

Assignment - 2

SECTION - A →

- ① (C) Relational algebra
- ② (d) Schema & instance
- ③ (C) Assignment
- ④ (b) Sigma (greek)
- ⑤ (a) Predicts, relation
- ⑥ (b) $\{t | P(t)\}$
- ⑦ (b) DDL
- ⑧ (C) Fixed, variable
- ⑨ (b) Cartesian product
- ⑩ (b) Intersect

SECTION - B →

- ① SQL includes data definition language (DDL) Statements and data manipulation language (DML) Statements.
- ② Relational algebra is a procedural query language which takes relation as input &

generates relation as output. Relational algebra operators & ch. of relational algebra operators.

③ Relational calculus is a declarative language. TRC uses tuple variable & checks every row with predicate expression condition.

There are 2 types of relational algebra →

- (i) tuple relational calculus.
- (ii) domain relational calculus.

④ A data dictionary is a collection of names, definition and attributes about data elements that are being used or captured in a database.

SECTION-C

① There are several types of integrity constraints commonly used in relational databases →

(i) Entity integrity constraints → These constraints ensure that each row in a table represents a unique entity by specifying a primary key. A primary key is a unique identifier for a row in a table & can consist of one or more columns.

- (ii) Referential integrity constraints → These const. establish relationships b/w tables by defining foreign keys. A foreign key is a column in a table that refers to the primary key for another table.
- (iii) Domain integrity constraints → These constraints define the allowed values of data types for columns in a table.
- (iv) Check constraints → These constraints specify whether a column can have a null value, which represents the absence of a value.
- (v) Unique constraints → These constraints ensure that the value in a column, or combination of columns, are unique within the table.
- (vi) Null constraints → Null constraints ensure that columns that are req. that a value are not left empty, while allowing for optional data.
- (vii) Multi-table constraints → These constraints involve multiple tables & define relationships and dependencies b/w them.

② Union, Set difference & Set intersection are 3 fundamental Set operations in relational algebra, which is a mathematical query used to manipulate & retrieve data from relational databases.

(i) Union (\cup) \rightarrow The union operations combines the rows of two or more tables, or relational into a single result set.

$R \cup S$, where R & S are two relations.

(ii) Set difference \rightarrow The Set difference operation, also known as minus or except, return the rows that are present in one table but not in another.

\rightarrow It is denoted by Symbol " $-$ ".

$R - S$, where R & S are two relations.

(iii) Set intersection (\cap) \rightarrow The Set intersection operation return the common rows that are present in both table & rows.

$R \cap S$, where R & S are 2 relations.

SECTION - 9 ->

① Cartesian product & natural join are two operations in relational algebra that involve combining tables or relations in a database.

(i) Cartesian product \rightarrow The Cartesian product of two tables, denoted by symbol ' \times ' or by using the CROSS JOIN operation in relational algebra, returns a new table that contains all possible combinations of rows from the two original tables.

Eg \rightarrow Consider two tables A & B \rightarrow

table A \rightarrow

A1	A2
1	x
2	y

table B \rightarrow

B1	B2
10	p
20	q

$A \times B \rightarrow$

A1	A2	B1	B2
1	x	10	p
1	x	20	q
2	y	10	p
2	y	20	q

(ii) Natural Join \rightarrow The natural join operation, denoted by the symbol \bowtie or by using the JOIN operation in relational algebra, is a type of join that combines two tables based on their common attributes without explicitly specifying the join condition.

Eg \rightarrow consider two tables A & B \rightarrow

table A \rightarrow

A1	A2	A3
1	x	100
2	y	200

table B \rightarrow

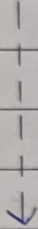
A1	B1	B2
1	10	P
2	20	Q

A \bowtie B :

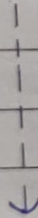
A1	A2	A3	B1	B2
1	x	100	10	P
2	y	200	20	Q

② The relational Schema diagram from a given database, including the foreign keys, would look like this →

STUDENT		COURSE
SSN (PK)	←-----→	Course # (PK)
Name		Cname
6 data		Dept



ENROLL	BOOK-ADOPTION
SSN (FK)	Course # (FK)
Course # (FK)	Quarter (FK)
Quarter	Book ISBN
Grade	



TEXT
Book - ISBN (PK)
Book - Title
Publisher
Author

- Primary Key (PK) columns are underlined.
- Foreign Key (FK) columns are indicated with an arrow pointing towards the ref. primary key.
- The STUDENT table has a primary key column SSN.