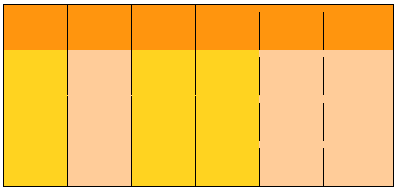
# DATA RETRIEVAL USING SELECT

# 1. SELECT statement

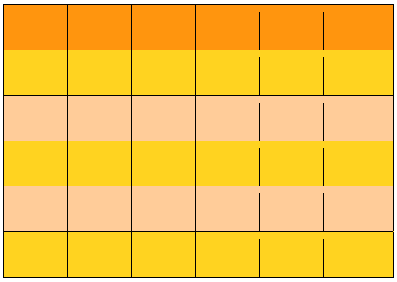
- The SELECT statement is used to retrieve the data from one or more tables
- Using SELECT statement the following operation can be performed

  - Projection : The operation displays subset of columns
  - Selection : The operation displays subset of rows

Projection means selecting set of columns from a table. In the below table, 3 columns marked with yellow colour suggesting that those columns are selected.



Selection means selecting set of rows from a table. In the below table, 3 rows marked with yellow colour suggesting that those rows are selected.



A Typical SELECT statement may consists of six clauses.

Syntax:

```
SELECT column_name [, column_list]
FROM table_name
[WHERE search_condition]
[GROUP BY grouping_column_name]
[HAVING aggregated_search_condition]
[ORDER BY sorting_column]
```

- The SELECT and FROM clauses of the statement are mandatory. The remaining four clauses are optional

Consider the following table CUSTOMER_DETAILS

| ID | Name | Age | Salary | Location | Creation_Date |
|---|---|---|---|---|---|
| 1 | Aamir | 31 | 12000 | Mumbai | 01/10/11 |
| 2 | Salman | 33 | 45000 | Kolkata | 03/03/00 |
| 3 | Sharukh | 45 | 67000 | Delhi | 08/19/11 |
| 4 | Priyanka | 21 | 48000 | Chennai | 04/30/09 |
| 5 | Deepika | 33 | 100000 | Mumbai | 08/19/11 |
| 6 | Kishore | 21 | 56000 | Banglore | 03/04/10 |
| 7 | Rahman | 32 | 15000 | | 04/30/09 |
| 8 | Saif Ali | 40 | 12000 | Kolkata | 01/10/11 |
| 9 | Kareena | 32 | 37900 | | 03/04/10 |
| 10 | Katrina | 40 | | Banglore | |

**Table 1.1 customer_details**

## 2. SELECT Statement (Projection)

To display all records from table SELECT keyword with asterisk(*) is used:

```
SQL>  select * from customer_details;

        ID NAME                          AGE     SALARY LOCATION   CREATION_ AVG_TRAN_AMT
---------- -------------------- ---------- ---------- ---------- --------- ------------
         1 Aamir                        31      38259 Mumbai     01-OCT-11
         2 Salman                       33      45000 Kolkata    03-MAR-00
         3 Sharukh                      45      67000 Delhi      19-AUG-11
         4 Priyanka                     21      35690 Chennai    30-APR-09
         5 Deepika                      33     100000 Mumbai     19-AUG-11
         6 Kishore                      21      56000 Banglore   04-MAR-10
         7 Rahman                       32      15000            30-APR-09
         8 Saif Ali                     40      12000 Kolkata    10-JAN-11
         9 Kareena                      32      37900            04-MAR-10
        10 Katrina                      40       2000 Banglore

10 rows selected.
```

The above query displays all the records from customer_details table.

- To retrieve selected columns from the table, specific column names separated by commas(,) are used

```
SQL> Select id, name, salary from customer_details;

        ID NAME                      SALARY
---------- -------------------- ----------
         1 Aamir                     38259
         2 Salman                    45000
         3 Sharukh                   67000
         4 Priyanka                  35690
         5 Deepika                  100000
         6 Kishore                   56000
         7 Rahman                    15000
         8 Saif Ali                  12000
         9 Kareena                   37900
        10 Katrina                    2000

10 rows selected.
```

The above query displays only id, name and salary columns from customer_details table.

## 3. SELECT Statement (selection)

The WHERE clause can be used to refine selection results by adding a search criteria.

- **Restricting Selection using comparison operators:**

```
SELECT name, age FROM customer_details WHERE salary>50000;
```

The above query displays name and age of customers whose salary is more than 50000, from customer_details table.

More than one condition can be used in the WHERE clause with the help of AND/OR operator.

- **Restricting Selection using IN keyword:**

```
SELECT name, location FROM customer_details WHERE location
IN ('Kolkata','Delhi');
```

The above query displays name and location of customers who are from Kolkata or Delhi.

- **Restrict based on pattern matching:**

```
 SELECT name, location FROM customer_details WHERE name
LIKE '_a%';
```

The above query displays name and location of customers whose name's second character  is a.
'_'  identifies that one character can appear before 'a' and ' % ' identifies that any numbers of character can appear after 'a'.

- **Restricting Selection using BETWEEN … AND keyword:**

```
SELECT name, location FROM customer_details WHERE age
BETWEEN 20 AND 40;
```

The above query displays name and location of customers whose age ranging from 20 to 40

- **Restricting based on NULL value:**

In Database terminology NULL is referred as value which is unspecified or unavailable.

NULL value may not display anything but it should not be misunderstood as zero or blank space.

Condition on NULL value can be given using the keyword IS.

```
SELECT name, age FROM customer_details WHERE location IS
NULL;
```

The above query displays name and age of customers whose location is not specified.

```
SELECT name, age FROM customer_details WHERE age IS NOT
NULL;
```

The above query displays name and age of customers whose age is available.

## 4. DISTINCT Keyword

To eliminate duplicate rows DISTINCT keyword is used in the SELECT statement.

```
SQL> select distinct(age) from customer_details;

       AGE
----------
         31
         21
         32
         33
         45
         40

6 rows selected.
```

The above query displays only the different age values.

## 5. ORDER BY Clause

ORDER BY clause in the SELECT statement is used to sort the results. Ascending order is the default sorting order.

```
SQL> Select name,age,salary from customer_details where salary>50000 order by salary;

NAME                    AGE     SALARY
-------------------- ---------- ----------
Kishore                  21      56000
Sharukh                  45      67000
Deepika                  33     100000
```

- Name, age and salary of the customers displayed as per the ascending order of their salary.
- DESC keyword can be used to display the list in descending order

## 6. GROUP BY Clause

The GROUP BY statement is used to group the result-set by one or more columns.

```
SQL>  select count(name), age from customer_details group by age;

COUNT(NAME)          AGE
----------- ----------
          1          31
          2          21
          2          32
          2          33
          1          45
          2          40

6 rows selected.
```

The above query displays number of customers present in particular age group.

## 7. Concatenation Operator

The concatenation( || ) operator can be used to combine character strings and values from table.

```
SQL> Select 'the name is '||name||' and age is '||age from customer_details;

'THENAMEIS'||NAME||'ANDAGEIS'||AGE
--------------------------------------------------------------------------------
the name is Aamir and age is 31
the name is Salman and age is 33
the name is Sharukh and age is 45
the name is Priyanka and age is 21
the name is Deepika and age is 33
the name is Kishore and age is 21
the name is Rahman and age is 32
the name is Saif Ali and age is 40
the name is Kareena and age is 32
the name is Katrina and age is 40

10 rows selected.
```

The Name, age and the strings are displayed as single output columns

## 8. Column Alias

Column heading in the result set can be changed with Column aliases using the keyword AS.

```
SQL> Select name as customer_name,id as customer_id from customer_details;

CUSTOMER_NAME        CUSTOMER_ID
-------------------- -----------
Aamir                          1
Salman                         2
Sharukh                        3
Priyanka                       4
Deepika                        5
Kishore                        6
Rahman                         7
Saif Ali                       8
Kareena                        9
Katrina                       10

10 rows selected.
```

- The columns name and id are displayed as customer_name and customer_id in the result set.
- Column headings in the table are unchanged.