





Data Types in SQL

I. Numeric Data Type :

INT stores integer - 2147483648 to 2147483647

SMALLINT " - 32768 to 32767

TINYINT " 0 to 255



ESTD.: 1999

TECHNOCRATS GROUP OF INSTITUTIONS
ENGINEERING | PHARMACY | MANAGEMENT | COMPUTER APPLICATIONS
ACCREDITED COLLEGE



UNBEATABLE
PLACEMENTS

NUMERIC (P,d)

$-10^{38}+1$ to $10^{38}-1$

FLOAT

Stores floating pt. no.

-1.79×10^{308}
to
 1.79×10^{308}

REAL

"

-3.40×10^{38} to
 3.40×10^{38}



ESTD.: 1999

TECHNOCRATS
INSTITUTIONS GROUP OF
NBA
ACCREDITED COLLEGE

ENGINEERING | PHARMACY | MANAGEMENT | COMPUTER APPLICATIONS

UNBEATABLE
PLACEMENTS

2. Character Data Types :-
CHAR (n) —
VARCHAR (n)



ESTD.: 1999

TECHNOCRATS GROUP OF INSTITUTIONS
ENGINEERING | PHARMACY | MANAGEMENT | COMPUTER APPLICATIONS

NBA
ACCREDITED COLLEGE

UNBEATABLE
PLACEMENTS

3. Date Time Data Types :-

DATETIME

JAN 31 1984

DATE

January 26, 1988

TIME

12:30 PM



Data Definition Language (DDL) Statements

CREATE TABLE

To create a table, you have to name that table and define its columns and datatype for each column.

Parameters used in syntax

table_name: It specifies the name of the table which you want to create.

column1, column2, ... column n: It specifies the columns which you want to add in the table. Every column must have a datatype. Every column should either be defined as "NULL" or "NOT NULL". In the case, the value is left blank; it is treated as "NULL" as default.

```
CREATE TABLE table_name
(
    column1 datatype [ NULL | NOT NULL ],
    column2 datatype [ NULL | NOT NULL ],
    ...
    column_n datatype [ NULL | NOT NULL ]
);
```



Here we are creating a table named customers.

```
CREATE TABLE customers
(
    customer_id number(10) NOT NULL,
    customer_name varchar2(50) NOT NULL,
    city varchar2(50)
);
```

This table contains three columns

- **customer_id:** It is the first column created as a number datatype (maximum 10 digits in length) and cannot contain null values.
- **customer_name:** it is the second column created as a varchar2 datatype (50 maximum characters in length) and cannot contain null values.
- **city:** This is the third column created as a varchar2 datatype. It can contain null values.





TECHNOCRATS
INSTITUTIONS

GROUP OF ENGINEERING | PHARMACY | MANAGEMENT | COMPUTER APPLICATIONS | ACCREDITED COLLEGE

NBA

Oracle ALTER TABLE Statement

UNBEATABLE
PLACEMENTS

In Oracle, ALTER TABLE statement specifies how to add, modify, drop or delete columns in a table.
It is also used to rename a table.

How to add column in a table

Syntax:

```
ALTER TABLE table_name
ADD column_name column-definition;
```

Consider that already existing table customers. Now, add a new column customer age into the table customers.

```
ALTER TABLE customers
ADD customer_age varchar2(50);
```

Now, a new column "customer_age" will be added in customers table.

How to add multiple columns in the existing table

Syntax:

```
ALTER TABLE table_name
  ADD (column_1 column-definition,
       column_2 column-definition,
       ...
       column_n column_definition);
```

Example

```
ALTER TABLE customers
  ADD (customer_type varchar2(50),
       customer_address varchar2(50));
```

Now, two columns customer_type and customer_address will be added in the table customers



Estd.: 1999

Syntax:

```
ALTER TABLE table_name  
MODIFY column_name column_type;
```

Example:

```
ALTER TABLE customers  
MODIFY customer_name varchar2(100) not null;
```

Now the column column_name in the customers table is modified to varchar2 (100) and forced the column to not allow null values

How to drop column of a table

Syntax:

```
ALTER TABLE table_name  
DROP COLUMN column_name;
```

Example:

```
ALTER TABLE customers  
DROP COLUMN customer_name;
```

This will drop the customer_name column from the table.



ESTD.: 1999

How to rename column of a table

Syntax:

```
ALTER TABLE table_name  
    RENAME COLUMN old_name to new_name;
```

Example:

```
ALTER TABLE customers  
    RENAME COLUMN customer_name to cname;
```

This will rename the column customer_name into cname.



Estd.: 1999

TECHNOCRATS
INSTITUTIONS GROUP OF
ENGINEERING | PHARMACY | MANAGEMENT | COMPUTER APPLICATIONS
NBA ACCREDITED COLLEGE

How to rename table

UNBEATABLE
PLACEMENTS

Syntax:

```
ALTER TABLE table_name  
RENAME TO new_table_name;
```

Example:

```
ALTER TABLE customers  
RENAME TO retailers;
```

This will rename the customer table into "retailers" table.



TECHNOCRATS
INSTITUTIONS

GROUP OF
NBA
ACCREDITED COLLEGE

ENGINEERING | PHARMACY | MANAGEMENT | COMPUTER APPLICATIONS

Oracle DROP TABLE Statement

**UNBEATABLE
PLACEMENTS**

Oracle DROP TABLE statement is used to remove or delete a table from the Oracle database

DROP TABLE Example

```
DROP TABLE customers;
```

This will drop the table named customers.

Oracle TRUNCATE TABLE

In Oracle, TRUNCATE TABLE statement is used to remove all records from a table

Consider a table named "customers" and execute the following query to truncate this

```
TRUNCATE TABLE customers;
```

RENAME THE TABLE

RENAME STUDENTS TO ARTISTS;

After that the **table "students"** will be changed into **table name "artists"**



Data Manipulation Language

DML commands are used to modify the database. It is responsible for all form of changes in the database.

Syntax

```
SELECT expressions
FROM tables
WHERE conditions;
```

Parameters

- 1) **expressions:** It specifies the columns or calculations that you want to retrieve.
- 2) **tables:** This parameter specifies the tables that you want to retrieve records from. There must be at least one table within the FROM clause.
- 3) **conditions:** It specifies the conditions that must be followed for selection.



Select Example: select all fields

**UNBEATABLE
PLACEMENTS**

Let's take an example to select all fields from an already created table named customers

```
SELECT *
```

```
FROM customers;
```

output

NAME	AGE	ADDRESS	SALARY
mohan	21	ghaziabad	20000
rohan	22	delhi	22000
sohan	25	noida	24000
alex	28	Paris	40000

4 rows returned in 0.02



Select Example: select specific fields

**UNBEATABLE
PLACEMENTS**

output

NAME	AGE	ADDRESS	SALARY
mohan	21	ghaziabad	20000
rohan	22	delhi	22000
sohan	25	noida	24000
alex	28	Paris	40000

Example

```
SELECT age, address, salary
FROM customers
WHERE age < 25
AND salary > '20000'
```

output

AGE	ADDRESS	SALARY
22	delhi	22000



Oracle Insert Statement

In Oracle, INSERT statement is used to add a single record or multiple records into the table.

INSERT INTO table

(column1, column2, ... column_n)

VALUES

(expression1, expression2, ... expression_n);

Oracle insert query is used to insert records into table. For example:

```
insert into customers values(101,'rahul','delhi');
```



ESTD.: 1999

TECHNOCRATS
INSTITUTIONS GROUP OF **NBA**
ENGINEERING | PHARMACY | MANAGEMENT | COMPUTER APPLICATIONS ACCREDITED COLLEGE

**UNBEATABLE
PLACEMENTS**

INSERT MULTIPLE ROWS

```
INSERT INTO student (ID, NAME)  
VALUES (1, 'ARMAAN'), (2, 'BILLY'), (3, 'CHARLIE');
```



Oracle UPDATE Statement

In Oracle, UPDATE statement is used to update the existing records in a table.

Parameters:

1) **column1, column2, ... column_n:**

It specifies the columns that you want to update.

2) **expression1, expression2, ... expression_n:**

This specifies the values to assign to the column1, column2, ?. column_n.

3) **conditions:** It specifies the conditions that must be fulfilled for execution of UPDATE stateme.

Syntax:

```
UPDATE table
SET column1 = expression1,
    column2 = expression2,
    ...
    column_n = expression_n
WHERE conditions;
```

```
update customers set name='bob', city='london' where id=101;
```

Oracle Delete Query

The **SQL DELETE statement** is used to delete rows from a table. Generally DELETE statement removes one or more records from a table.

Let's see the Syntax for the SQL DELETE statement:

```
DELETE FROM table_name [WHERE condition];
```

Oracle update query is used to delete records of a table from database. For example:

```
delete from customers where id=101;
```