**RELATIONAL ALGEBRA** is a widely used procedural query language. It collects instances of relations as input and gives occurrences of relations as output. It uses various operations to perform this action. SQL Relational algebra query operations are performed recursively on a relation. The output of these operations is a new relation, which might be formed from one or more input relations.

**Basic SQL Relational Algebra Operations**

Relational Algebra divided in various groups

**Unary Relational Operations**

* SELECT (symbol: σ)
* PROJECT (symbol: π)
* RENAME (symbol: ρ)

**Relational Algebra Operations From Set Theory**

* UNION (υ)
* INTERSECTION ( ),
* DIFFERENCE (-)
* CARTESIAN PRODUCT ( x )

**Binary Relational Operations**

* JOIN
* DIVISION

Let’s study them in detail with solutions:

**SELECT (σ)**

The SELECT operation is used for selecting a subset of the tuples according to a given selection condition. Sigma(σ)Symbol denotes it. It is used as an expression to choose tuples which meet the selection condition. Select operator selects tuples that satisfy a given predicate.

σp(r)

σ is the predicate

r stands for relation which is the name of the table

p is prepositional logic

**Example 1**

σ topic = "Database" (Tutorials)

**Output** – Selects tuples from Tutorials where topic = ‘Database’.

**Example 2**

σ topic = "Database" and author = "punam"( Tutorials)

**Output** – Selects tuples from Tutorials where the topic is ‘Database’ and ‘author’ is punam.

**Example 3**

σ sales > 50000 (Customers)

**Output** – Selects tuples from Customers where sales is greater than 50000

**Projection(π)**

The projection eliminates all attributes of the input relation but those mentioned in the projection list. The projection method defines a relation that contains a vertical subset of Relation.

This helps to extract the values of specified attributes to eliminates duplicate values. (pi) symbol is used to choose attributes from a relation. This operator helps you to keep specific columns from a relation and discards the other columns.

**Example of Projection:**

Consider the following table

|  |  |  |
| --- | --- | --- |
| **CustomerID** | **CustomerName** | **Status** |
| 1 | Google | Active |
| 2 | Amazon | Active |
| 3 | Apple | Inactive |
| 4 | Alibaba | Active |

Here, the projection of CustomerName and status will give

Π CustomerName, Status (Customers)

|  |  |
| --- | --- |
| **CustomerName** | **Status** |
| Google | Active |
| Amazon | Active |
| Apple | Inactive |
| Alibaba | Active |

**Rename (ρ)**

Rename is a unary operation used for renaming attributes of a relation.

ρ (a/b)R will rename the attribute ‘b’ of relation by ‘a’.

**Union operation (υ)**

UNION is symbolized by ∪ symbol. It includes all tuples that are in tables A or in B. It also eliminates duplicate tuples. So, set A UNION set B would be expressed as:

The result <- A ∪ B

For a union operation to be valid, the following conditions must hold –

* R and S must be the same number of attributes.
* Attribute domains need to be compatible.
* Duplicate tuples should be automatically removed.

Example

Consider the following tables.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table A** | |  | **Table B** | |
| **column 1** | **column 2** |  | **column 1** | **column 2** |
| 1 | 1 |  | 1 | 1 |
| 1 | 2 |  | 1 | 3 |

A ∪ B gives

|  |  |
| --- | --- |
| **Table A ∪ B** | |
| **column 1** | **column 2** |
| 1 | 1 |
| 1 | 2 |
| 1 | 3 |

**Set Difference (-)**

– Symbol denotes it. The result of A – B, is a relation which includes all tuples that are in A but not in B.

* The attribute name of A has to match with the attribute name in B.
* The two-operand relations A and B should be either compatible or Union compatible.
* It should be defined relation consisting of the tuples that are in relation A, but not in B.

**Example**

A-B

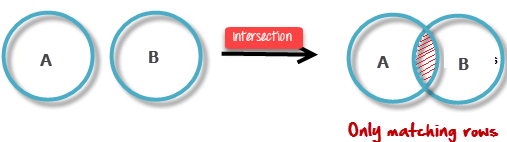
|  |  |
| --- | --- |
| **Table A – B** | |
| **column 1** | **column 2** |
| 1 | 2 |

**Intersection**

An intersection is defined by the symbol ∩

A ∩ B

Defines a relation consisting of a set of all tuple that are in both A and B. However, A and B must be union-compatible.

Visual Definition of Intersection

Example:

A ∩ B

|  |  |
| --- | --- |
| **Table A ∩ B** | |
| **column 1** | **column 2** |
| 1 | 1 |

**Cartesian Product(X) in DBMS**

**Cartesian Product in DBMS** is an operation used to merge columns from two relations. Generally, a cartesian product is never a meaningful operation when it performs alone. However, it becomes meaningful when it is followed by other operations. It is also called Cross Product or Cross Join.

**Example – Cartesian product**

σ column 2= ‘1’(A X B)

Output – The above example shows all rows from relation A and B whose column 2 has value 1

|  |  |
| --- | --- |
| **σ column 2 = ‘1’ (A X B)** | |
| **column 1** | **column 2** |
| 1 | 1 |
| 1 | 1 |

**Join Operations**

Join operation is essentially a cartesian product followed by a selection criterion.

Join operation denoted by ⋈.

JOIN operation also allows joining variously related tuples from different relations.

**Types of JOIN:**

Various forms of join operation are:

Inner Joins:

* Theta join
* EQUI join
* Natural join

Outer join:

* Left Outer Join
* Right Outer Join
* Full Outer Join

**Inner Join:**

In an inner join, only those tuples that satisfy the matching criteria are included, while the rest are excluded. Let’s study various types of Inner Joins:

**Theta Join:**

The general case of JOIN operation is called a Theta join. It is denoted by symbol **θ**

Example

A ⋈θ B

Theta join can use any conditions in the selection criteria.

For example:

A ⋈ A.column 2 > B.column 2 (B)

|  |  |
| --- | --- |
| **A ⋈ A.column 2 > B.column 2 (B)** | |
| **column 1** | **column 2** |
| 1 | 2 |

**EQUI join:**

When a theta join uses only equivalence condition, it becomes a equi join.

For example:

A ⋈ A.column 2 = B.column 2 (B)

|  |  |
| --- | --- |
| **A ⋈ A.column 2 = B.column 2 (B)** | |
| **column 1** | **column 2** |
| 1 | 1 |

EQUI join is the most difficult operations to implement efficiently using SQL in an RDBMS and one reason why RDBMS have essential performance problems.

**NATURAL JOIN (⋈)**

Natural join can only be performed if there is a common attribute (column) between the relations. The name and type of the attribute must be same.

Example

Consider the following two tables

|  |  |
| --- | --- |
| **C** | |
| **Num** | **Square** |
| 2 | 4 |
| 3 | 9 |

|  |  |
| --- | --- |
| **D** | |
| **Num** | **Cube** |
| 2 | 8 |
| 3 | 27 |

C ⋈ D

|  |  |  |
| --- | --- | --- |
| **C ⋈ D** | | |
| **Num** | **Square** | **Cube** |
| 2 | 4 | 8 |
| 3 | 9 | 27 |

**OUTER JOIN**

In an outer join, along with tuples that satisfy the matching criteria, we also include some or all tuples that do not match the criteria.

**Left Outer Join(A https://www.guru99.com/images/1/100518_0535_RelationalA5.png B)**

In the left outer join, operation allows keeping all tuple in the left relation. However, if there is no matching tuple is found in right relation, then the attributes of right relation in the join result are filled with null values.



Consider the following 2 Tables

|  |  |
| --- | --- |
| **A** | |
| **Num** | **Square** |
| 2 | 4 |
| 3 | 9 |
| 4 | 16 |

|  |  |
| --- | --- |
| **B** | |
| **Num** | **Cube** |
| 2 | 8 |
| 3 | 18 |
| 5 | 75 |

A https://www.guru99.com/images/1/100518_0535_RelationalA5.png B

|  |  |  |
| --- | --- | --- |
| **A ⋈ B** | | |
| **Num** | **Square** | **Cube** |
| 2 | 4 | 8 |
| 3 | 9 | 18 |
| 4 | 16 | – |

**Right Outer Join: ( A https://www.guru99.com/images/1/100518_0535_RelationalA7.png B )**

In the right outer join, operation allows keeping all tuple in the right relation. However, if there is no matching tuple is found in the left relation, then the attributes of the left relation in the join result are filled with null values.



A https://www.guru99.com/images/1/100518_0535_RelationalA7.png B

|  |  |  |
| --- | --- | --- |
| **A ⋈ B** | | |
| **Num** | **Cube** | **Square** |
| 2 | 8 | 4 |
| 3 | 18 | 9 |
| 5 | 75 | – |

**Full Outer Join: ( A https://www.guru99.com/images/1/100518_0535_RelationalA9.png B)**

In a full outer join, all tuples from both relations are included in the result, irrespective of the matching condition.

A https://www.guru99.com/images/1/100518_0535_RelationalA9.png B

|  |  |  |
| --- | --- | --- |
| **A ⋈ B** | | |
| **Num** | **Cube** | **Square** |
| 2 | 4 | 8 |
| 3 | 9 | 18 |
| 4 | 16 | – |
| 5 | – | 75 |

**Summary**

|  |  |
| --- | --- |
| **Operation(Symbols)** | **Purpose** |
| Select(σ) | The SELECT operation is used for selecting a subset of the tuples according to a given selection condition |
| Projection(π) | The projection eliminates all attributes of the input relation but those mentioned in the projection list. |
| Union Operation(∪) | UNION is symbolized by symbol. It includes all tuples that are in tables A or in B. |
| Set Difference(-) | – Symbol denotes it. The result of A – B, is a relation which includes all tuples that are in A but not in B. |
| Intersection(∩) | Intersection defines a relation consisting of a set of all tuple that are in both A and B. |
| Cartesian Product(X) | Cartesian operation is helpful to merge columns from two relations. |
| Inner Join | Inner join, includes only those tuples that satisfy the matching criteria. |
| Theta Join(θ) | The general case of JOIN operation is called a Theta join. It is denoted by symbol θ. |
| EQUI Join | When a theta join uses only equivalence condition, it becomes a equi join. |
| Natural Join(⋈) | Natural join can only be performed if there is a common attribute (column) between the relations. |
| Outer Join | In an outer join, along with tuples that satisfy the matching criteria. |
| Left Outer Join(https://www.guru99.com/images/1/100518_0535_RelationalA10.png ) | In the left outer join, operation allows keeping all tuple in the left relation. |
| Right Outer join(https://www.guru99.com/images/1/100518_0535_RelationalA11.png) | In the right outer join, operation allows keeping all tuple in the right relation. |
| Full Outer Join(https://www.guru99.com/images/1/100518_0535_RelationalA12.png) | In a full outer join, all tuples from both relations are included in the result irrespective of the matching condition. |

**The relational algebra query language is \_\_\_.**

1. Analytical
2. Procedural
3. Symmetrical
4. Instrumental

**Answer:** B) Procedural

**Explanation:**

The relational algebra query language is procedural.

[Discuss this Question](https://www.includehelp.com/mcq/the-relational-algebra-query-language-is.aspx)

**2. In Relational Algebra, queries are performed using \_\_\_.**

1. Entities
2. Relationships
3. Operators
4. Objects

**Answer:** C) Operators

**Explanation:**

In Relational Algebra, queries are performed using operators.

[Discuss this Question](https://www.includehelp.com/mcq/in-relational-algebra-queries-are-performed-using.aspx)

**3. How many types of relational operations are there?**

1. 6
2. 7
3. 8
4. 9

**Answer:** B) 7

**Explanation:**

There are 7 relational operations.

1. Select Operation
2. Project Operation
3. Union Operation
4. Set Intersection
5. Set Difference
6. Cartesian Product
7. Rename Operation

[Discuss this Question](https://www.includehelp.com/mcq/how-many-types-of-relational-operations-are-there.aspx)

**4. Which of the following is NOT a type of relational operation?**

1. Select Operation
2. Set Difference
3. Set Update
4. Cartesian Product

**Answer:** C) Set Update

**Explanation:**

Set Update is NOT a type of relational operation.

[Discuss this Question](https://www.includehelp.com/mcq/which-of-the-following-is-not-a-type-of-relational-operation.aspx)

**5. Which of the following is a type of relational operation?**

1. Project Operation
2. Union Operation
3. Set Difference
4. All of the above

**Answer:** D) All of the above

**Explanation:**

The types of relational operations are -

1. Select Operation
2. Project Operation
3. Union Operation
4. Set Intersection
5. Set Difference
6. Cartesian Product
7. Rename Operation

[Discuss this Question](https://www.includehelp.com/mcq/which-of-the-following-is-a-type-of-relational-operation.aspx)

**6. Using the select operation, you can select \_\_\_ that satisfy certain criteria.**

1. Tuples
2. Attributes
3. Operators
4. None

**Answer:** A) Tuples

**Explanation:**

Using the select operation, you can select tuples that satisfy certain criteria.

[Discuss this Question](https://www.includehelp.com/mcq/using-the-select-operation-you-can-select-that-satisfy-certain-criteria.aspx)

**7. Select operation is denoted by -**

1. σ
2. R
3. P
4. S

**Answer:** A) σ

**Explanation:**

Select operation is denoted by σ.

[Discuss this Question](https://www.includehelp.com/mcq/select-operation-is-denoted-by.aspx)

**8. Propositional logic is conducted using p as a formula, using connectors like -**

1. AND
2. OR
3. NOT
4. All of the above

**Answer:** D) All of the above

**Explanation:**

Propositional logic is conducted using p as a formula, using connectors like: AND, OR and NOT.

[Discuss this Question](https://www.includehelp.com/mcq/propositional-logic-is-conducted-using-p-as-a-formula-using-connectors-like.aspx)

**9. A list of the attributes we wish to appear in the result is displayed in \_\_\_ operation.**

1. Select
2. Project
3. Union
4. Rename

**Answer:** B) Project

**Explanation:**

A list of the attributes we wish to appear in the result is displayed in Project operation.

[Discuss this Question](https://www.includehelp.com/mcq/a-list-of-the-attributes-we-wish-to-appear-in-the-result-is-displayed-in-operation.aspx)

**10. What is Project Operation denoted by –**

1. P
2. R
3. O
4. ∏

**Answer:** D) ∏

**Explanation:**

Project Operation is denoted by ∏.

[Discuss this Question](https://www.includehelp.com/mcq/what-is-project-operation-denoted-by.aspx)

**11. If there are 2 types of tuples, A & B, the \_\_\_ operation contains all the tuples that are either in A or B or both in A & B.**

1. Project
2. Select
3. Union
4. Rename

**Answer:** C) Union

**Explanation:**

If there are 2 types of tuples, A & B, the union operation contains all the tuples that are either in A or B or both in A & B.

[Discuss this Question](https://www.includehelp.com/mcq/if-there-are-2-types-of-tuples-a-b-the-operation-contains-all-the-tuples-that.aspx)

**12. Union operation eliminates the \_\_\_ tuples.**

1. Simple
2. Single
3. Duplicate
4. NULL

**Answer:** C) Duplicate

**Explanation:**

Union operation eliminates the Duplicate tuples.

[Discuss this Question](https://www.includehelp.com/mcq/union-operation-eliminates-the-tuples.aspx)

**13. Union operation is denoted by -**

1. U
2. ^
3. \*
4. !

**Answer:** A) U

**Explanation:**

Union operation is denoted by U.

[Discuss this Question](https://www.includehelp.com/mcq/union-operation-is-denoted-by.aspx)

**14. The following conditions must be met by a union operation.**

1. There must be a common attribute between A and B.
2. A duplicate tuple is automatically discarded.
3. Both A. and B.
4. None of the above

**Answer:** C) Both A. and B.

**Explanation:**

The following conditions must be met by a union operation:

1. There must be a common attribute between A and B.
2. A duplicate tuple is automatically discarded.

[Discuss this Question](https://www.includehelp.com/mcq/the-following-conditions-must-be-met-by-a-union-operation.aspx)

**15. If all the tuples are there in the given two tuples, what operation is it called?**

1. Union
2. Select
3. Rename
4. Set Intersection

**Answer:** D) Set Intersection

**Explanation:**

If all the tuples are there in the given two tuples, this operation is called Set Intersection.

[Discuss this Question](https://www.includehelp.com/mcq/if-all-the-tuples-are-there-in-the-given-two-tuples-what-operation-is-it-called.aspx)

**16. What does Set Intersection Operation denote as?**

1. U
2. ∩
3. ∏
4. I

**Answer:** B) ∩

**Explanation:**

Set Intersection operation is denoted by ∩.

[Discuss this Question](https://www.includehelp.com/mcq/what-does-set-intersection-operation-denote-as.aspx)

**17. What does Set Difference Operation denote as?**

1. +
2. -
3. U
4. ∩

**Answer:** B) -

**Explanation:**

Set Difference Operation is denoted as (-).

[Discuss this Question](https://www.includehelp.com/mcq/what-does-set-difference-operation-denote-as.aspx)

**18. In \_\_\_ products, the rows in one table are combined with the rows in another table.**

1. Cross
2. Cartesian
3. Both A and B
4. None of the above

**Answer:** C) Both A and B

**Explanation:**

In Cartesian or Cross products, the rows in one table are combined with the rows in another table.

[Discuss this Question](https://www.includehelp.com/mcq/in-products-the-rows-in-one-table-are-combined-with-the-rows-in-another-table.aspx)

**19. Renaming the output relation is done via the \_\_\_ operation.**

1. Update
2. Alter
3. Rename
4. Reverse

**Answer:** C) Rename

**Explanation:**

Renaming the output relation is done via the rename operation.