPI
2345	Shankar	shankar@math	**X**	9.4
1287	Swati	swati@ee	19	9.5
7853	Shankar	shankar@cse	19	9.4
9876	Swati	swati@mech	18	9.3
8765	Ganesh	ganesh@civil	19	8.7

What should not be the value of X. Justify your answer?

GATE2014

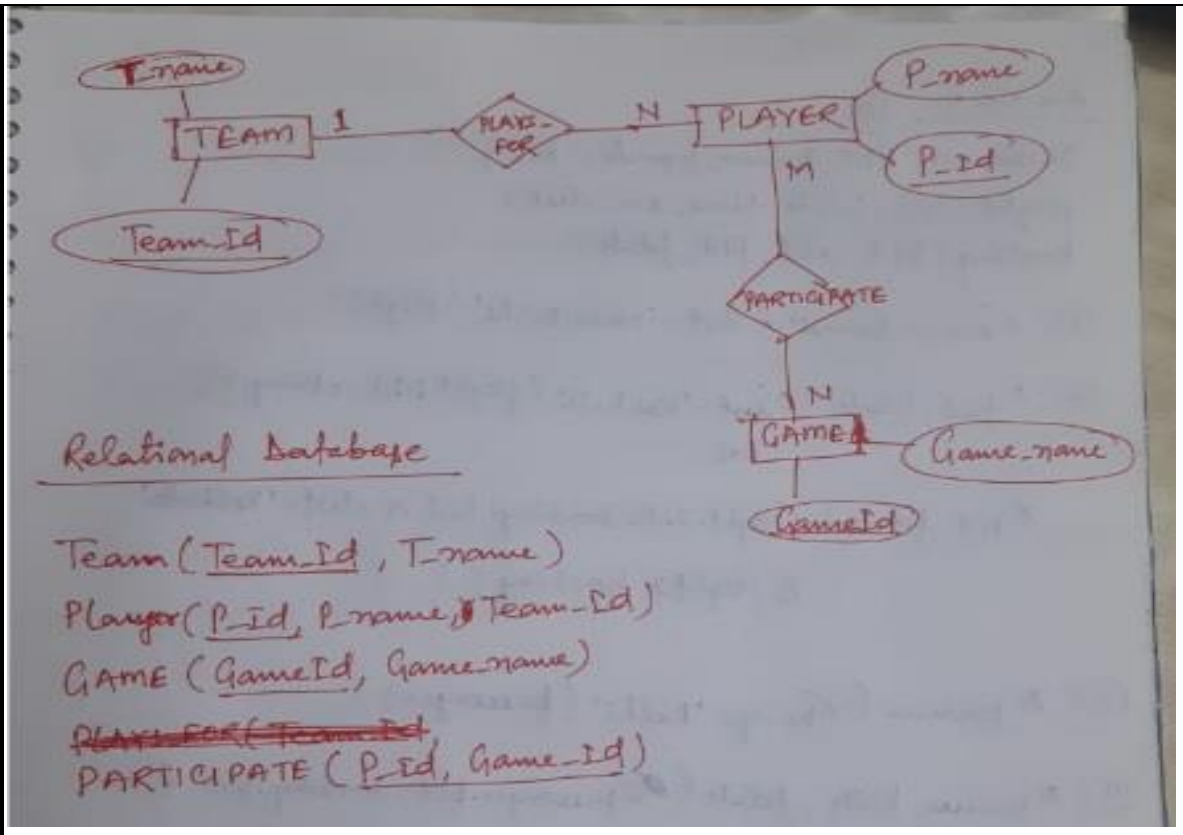
To determine whether STUDENTS(StudentName, StudentAge) can be a key for the given instance, we need to check if this combination of attributes uniquely identifies each tuple (row) in the relation. In a key, each combination of values must be unique.

The combination of StudentName and StudentAge should uniquely identify each student. Let's examine the conditions for uniqueness:

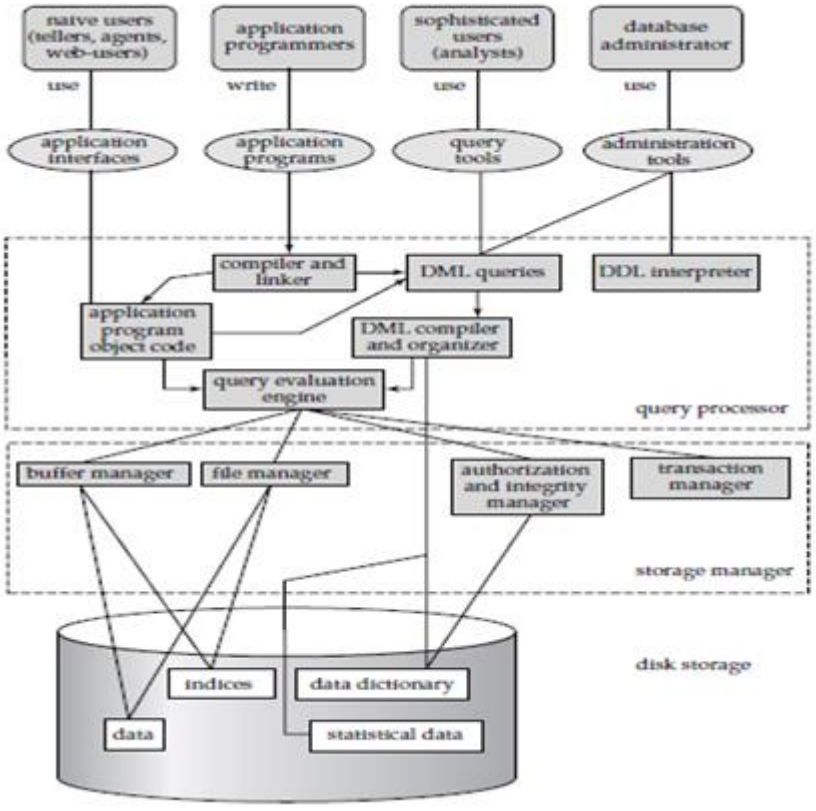
Shankar, X:

Appears twice in the table, so it's not unique.

	Swati, 19: **Appears twice in the table, so it's not unique.** **Shankar, 19:** **Appears once in the table.** **Swati, 18:** **Appears once in the table.** **Ganesh, 19:** **Appears once in the table.** **For STUDENTS(StudentName, StudentAge) to be a key, every combination of values in these columns must be unique. The only unique combination in this instance is (Shankar, 19). Therefore, Shankar's age, denoted by X, should not be equal to 19. If Shankar's age is 19, then the combination (Shankar, 19) would not be unique, violating the key constraint.** **So, Shankar's age (X) should not be equal to 19 for STUDENTS(StudentName, StudentAge) to be a key in this instance.**
b)	Define arity of relation in RDBMS. Also find the same for thHealthcare system: - • Patient(Patient ID, FirstName, LastName, DoB, Gender) • Doctor(DoctorID,FirstName,LastName, Specialization) • Appointment(AppointmentID,PatientID,DoctorId,Appointment Date, Status) • MedicalRecord(RecodID,PatientID,DoctorID,Diagnosis,Prescription) **GATE 2023**
	In Relational Database Management Systems (RDBMS), the arity of a relation refers to the number of attributes or columns in that relation. Arity is a fundamental concept in the database design and schema definition. The arity of a relation is essentially the number of properties or pieces of information associated with each record (or tuple) in that relation. **Let's find the arity for the relations in the Healthcare System:** • **Patient:PatientIDFirstNameLastNameDOB (Date of Birth)Gender Arity = 5** • **Doctor:DoctorIDFirstNameLastNameSpecialization Arity = 4** • **Appointment:AppointmentIDPatientIDDoctorIDAppointmentDateStatus Arity = 5** • **MedicalRecord:RecordIDPatientIDDoctorIDDiagnosisPrescription Arity = 5** **So, in the given Healthcare System, the arities of the relations are as follows:** • **Patient: Arity 5** • **Doctor: Arity 4** • **Appointment: Arity 5** • **MedicalRecord: Arity 5**
3	**Attempt ANY ONE part from the following**
a)	A database is being constructed to keep track of the teams and games of a sport league. A team has a number of players, not allof whom participate in each game. It is desired to keep track of players participating in each game for each team, the positions they play in that game and the result of the game. 1. Design an E-R schema diagram for this application. 2. Map the E-R diagram into relational model.



b) Draw overall structure of DBMS and explain its components in brief with neat diagram?



The functional components of a database system can be broadly divided into

- The Storage Manager and
- The Query Processor.

Storage Manager

The storage manager components include:

- **Authorization and integrity manager**
- **Transaction manager**
- **File manager**
- **Buffer manager**

The storage manager implements several data structures:

- ***Data files****,* which store the database itself.
- ***Data dictionary****,* which stores metadata about the structure of the database, in particular the schema of the database.
- ***Indices****,* which provide fast access to data items that hold particular values.

The Query Processor

The query processor components include

- ***DDL interpreter,*** which interprets DDL statements and records the definitions in the data dictionary.
- ***DML compiler,*** which translates DML statements in a query language into an evaluation plan consisting of low-level instructions that the query evaluation engine understands.
- **Query evaluation engine,** which executes low-level instructions generated by the DML compiler.

Database Users

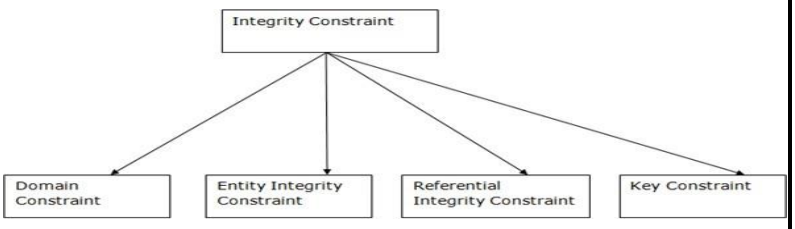
- Naïve Users
- Sophisticated Users
- Application Programmers
- Database Administrator

4	**Attempt ANY ONE part from the following**
a)	What are different integrity constraints? Explain each one of them briefly?

Integrity Constraints

- Integrity constraints are a set of rules. It is used to maintain the quality of information.
- Integrity constraints ensure that the data insertion, updating, and other processes haveto be performed in such a way that data integrity is not affected.
- Thus, integrity constraint is used to guard against accidental damage to the database. Types of Integrity Constraint

Types of Integrity Constraint



1. Domain Constraints:

- Domain constraints can be defined as the definition of a valid set of values for anattribute.
- It includes data types, length, size, range, not null, default values and check values. The data type of domain includes string, character, integer, time, date, currency, etc.
- It ensures that the values of the attributes must be available in the corresponding domain.

Example:

ID	NAME	SEMENSTER	AGE
1000	Tom	1st	17
1001	Johnson	2nd	24
1002	Leonardo	5th	21
1003	Kate	3rd	19
1004	Morgan	8th	A

Not allowed. Because AGE is an integer attribute

2. Entity integrity constraints

- The entity integrity constraint states that primary key value can't be null.
- This is because the primary key value is used to identify individual rows in relationand if the primary key has a null value, then we can't identify those rows.
- A table can contain a null value other than the primary key field.

Example:

EMPLOYEE

EMP_ID	EMP_NAME	SALARY
123	Jack	30000
142	Harry	60000
164	John	20000
	Jackson	27000

Not allowed as primary key can't contain a NULL value

3. Referential Integrity Constraints

- The referential integrity constraint is specified between two relations and is used to maintain the consistency among tuples in the two relations. Informally, the referential integrity constraint states that a tuple in one relation that refers to another relation mustrefer to an existing tuple in that relation.
- In the Referential integrity constraints, if a foreign key in Table 1 refers to the Primary Key of Table 2, then every value of the Foreign Key in Table 1 must be null or be available in Table 2.

Example:

(Table 1)

EMP_NAME	NAME	AGE	D_No
1	Jack	20	11
2	Harry	40	24
3	John	27	18
4	Devil	38	13

Foreign key

Not allowed as D_No 18 is not defined as a Primary key of table 2 and In table 1, D_No is a foreign key defined

Relationships

(Table 2)

Primary Key

D_No	D_Location
11	Mumbai
24	Delhi
13	Noida

A set of attributes FK in relation schema R1 is a foreign key of R1 that references relationR2 if it satisfies the following rules:

1. Attributes in FK have the same domain(s) as the primary key attributes PK of R2;the attributes FK are said to reference or refer to the relation R2.
2. A value of FK in a tuple t1 of the current state r1(R1) either occurs as a value of PKfor some tuple t2 in the current state r2(R2) or is NULL.

In the former case, we have t1[FK] = t2[PK], and we say that the tuple t1 references or refers to the tuple t2.

In this definition, R1 is called the referencing relation and R2 is the referenced relation. If these two conditions hold, a referential integrity constraint from R1 to R2 is said to hold.

4. Key constraints

- Keys are the entity set that is used to identify an entity within its entity set uniquely.
- An entity set can have multiple keys, but out of which one key will be the primary key.A primary key can contain a unique and null value in the relational table.

Example:

ID	NAME	SEMENSTER	AGE
1000	Tom	1st	17
1001	Johnson	2nd	24
1002	Leonardo	5th	21
1003	Kate	3rd	19
1002	Morgan	8th	22

Not allowed. Because all row must be unique

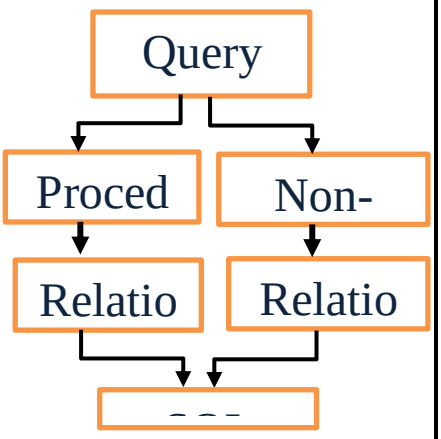
b) Define database language? List out different types of datatypes in SQL?

Define database language

- A Language which is used to store, manipulate and retrieve data from database is known as DB language. For example – SQL
- Language having expression and Query by which the user request some information from the database

There are two types of DB query language:

- Procedural Query language
- Non-procedural query language



Procedural Query language:

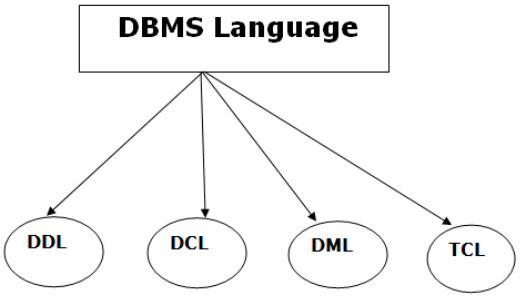
In procedural query language, user instructs the system to perform a series of operations to produce the desired results. ***Here users tell what data to be retrieved from database and how to retrieve it.***

Non-procedural query language:

In Non-procedural query language, user instructs the system to produce the desired result without telling the step by step process. ***Here users tells what data to be retrieved from database but doesn't tell how to retrieve it.***

- Relational algebra and calculus are the theoretical concepts used on relational model.
- SQL is a practical implementation of relational algebra and calculus. It is a non-procedural language.

SQL-DB Language:



Data Definition Language (DDL)

DDL stands for **D**ata **D**efinition **L**anguage. It is used to define database structure or pattern. It is used to create schema, tables, indexes, constraints, etc. in the database. Using the DDL statements, you can create the structure and skeleton of the database. Data definition language is used to store the information of metadata like the number of tables and schemas, their names, indexes, columns in each table, constraints, etc.

DDL Commands:

1. **Create:** It is used to create objects in the database.
2. **Alter:** It is used to alter the structure of the database.
3. **Drop:** It is used to delete objects from the database.
4. **Truncate:** It is used to remove all records from a table.
5. **Rename:** It is used to rename an object.
6. **Comment:** It is used to comment on the data dictionary.

Data Manipulation Language (DML)

DML stands for **D**ata **M**anipulation **L**anguage. It is used for accessing and manipulating data in a database. It handles user requests.

DML Commands

1. Select: It is used to retrieve data from a database.
2. Insert: It is used to insert data into a table.
3. Update: It is used to update existing data within a table.
4. Delete: It is used to delete all records from a table.
5. Lock Table: It controls concurrency.

Data Control Language (DCL)

DCL stands for Data Control Language. It is used to retrieve the stored or saved data. The DCL execution is transactional. It also has rollback parameters.

DCL Commands:

1. **Grant:** It is used to give user access privileges to a database.
2. **Revoke:** It is used to take back permissions from the user.

Transaction Control Language (TCL)

TCL is used to run the changes made by the DML statement. TCL can be grouped into a logical transaction.

TCL Commands:

1. **Commit:** It is used to save the transaction on the database.
2. **Rollback:** It is used to restore the database to original since the last Commit.

List out different types of data-types in SQL?

Numeric Data types

- Smallint
- Int
- Integer
- Number(p,s)
- Numeric(p,s)
- Decimal(p,s)
- Dec(p,s)
- float(p)
- real
- double precision

Character Data types

Type	Description	Byte Size
CHAR (s)	Fixed-length character with a length of s.	2,000 bytes.
NCHAR (s)	Fixed length national character with a length of s.	2,000 bytes.
VARCHAR(s)	Variable-length character string with a maximum length of s bytes	4000 bytes
VARCHAR2 (s)	Variable-length character string with a maximum length of s bytes	32,767 bytes
NVARCHAR2 (s)	Variable-length national character string with a maximum length of s bytes	32,767 bytes
LONG	Variable-length character string. Larger than VARCHAR2. Deprecated	2 GB
RAW (s)	Raw binary data of length size	32,767 bytes
LONG RAW	Raw binary data of variable length. Deprecated	2 GB
BLOB	Stores large unstructured binary objects. Recommended for JSON fields.	8 TB to 128 TB
CLOB	Stores large single-byte and multi-byte objects.	8 TB to 128 TB

Other Data Types

Logical Data Types	BOOLEAN	Currently only supported for accounts provisioned after January 25, 2016.
Date & Time Data Types	DATE	
	DATETIME	Alias for TIMESTAMP_NTZ
	TIME	
	TIMESTAMP	Alias for one of the TIMESTAMP variations (TIMESTAMP_NTZ by default).
	TIMESTAMP_LTZ	TIMESTAMP with local time zone; time zone, if provided, is not stored.
	TIMESTAMP_NTZ	TIMESTAMP with no time zone; time zone, if provided, is not stored.
	TIMESTAMP_TZ	TIMESTAMP with time zone.
Semi-structured Data Types	VARIANT	
	OBJECT	
	ARRAY	

5 **Attempt ANY ONE part from the following**

a) Consider the following relational database schema consisting of the fouelation schemas:
passenger (pid, pname, pgender, pcity)flight (fid, fdate, time, src, dest) booking (pid, aid, fid, fdate)

Write relational algebra queries for the following;

1. Get the details about all flights from Chennai to New Delhi.
2. Get the Flight id and flight date of all flights to Kolkata.
3. Get the passengers name, who lived in Delhi.
4. Get the passenger name, passenger city and flight date who booke dthe flight.
5. Rename pid as pes_id attribute of the passenger relation.

1. **Get the details about all flights from Chennai to New Delhi.**

σ (src='Chennai' $\wedge$ dest = 'New Delhi') (flight)

2. **Get the Flight id and flight date of all flights to Kolkata.**

Π fid, fdate (σ (dest = 'Kolkata') (flight))

3. **Get the passengers name, who lived in Delhi.**

Π pname (σ (pcity= 'Delhi') (passenger))

4. **Get the passenger name, passenger city and flight date who booked the flight.**

Π pname, pcity, fdate (passenger * booking) **// Natural join**

Or

Πpname, pcity, fdate (σ pid=pid (passenger X booking) **// Equi join**

5. **Rename pid as pes_id**

ρ**{pes_id/pid}**(passenger)

Or

ρ**{pes_id, pname, pgender, pcity }**(passenger)

// only pid is changed and other will be same

b)

Define DDL? Consider a Table - **Emp (EID, Ename, Esal)**
Write the SQL query for table Emp--

1. Add one column (Eadd) to the relation Emp
2. Add constraints to Eid as PRIMARY KEY and Eadd asUNIQUE.
3. Modify the type of the column (Varchar2(50)) of EMP.
4. Drop column Esal of Emp.
5. Truncate table EMP

Define DDL:

DDL stands for **D**ata **D**efinition **L**anguage. It is used to define database structure or pattern. It is used to create schema, tables, indexes, constraints, etc. in the database. Using the DDL statements, you can create the structure and skeleton of the database. Data definition language is used to store the information of metadata like the number of tables and schemas, their names, indexes, columns in each table, constraints, etc.

DDL Commands:

1. **Create:** It is used to create objects in the database.
2. **Alter:** It is used to alter the structure of the database.

	3. **Drop:** It is used to delete objects from the database. 4. **Truncate:** It is used to remove all records from a table. 5. **Rename:** It is used to rename an object. 6. **Comment:** It is used to comment on the data dictionary. **1. Add one column (Eadd) to the relation Emp**
	ALTER TABLE emp *ADD* (Eadd varchar2(50)) // Oracle 10g
	2. Add constraints to Eid as PRIMARY KEY and Eadd as UNIQUE.
	ALTER TABLE emp MODIFY (EID PRIMARY KEY, Eadd UNIQUE) // Oracle 10g
	3. Modify the type of the column (Varchar2(50)) of EMP.
	ALTER TABLE emp MODIFY Ename Varchar2 (50) // Oracle 10g ***(// if students attempt for all or any column then it would be correct)***
	4. Drop column Esal of Emp.
	ALTER TABLE emp DROP COLUMN Esal // Oracle 10g
	5. Truncate table EMP
	TRUNCATE TABLE emp // Oracle 10g

CO Course Outcomes mapped with respective question
KL Bloom's knowledge Level (K1, K2, K3, K4, K5, K6)
K1- Remember, K2- Understand, K3-Apply, K4- Analyze, K5: Evaluate, K6- Create