Operators:

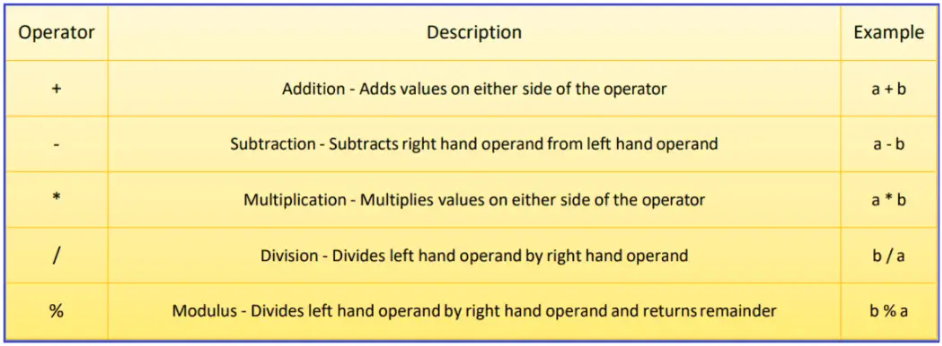
Operators are used to express the conditions in Select statements. The operator manipulates individual data items and returns a result. The data items are called operands or arguments.

* **Arithmetic operators**
* **Assignment operator**
* **Relational operators**
* **Logical operators**
* **Special Operators**
* **Set Operators**

**Arithmetic Operators in Oracle:**

The Arithmetic Operators in Oracle are used for performing mathematical calculations such as Addition, Subtraction, Multiplication, Module, and Division represented by the expected +, -, \* , %, and / respectively on the given operand values.

1. The arithmetic operations can be used to create expressions on number and date data.
2. The arithmetic operators can be used in any clause of a SQL statement.
3. SQL \* Plus ignores the blank spaces before and after the arithmetic operator.



 Operator Precedence

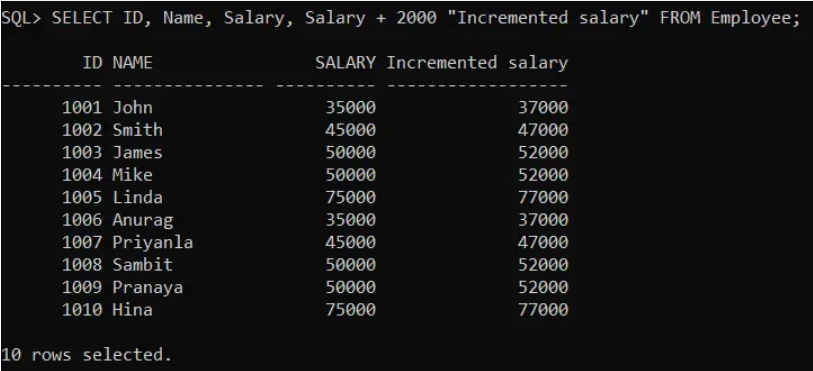
If the arithmetic expression contains more than one operator, then the multiplication and division operators are the highest priority and are evaluated first, and then the addition and minus (subtraction) operators are evaluated.

When two operators are having the same priority, then the expression is going to be evaluated from left to right in the Oracle database. The Parentheses in Oracle can also be used to force an operation to take priority over any other operators.

##### ****Arithmetic Operator Addition (+)****

Example: Display salary of employees with 2000 increments in their salary.

**SELECT ID, Name, Salary, Salary + 2000 “Incremented salary” FROM Employee;**

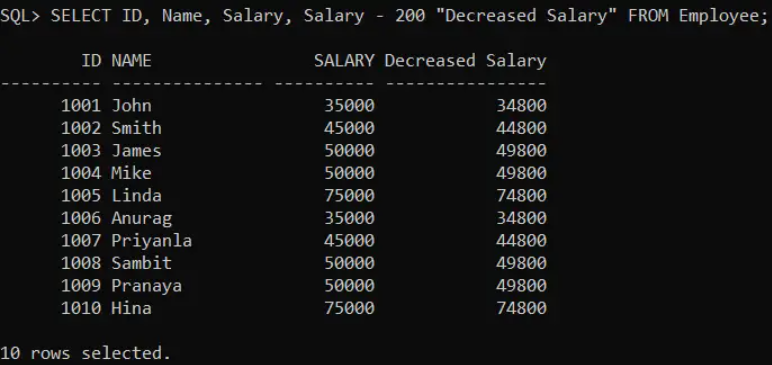


##### ****Arithmetic Operator Subtraction (-):****

This operator is used to perform the subtraction between two numbers

**Example:**Display the details of employees decreasing their salary by 200.

**SELECT ID, Name, Salary, Salary – 200 “Decreased Salary” FROM Employee;**

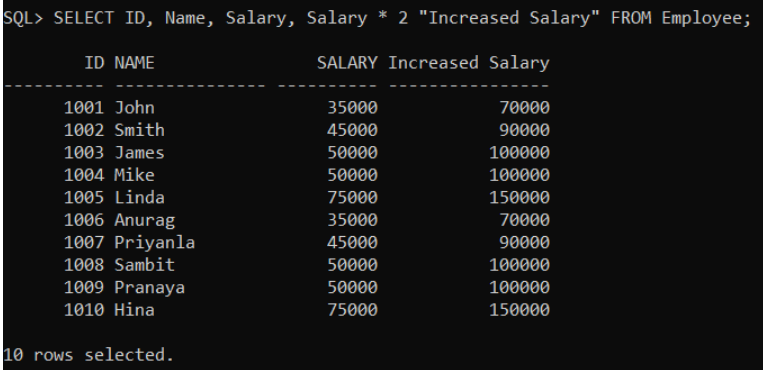


##### ****Arithmetic Operator Multiplication (\*):****

This operator is used to perform multiplication.

**Example**: Display the details of the employees Incrementing their salary two times.

**SELECT ID, Name, Salary, Salary \* 2 “Increased Salary” FROM Employee;**

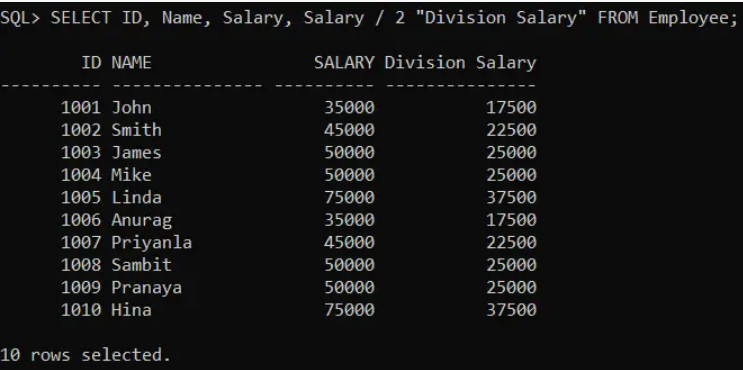


##### ****Arithmetic Operator Division (/):****

This operator is used to perform a Division test. The division will display only the Quotient value, not the remainder value

**Example**: Display half of the salary of employees.

**SELECT ID, Name, Salary, Salary / 2 “Division Salary” FROM Employee;**



In oracle, it is not possible to perform any calculations on String values (such as Varchar or Char).

##### ****Assignment Operator in Oracle****

The Assignment Operator in Oracle is used to assign or compare a value to a column or a field of a table. The equal sign (=) is the assignment operator where the value on the right is assigned to the value on the left.

Relational Operators:

 operators are used for comparing one expression with another expression. The relational operators determine whether the two values are equal or a value is greater than the other, or less than the other. The result of a comparison can be TRUE, FALSE, or NULL (When one or both the expressions contain NULL values).

##### ****Types of Relational Operators in Oracle****

1. **Not Equal (!= or <>) Operator**
2. **Greater Than (>) Operator**
3. **Less Than (<) Operator**
4. **Greater Than or Equal To (>=) Operator**
5. **Less Than or Equal To (<=) Operator**

SET linesize 300;

CREATE TABLE Employee (

Id INT PRIMARY KEY,

Name VARCHAR(15),

Department VARCHAR(10),

Salary NUMBER(8, 2),

Gender VARCHAR(10),

Age INT,

City VARCHAR(10)

);

INSERT INTO Employee (Id, Name, Department, Salary, Gender, Age, City) VALUES (1001, 'John', 'IT', 35000, 'Male', 25, 'London');

INSERT INTO Employee (Id, Name, Department, Salary, Gender, Age, City) VALUES (1002, 'Smith', 'HR', 45000, 'Female', 27, 'Mumbai');

INSERT INTO Employee (Id, Name, Department, Salary, Gender, Age, City) VALUES (1003, 'James', 'Finance', 50000, 'Male', 28, 'Delhi');

INSERT INTO Employee (Id, Name, Department, Salary, Gender, Age, City) VALUES (1004, 'Mike', 'Finance', 50000, 'Male', 28, 'London');

INSERT INTO Employee (Id, Name, Department, Salary, Gender, Age, City) VALUES (1005, 'Linda', 'HR', 75000, 'Female', 26, 'Mumbai');

INSERT INTO Employee (Id, Name, Department, Salary, Gender, Age, City) VALUES (1006, 'Anurag', 'IT', 35000, 'Male', 25, 'London');

INSERT INTO Employee (Id, Name, Department, Salary, Gender, Age, City) VALUES (1007, 'Priyanla', 'HR', 45000, 'Female', 27, 'Mumbai');

INSERT INTO Employee (Id, Name, Department, Salary, Gender, Age, City) VALUES (1008, 'Sambit', 'IT', 50000, 'Male', 28, 'London');

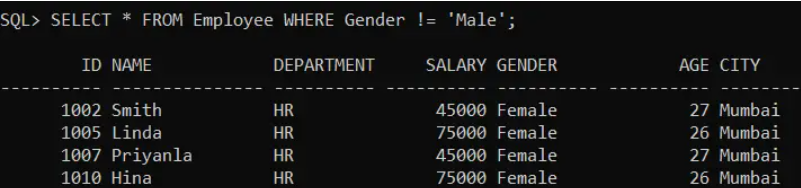
INSERT INTO Employee (Id, Name, Department, Salary, Gender, Age, City) VALUES (1009, 'Pranaya', 'IT', 50000, 'Male', 28, 'London');

INSERT INTO Employee (Id, Name, Department, Salary, Gender, Age, City) VALUES (1010, 'Hina', 'HR', 75000, 'Female', 26, 'Mumbai');

##### ****Not Equal (!=) Relational Operator in Oracle****

The Not Equal (!=) Operator in Oracle is just the opposite of the equal operator. That means this operator is used to check whether the two expressions are equal or not. If both the expressions are not equal then the condition becomes true and will return the not-matched records.

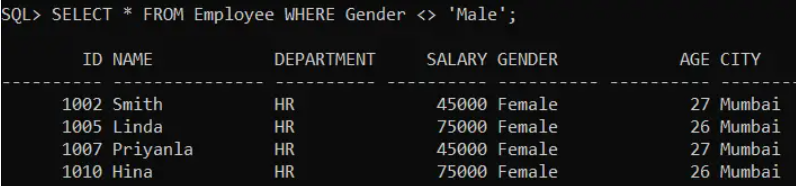
**SELECT \* FROM Employee WHERE Gender != ‘Male’;**



##### ****Not Equal (<>) Relational Operator in Oracle****

The Not Equal (<>) Operator in Oracle is the same as the Not Equal (!=) operator.

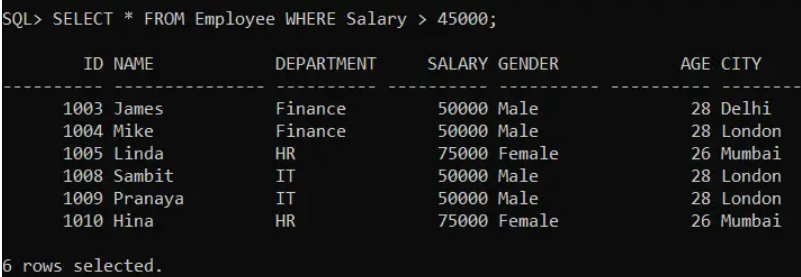
**SELECT \* FROM Employee WHERE Gender <> ‘Male’;**



##### ****Greater Than (>) Relational Operator in Oracle****

The Greater Than (>) Operator in Oracle is used to check whether the left-hand expression value is higher than the right-hand expression value. If the left-hand expression value is higher than the right-hand expression value then the condition becomes true and it will return the matched records

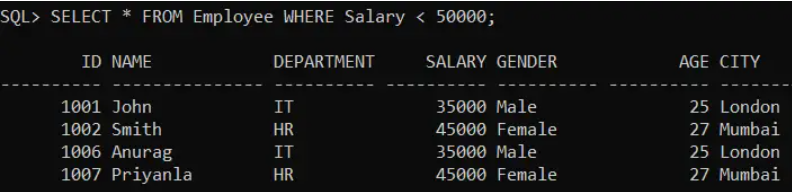
**SELECT \* FROM Employee WHERE Salary > 45000;**



##### ****Less Than (<) Relational Operator in Oracle****

The Less Than (>) Operator in Oracle is used to check whether the left-hand expression value is lower than the right-hand expression value. If the left-hand expression value is lower than the right-hand expression value then the condition becomes true and will return the matched records

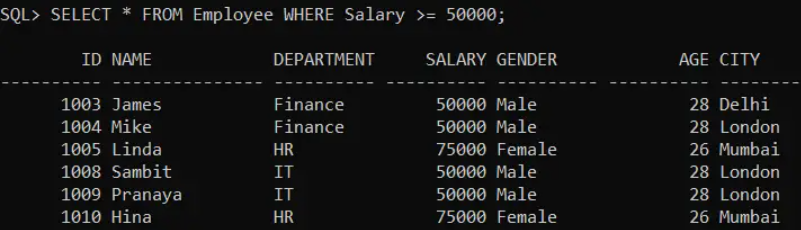
**SELECT \* FROM Employee WHERE Salary < 50000;**



##### ****Greater Than or Equal To (>=) Operator in Oracle****

The Greater than or Equal To (>=) Operator in Oracle is used to check whether the left-hand expression value is higher than or equals to the right-hand expression value or not. If the left-hand expression value is higher than or equals to the right-hand expression value then the condition becomes true and will return all the matched records.

**SELECT \* FROM Employee WHERE Salary >= 50000;**



##### ****Less Than or Equal To (<=) Operator in Oracle****

The Less than or Equal To (<=) Operator in Oracle is used to check whether the left-hand expression value is lower than or equal to the right-hand expression value or not. If the left-hand expression value is lower than or equals to the right-hand expression value then the condition becomes true and it will return all the matching records.

**SELECT \* FROM Employee WHERE Salary <= 50000;**

