**Oracle Database 10g: Manual**

ORACLE

It is a very large multi-user based database management system. Oracle is a relational database management system developed by 'Oracle Corporation'.

Oracle works to efficiently manage its resources, a database of information among the multiple clients requesting and sending data in the network.

It is an excellent database server choice for client/server computing. Oracle supports all major operating systems for both clients and servers, including MSDOS, Windows, UnixWare, OS/2 and most UNIX flavors.

**SQL ─ Overview**

**SQL** is a language to operate databases; it includes database creation, deletion, fetching rows, modifying rows, etc. SQL is an ANSI (American National Standards Institute) standard language.

What is SQL? .

**SQL** is *Structured Query Language*, which is a computer language for storing, manipulating and retrieving data stored in a relational database.

**SQL** is the standard language for Relational Database System. All the Relational Database Management Systems (RDMS) like MySQL, MS Access, Oracle, Sybase, Informix, Postgres and SQL Server use SQL as their standard database language.

Why SQL?

SQL is widely popular because it offers the following advantages:

* Allows users to access data in the relational database management systems.
* Allows users to describe the data.
* Allows users to define the data in a database and manipulate that data.
* Allows to embed within other languages using SQL modules, libraries & pre-compilers.
* Allows users to create and drop databases and tables.
* Allows users to create view, stored procedure, functions in a database.
* Allows users to set permissions on tables, procedures and views.

SQL Commands

The standard SQL commands to interact with relational databases are CREATE, SELECT, INSERT, UPDATE, DELETE and DROP. These commands can be classified into the following groups based on their nature:

|  |  |
| --- | --- |
| **Command** | **Description** |
| SELECT  INSERT UPDATE DELETE  MERGE | Retrieves data from the database, enters new rows, changes existing rows, and removes unwanted rows from tables in the database, respectively. Collectively known as **data manipulation language (DML)**. |
| CREATE  ALTER  DROP  RENAME TRUNCATE COMMENT | Sets up, changes, and removes data structures from tables. Collectively known as **data definition language (DDL**). |
| GRANT REVOKE | Gives or removes access rights to both the Oracle database and the structures within it. |
| COMMIT ROLLBACK SAVEPOINT | Manages the changes made by DML statements. Changes to the data can be grouped together into logical transactions. |

**SQL ─ RDBMS Concepts**

What is RDBMS?

**RDBMS** stands for **R**elational **D**atabase **M**anagement **S**ystem. RDBMS is the basis for SQL, and for all modern database systems like MS SQL Server, IBM DB2, Oracle, MySQL, and Microsoft Access.

A Relational database management system (RDBMS) is a database management system (DBMS) that is based on the relational model as introduced by E. F. Codd.

What is a table?

The data in an RDBMS is stored in database objects which are called as tables. This table is basically a collection of related data entries and it consists of numerous columns and rows.

Remember, a table is the most common and simplest form of data storage in a relational database. The following program is an example of an **EMPLOYEES** table:



What is a field?

Every table is broken up into smaller entities called fields. The fields in the employees table consist of EMPID, FIRSTNAME, LASTNAME, DEPTID, and SALARY.

A field is a column in a table that is designed to maintain specific information about every record in the table.

What is a Record or a Row?

A record is also called as a row of data is each individual entry that exists in a table. For example, there are 5 records in the above **EMPLOYEES** table. Following is a single row of data or record in the **EMPLOYEES** table:

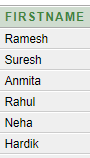


A record is a horizontal entity in a table.

**What is a column?**

A column is a vertical entity in a table that contains all information associated with a specific field in a table.

**For example**, a column in the **EMPLOYEES** table is **FISRTNAME**, which represents location description and would be as shown below:



**SQL** **─ Data Types**

|  |  |
| --- | --- |
| **Data Type** | **Description** |
| VARCHAR2(size) | Variable-length character data (A maximum size must be specified: minimum size is 1; maximum size is 4,000.) |
| CHAR [(size)] | Fixed-length character data of length size bytes (Default and minimum size is 1; maximum size is 2,000.) |
| NUMBER [(p,s)] | Number having precision p and scale s (The precision is the total number of decimal digits, and the scale is the number of digits to the right of the decimal point. |
| DATE | Date and time values to the nearest second between January 1, 4712 B.C., and December 31, 9999 A.D. |
| LONG | Variable-length character data (up to 2 GB) |
| CLOB | Character data (up to 4 GB |

SQL Data Type is an attribute that specifies the type of data of any object. Each column, variable and expression has a related data type in SQL. You can use these data types while creating your tables. You can choose a data type for a table column based on your requirement.

**CREATE TABLE Statement**

Creating a basic table involves naming the table and defining its columns and each column's data type.

The SQL **CREATE TABLE** statement is used to create a new table.

**Syntax:**

The basic syntax of the **CREATE TABLE** statement is as follows:

**CREATE TABLE** table\_name (

column1 datatype,

column2 datatype,

column3 datatype,

.....

columnN datatype,

PRIMARY KEY (one or more columns)

);

* Creating the **EMPLOYEE TABLE**

Enter the following SQL command to create the **EMPLOYEES** table.

create table Employees (

EmpId number,

FirstName varchar(20),

LastName Varchar(20),

EmailId Varchar(50),

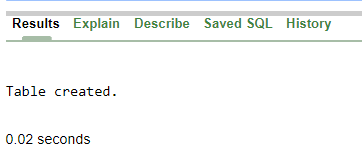
Gender char(1),

mobileNo char(10),

DateOfJoining date default sysdate,



DeptId int);



|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

* Describe the Table **EMPLOYEES**



desc Employees;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','4356558137865441','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','4356558137865441','Close')) | **EMPLOYEES** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |  |  | | |
| [EMPLOYEES](javascript:ret_Column('SYSTEM.EMPLOYEES');) | [EMPID](javascript:ret_Column('EMPID');) | Number | - | - | - | - | nullable | - | - |  |  | | |
|  | [FIRSTNAME](javascript:ret_Column('FIRSTNAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |  |  | | |
|  | [LASTNAME](javascript:ret_Column('LASTNAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |  |  | | |
|  | [EMAILID](javascript:ret_Column('EMAILID');) | Varchar2 | 50 | - | - | - | nullable | - | - |  |  | | |
|  | [GENDER](javascript:ret_Column('GENDER');) | Char | 1 | - | - | - | nullable | - | - |  |  | | |
|  | [MOBILENO](javascript:ret_Column('MOBILENO');) | Char | 10 | - | - | - | nullable | - | - |  |  | | |
|  | [DATEOFJOINING](javascript:ret_Column('DATEOFJOINING');) | Date | 7 | - | - | - | nullable | sysdate | - |  |  | | |
|  | [DEPTID](javascript:ret_Column('DEPTID');) | Number | - | - | 0 | - | nullable | - | - |  |  | | |
| |  |  |  | | --- | --- | --- | |  |  | 1 - 8 | | | | | | | | | | | | |  |  | |

**ALTER TABLE**

* **Altering the table structure**

After you create a table, you may need to change the table structure for any of the following reasons:

• You omitted a column.

• Your column definition needs to be changed.

• You need to remove columns.

All changes in the table structure are made by using the **ALTER TABLE** command,

Followed by a keyword that produces the specific change you want to make.

Four options are available:

|  |
| --- |
| Adding Column |
| Modifying Column |
| Dropping Column |
| Renaming Column |

* Adding Column

The basic syntax to **ADD** columns is:

ALTER TABLE tablename

{ADD }( columnname datatype [ {ADD } columnname datatype]);



Alter table Employees add DOB date;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table altered.

0.05 seconds

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |



Desc Employees;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','4356558137865441','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','4356558137865441','Close')) | **EMPLOYEES** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [EMPLOYEES](javascript:ret_Column('SYSTEM.EMPLOYEES');) | [EMPID](javascript:ret_Column('EMPID');) | Number | - | - | - | - | nullable | - | - |
|  | [FIRSTNAME](javascript:ret_Column('FIRSTNAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |
|  | [LASTNAME](javascript:ret_Column('LASTNAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |
|  | [EMAILID](javascript:ret_Column('EMAILID');) | Varchar2 | 50 | - | - | - | nullable | - | - |
|  | [GENDER](javascript:ret_Column('GENDER');) | Char | 1 | - | - | - | nullable | - | - |
|  | [MOBILENO](javascript:ret_Column('MOBILENO');) | Char | 10 | - | - | - | nullable | - | - |
|  | [DATEOFJOINING](javascript:ret_Column('DATEOFJOINING');) | Date | 7 | - | - | - | nullable | sysdate | - |
|  | [DEPTID](javascript:ret_Column('DEPTID');) | Number | - | - | 0 | - | nullable | - | - |
|  | [**DOB**](javascript:ret_Column('DOB');) | **Date** | **7** | **-** | **-** | **-** | **nullable** | **-** | - |
| |  |  |  |  | | --- | --- | --- | --- | |  |  | 1 - 9 |  | | | | | | | | | | |

DOB added

Alter table Employees add (Salary number (10, 2), pin number);



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) |  |

Table altered.

0.00 seconds



Desc Employees;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','4356558137865441','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','4356558137865441','Close')) | **EMPLOYEES** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [EMPLOYEES](javascript:ret_Column('SYSTEM.EMPLOYEES');) | [EMPID](javascript:ret_Column('EMPID');) | Number | - | - | - | - | nullable | - | - |
|  | [FIRSTNAME](javascript:ret_Column('FIRSTNAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |
|  | [LASTNAME](javascript:ret_Column('LASTNAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |
|  | [EMAILID](javascript:ret_Column('EMAILID');) | Varchar2 | 50 | - | - | - | nullable | - | - |
|  | [GENDER](javascript:ret_Column('GENDER');) | Char | 1 | - | - | - | nullable | - | - |
|  | [MOBILENO](javascript:ret_Column('MOBILENO');) | Char | 10 | - | - | - | nullable | - | - |
|  | [DATEOFJOINING](javascript:ret_Column('DATEOFJOINING');) | Date | 7 | - | - | - | nullable | sysdate | - |
|  | [DEPTID](javascript:ret_Column('DEPTID');) | Number | - | - | 0 | - | nullable | - | - |
|  | [DOB](javascript:ret_Column('DOB');) | Date | 7 | - | - | - | nullable | - | - |
|  | [**SALARY**](javascript:ret_Column('SALARY');) | **Number** | **-** | **10** | **2** | **-** | **nullable** | **-** | **-** |
|  | [**PIN**](javascript:ret_Column('PIN');) | **Number** | **-** | **-** | **-** | **-** | **nullable** | **-** | **-** |
| |  |  |  | | --- | --- | --- | |  |  | 1 - 11 | | | | | | | | | | |

Salary & Pin added

* Modifying Column

The basic syntax to **Modify** columns is:

ALTER TABLE tablename

{MODIFY }( columnname datatype [ {MODIFY } columnname datatype]);



Alter table Employees modify firstname varchar(30);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table altered.

Desc Employees;



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:4356558137865441::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','4356558137865441','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','4356558137865441','Close')) | **EMPLOYEES** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [EMPLOYEES](javascript:ret_Column('SYSTEM.EMPLOYEES');) | [EMPID](javascript:ret_Column('EMPID');) | Number | - | - | - | - | nullable | - | - |
|  | [FIRSTNAME](javascript:ret_Column('FIRSTNAME');) | Varchar2 | **30** | - | - | - | nullable | - | - |
|  | [LASTNAME](javascript:ret_Column('LASTNAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |
|  | [EMAILID](javascript:ret_Column('EMAILID');) | Varchar2 | 50 | - | - | - | nullable | - | - |
|  | [GENDER](javascript:ret_Column('GENDER');) | Char | 1 | - | - | - | nullable | - | - |
|  | [MOBILENO](javascript:ret_Column('MOBILENO');) | Char | 10 | - | - | - | nullable | - | - |
|  | [DATEOFJOINING](javascript:ret_Column('DATEOFJOINING');) | Date | 7 | - | - | - | nullable | sysdate | - |
|  | [DEPTID](javascript:ret_Column('DEPTID');) | Number | - | - | 0 | - | nullable | - | - |
|  | [DOB](javascript:ret_Column('DOB');) | Date | 7 | - | - | - | nullable | - | - |
|  | [SALARY](javascript:ret_Column('SALARY');) | Number | - | 10 | 2 | - | nullable | - | - |
|  | [PIN](javascript:ret_Column('PIN');) | Number | - | - | - | - | nullable | - | - |
| |  |  |  | | --- | --- | --- | |  |  | 1 - 11 | | | | | | | | | | |

Size 30 modified

* Deleting Column

The basic syntax to **Drop** columns is:

ALTER TABLE tablename

{DROP }( columnname datatype [ {DROP } columnname datatype]);



Alter table Employees drop column pin;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table dropped.

0.48 seconds

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Desc Employees;



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','3828942288315066','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','3828942288315066','Close')) | **EMPLOYEES** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [EMPLOYEES](javascript:ret_Column('SYSTEM.EMPLOYEES');) | [EMPID](javascript:ret_Column('EMPID');) | Number | - | - | - | - | nullable | - | - |
|  | [FIRSTNAME](javascript:ret_Column('FIRSTNAME');) | Varchar2 | 30 | - | - | - | nullable | - | - |
|  | [LASTNAME](javascript:ret_Column('LASTNAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |
|  | [EMAILID](javascript:ret_Column('EMAILID');) | Varchar2 | 50 | - | - | - | nullable | - | - |
|  | [GENDER](javascript:ret_Column('GENDER');) | Char | 1 | - | - | - | nullable | - | - |
|  | [MOBILENO](javascript:ret_Column('MOBILENO');) | Char | 10 | - | - | - | nullable | - | - |
|  | [DATEOFJOINING](javascript:ret_Column('DATEOFJOINING');) | Date | 7 | - | - | - | nullable | sysdate | - |
|  | [DEPTID](javascript:ret_Column('DEPTID');) | Number | - | - | 0 | - | nullable | - | - |
|  | [DOB](javascript:ret_Column('DOB');) | Date | 7 | - | - | - | nullable | - | - |
|  | [SALARY](javascript:ret_Column('SALARY');) | Number | - | 10 | 2 | - | nullable | - | - |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | |  |  | 1 - 10 |  |  | | | | | | | | | | |

Pin deleted

* Renaming Column

The basic syntax to **Rename** columns is:

ALTER TABLE tablename

{RENAME}( columnname to columnname);



Alter table Employees rename column Dateofjoining to DOJ;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table altered.

0.02 seconds

Desc Employees;



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:3828942288315066::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','3828942288315066','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','3828942288315066','Close')) | **EMPLOYEES** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [EMPLOYEES](javascript:ret_Column('SYSTEM.EMPLOYEES');) | [EMPID](javascript:ret_Column('EMPID');) | Number | - | - | - | - | nullable | - | - |
|  | [FIRSTNAME](javascript:ret_Column('FIRSTNAME');) | Varchar2 | 30 | - | - | - | nullable | - | - |
|  | [LASTNAME](javascript:ret_Column('LASTNAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |
|  | [EMAILID](javascript:ret_Column('EMAILID');) | Varchar2 | 50 | - | - | - | nullable | - | - |
|  | [GENDER](javascript:ret_Column('GENDER');) | Char | 1 | - | - | - | nullable | - | - |
|  | [MOBILENO](javascript:ret_Column('MOBILENO');) | Char | 10 | - | - | - | nullable | - | - |
|  | [DOJ](javascript:ret_Column('DOJ');) | Date | 7 | - | - | - | nullable | sysdate | - |
|  | [DEPTID](javascript:ret_Column('DEPTID');) | Number | - | - | 0 | - | nullable | - | - |
|  | [DOB](javascript:ret_Column('DOB');) | Date | 7 | - | - | - | nullable | - | - |
|  | [SALARY](javascript:ret_Column('SALARY');) | Number | - | 10 | 2 | - | nullable | - | - |
| |  |  |  |  | | --- | --- | --- | --- | |  |  | 1 - 10 |  | | | | | | | | | | |

Rename Dateofjoining to DOJ

**DROP TABLE**

The **DROP TABLE** command permanently deletes a table (and thus its data) from the database.

**Syntax:**

DROP TABLE tablename ;



drop table Employees;

**TRUNCAT TABLE**

Removes all rows from a table, leaving the table empty and the table structure intact.

**Syntax:**

TRUNCATE TABLE table name;

;



truncate table Employees;

**RENAME TABLE**

To rename a table , you use the following Oracle RENAME table statement as follows:

**Syntax:**

RENAME Table\_name TO new\_name ;

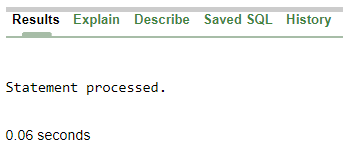
;

In the rename table statement:

1. First specify the name of the existing table which you want to rename.
2. Second specify the new table name.



Rename Employees to EmployeesOne;



Rename EmployeesOne to Employees;

**Constraints**

The Oracle server uses constraints to prevent invalid data entry into tables.

You can use constraints to do the following:

* + Is a rule to which data must conform
  + Constraints name are optional
  + Can be added a Column level or Table level
  + Can be enabled/disabled/dropped

|  |  |
| --- | --- |
| Constraint | Description |
| NOT NULL | Specifies that the column cannot contain a null value |
| UNIQUE | Specifies a column or combination of columns whose values must be unique for all rows in the table |
| PRIMARY KEY | Uniquely identifies each row of the table |
| CHECK | Specifies a condition that must be true |
| FOREIGN KEY | Establishes and enforces a foreign key relationship between the column and a column of the referenced table |

**For Existing Table** The **ALTER TABLE** command can also be used to add /drop table constraints.

the syntax would be:

ALTER TABLE tablename

ADD /DROP constraint [ ADD / DROP constraint ]

**NOT NULL Constraint**

**Example :**

Create table Sample1(

Col1 number constraint Sample1\_Col1\_nn not null,



Col2 number not null);

(**Col1:** if we will not give name oracle server automatically assign the name, but it will better if you will pass a name by yourself, because you can enabled/disabled/dropped constraint easily.)

(**Col2:** Directly write constraint name)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table created.

0.14 seconds



desc Sample1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','3310407148764960','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','3310407148764960','Close')) | **SAMPLE1** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [SAMPLE1](javascript:ret_Column('SYSTEM.SAMPLE1');) | [COL1](javascript:ret_Column('COL1');) | Number | - | - | - | - | - | - | - |
|  | [COL2](javascript:ret_Column('COL2');) | Number | - | - | - | - | - | - | - |
| |  |  |  |  | | --- | --- | --- | --- | |  |  | 1 - 2 |  | | | | | | | | | | |

Both column are NOT NULL

If you want to pass the existing column or existing table which you have already created table name ***Employees*.**

Here Firstname column we want to pass not null constraint by **ALTER** command.

alter table Employees modify firstname varchar2(30)



Constraint Employees\_firstname\_nn not null;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) |  | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table altered.

0.01 seconds

desc Employees;



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','3310407148764960','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','3310407148764960','Close')) | **EMPLOYEES** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [EMPLOYEES](javascript:ret_Column('SYSTEM.EMPLOYEES');) | [EMPID](javascript:ret_Column('EMPID');) | Number | - | - | - | - | nullable | - | - |
|  | [FIRSTNAME](javascript:ret_Column('FIRSTNAME');) | Varchar2 | 30 | - | - | - | - | - | - |
|  | [LASTNAME](javascript:ret_Column('LASTNAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |
|  | [EMAILID](javascript:ret_Column('EMAILID');) | Varchar2 | 50 | - | - | - | nullable | - | - |
|  | [GENDER](javascript:ret_Column('GENDER');) | Char | 1 | - | - | - | nullable | - | - |
|  | [MOBILENO](javascript:ret_Column('MOBILENO');) | Char | 10 | - | - | - | nullable | - | - |
|  | [DOJ](javascript:ret_Column('DOJ');) | Date | 7 | - | - | - | nullable | sysdate | - |
|  | [DEPTID](javascript:ret_Column('DEPTID');) | Number | - | - | 0 | - | nullable | - | - |
|  | [DOB](javascript:ret_Column('DOB');) | Date | 7 | - | - | - | nullable | - | - |
|  | [SALARY](javascript:ret_Column('SALARY');) | Number | - | 10 | 2 | - | nullable | - | - |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | |  |  | 1 - 10 |  |  | | | | | | | | | | |

Firstname is NOT NULL

**UNIQUE Constraint**

The **UNIQUE** specification creates a unique index in the respective attribute. Use

it to avoid duplicated values in a column.

**Example :**

Create table Sample1(

Col1 number constraint Sample1\_Col1\_un unique,

Col2 number ,

Col3 number unique,

Constraint Sample1\_unique unique(col2) );



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table created.

0.14 seconds

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','3310407148764960','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','3310407148764960','Close')) | **SAMPLE1** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [SAMPLE1](javascript:ret_Column('SYSTEM.SAMPLE1');) | [COL1](javascript:ret_Column('COL1');) | Number | - | - | - | - | nullable | - | - |
|  | [COL2](javascript:ret_Column('COL2');) | Number | - | - | - | - | nullable | - | - |
|  | [COL3](javascript:ret_Column('COL3');) | Number | - | - | - | - | nullable | - | - |
| |  |  |  |  | | --- | --- | --- | --- | |  |  | 1 - 3 |  | | | | | | | | | | |

In our **Employees table** { EmailId } pass unique constraint by ALTER command.

alter table Employees add constrant Employees\_Email\_un unique(EmailId);



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table altered.

0.05 seconds



desc Employees;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:3310407148764960::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','3310407148764960','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','3310407148764960','Close')) | **EMPLOYEES** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [EMPLOYEES](javascript:ret_Column('SYSTEM.EMPLOYEES');) | [EMPID](javascript:ret_Column('EMPID');) | Number | - | - | - | - | nullable | - | - |
|  | [FIRSTNAME](javascript:ret_Column('FIRSTNAME');) | Varchar2 | 30 | - | - | - | - | - | - |
|  | [LASTNAME](javascript:ret_Column('LASTNAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |
|  | [EMAILID](javascript:ret_Column('EMAILID');) | Varchar2 | 50 | - | - | - | nullable | - | - |
|  | [GENDER](javascript:ret_Column('GENDER');) | Char | 1 | - | - | - | nullable | - | - |
|  | [MOBILENO](javascript:ret_Column('MOBILENO');) | Char | 10 | - | - | - | nullable | - | - |
|  | [DOJ](javascript:ret_Column('DOJ');) | Date | 7 | - | - | - | nullable | sysdate | - |
|  | [DEPTID](javascript:ret_Column('DEPTID');) | Number | - | - | 0 | - | nullable | - | - |
|  | [DOB](javascript:ret_Column('DOB');) | Date | 7 | - | - | - | nullable | - | - |
|  | [SALARY](javascript:ret_Column('SALARY');) | Number | - | 10 | 2 | - | nullable | - | - |
| |  |  |  | | --- | --- | --- | |  |  | 1 - 10 | | | | | | | | | | |

**Primary Key Constraint**

A primary key is a field in a table which uniquely identifies each row/record in a database table. Primary keys must contain unique values. A primary key column cannot have NULL values.

Here is the syntax to define the ID attribute as a primary key in a CUSTOMERS table.

CREATE TABLE CUSTOMERS(

ID INT PRIMARY KEY,

NAME VARCHAR (20),

AGE DATEI ,



ADDRESS CHAR (25));

Or we can write Multiple column.

PRIMARY KEY (ID ,NAME));

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table created.

0.02 seconds

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

desc Customers;



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','1005107903632568','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','1005107903632568','Close')) | **CUSTOMERS** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [CUSTOMERS](javascript:ret_Column('SYSTEM.CUSTOMERS');) | [ID](javascript:ret_Column('ID');) | Number | - | - | - | 1 | - | - | - |
|  | [NAME](javascript:ret_Column('NAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |
|  | [AGE](javascript:ret_Column('AGE');) | Number | - | - | - | - | nullable | - | - |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | |  |  | 1 - 3 |  |  | | | | | | | | | | |

ID is PRIMARY KEY

Delete Primary Key You can clear the primary key constraints from the table with the syntax given below.



ALTER TABLE CUSTOMERS DROP PRIMARY KEY ;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table altered.

0.00 seconds



desc Customers;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','1005107903632568','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','1005107903632568','Close')) | **CUSTOMERS** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [CUSTOMERS](javascript:ret_Column('SYSTEM.CUSTOMERS');) | [ID](javascript:ret_Column('ID');) | Number | - | - | - | - | nullable | - | - |
|  | [NAME](javascript:ret_Column('NAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |
|  | [AGE](javascript:ret_Column('AGE');) | Number | - | - | - | - | nullable | - | - |
| |  |  |  |  | | --- | --- | --- | --- | |  |  | 1 - 3 |  |   Delete Primary Key | | | | | | | | | |

To create a PRIMARY KEY constraint on the multiple column when the CUSTOMERS table already exists, use the following SQL syntax:

ALTER TABLE CUSTOMERS ADD CONSTRAINT CUSTOMERS\_NAME\_PK PRIMARY KEY (ID, NAME);



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table altered.

desc Customers;



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','1005107903632568','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','1005107903632568','Close')) | **CUSTOMERS** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [CUSTOMERS](javascript:ret_Column('SYSTEM.CUSTOMERS');) | [ID](javascript:ret_Column('ID');) | Number | - | - | - | 1 | - | - | - |
|  | [NAME](javascript:ret_Column('NAME');) | Varchar2 | 20 | - | - | 2 | - | - | - |
|  | [AGE](javascript:ret_Column('AGE');) | Number | - | - | - | - | nullable | - | - |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | |  |  | 1 - 3 |  |  | | | | | | | | | | |

Primary Key ID & NAME

**CHECK Constraint**

The CHECK Constraint enables a condition to check the value being entered into a record. If the condition evaluates to false, the record violates the constraint and isn't entered the table.

**Example** the following program creates a new table called CUSTOMERS and adds three columns. Here, we add a CHECK with AGE column, so that you cannot have any CUSTOMER who is below 18 years.

CREATE TABLE CUSTOMERS(

ID INT ,

NAME VARCHAR (20),



AGE INT CHECK (AGE >= 18));

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','1005107903632568','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','1005107903632568','Close')) | **CUSTOMERS** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [CUSTOMERS](javascript:ret_Column('SYSTEM.CUSTOMERS');) | [ID](javascript:ret_Column('ID');) | Number | - | - | - | - | nullable | - | - |
|  | [NAME](javascript:ret_Column('NAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |
|  | [AGE](javascript:ret_Column('AGE');) | Number | - | - | 0 | - | nullable | - | - |
| |  |  |  | | --- | --- | --- | |  |  | 1 - 3 | | | | | | | | | | |

If the CUSTOMERS table has already been created, then to add a CHECK constraint to AGE column, you would write a statement like the one given below.

ALTER TABLE CUSTOMERS

MODIFY AGE INT CHECK (AGE >= 18 );

OR

ALTER TABLE CUSTOMERS

MODIFY AGE INT CONSTRAINT CUSTOMERS\_AGE\_CK CHECK (AGE >= 18);



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table altered.

0.01 seconds

You can also use the following syntax, which supports naming the constraint in multiple columns as well:

ALTER TABLE CUSTOMERS



ADD CONSTRAINT CUSTOMERS\_AGE\_CK CHECK (AGE >= 18);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table altered.

0.01 second

DROP a CHECK Constraint To drop a CHECK constraint, use the following SQL syntax.

ALTER TABLE CUSTOMERS



DROP CONSTRAINT CUSTOMERS\_AGE\_CK;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table dropped.

0.00 seconds

**Our existing TABLE Employees**

Check Gender “**M**” or “**F**”

ALTER TABLE EMPLOYEES

ADD CONSTRAINT EMP\_GENDER\_CHECK CHECK(GENDER IN(‘M’,’F’));



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:2876977977089158::NO:::) |  | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:2876977977089158::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:2876977977089158::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:2876977977089158::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:2876977977089158::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table altered.

0.02 seconds

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

**Foreign Key Constraint**

A foreign key is a key used to link two tables together. This is sometimes also called as a referencing key.

A Foreign Key is a column or a combination of columns whose values match a Primary Key in a different table.

The relationship between 2 tables matches the Primary Key in one of the tables with a Foreign Key in the second table.

If a table has a primary key defined on any field(s), then you cannot have two records having the same value of that field(s).

Example Consider the structure of the following two tables.

CUSTOMERS Table:

CREATE TABLE CUSTOMERS(

CUSTID INT PRIMARY KEY,



NAME VARCHAR (20));

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','1005107903632568','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','1005107903632568','Close')) | **CUSTOMERS** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [CUSTOMERS](javascript:ret_Column('SYSTEM.CUSTOMERS');) | [CUSTID](javascript:ret_Column('CUSTID');) | Number | - | - | - | 1 | - | - | - |
|  | [NAME](javascript:ret_Column('NAME');) | Varchar2 | 20 | - | - | - | nullable | - | - |
| |  |  |  |  | | --- | --- | --- | --- | |  |  | 1 - 2 |  | | | | | | | | | | |

ORDERS Table

CREATE TABLE ORDERS (

ID INT PRIMARY KEY,

CUSTOMER\_ID INT references CUSTOMERS (CUSTID),



AMOUNT NUMBER);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:1005107903632568::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','1005107903632568','Close')) | **TABLE** | [Object](javascript:popupFieldHelp('22174606305655533','1005107903632568','Close')) | **ORDERS** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [ORDERS](javascript:ret_Column('SYSTEM.ORDERS');) | [ID](javascript:ret_Column('ID');) | Number | - | - | - | 1 | - | - | - |
|  | [CUSTOMERID](javascript:ret_Column('CUSTOMERID');) | Number | - | - | 0 | - | nullable | - | - |
|  | [AMOUNT](javascript:ret_Column('AMOUNT');) | Number | - | - | - | - | nullable | - | - |
| |  |  |  |  | | --- | --- | --- | --- | |  |  | 1 - 3 |  | | | | | | | | | | |

If the ORDERS table has already been created and the foreign key has not yet been set, the use the syntax for specifying a foreign key by altering a table.

ALTER TABLE ORDERS

ADD CONSTRANT ORDER\_FK FOREIGN KEY (CustomerID)



REFERENCES CUST**OMER**S (CUSTID);

REFERENCES identifies the table and column in the parent table.

• ON DELETE CASCADE indicates that when the row in the parent table is deleted, the dependent rows in the child table are also deleted.

• ON DELETE SET NULL converts foreign key values to null when the parent value is removed.

DROP a FOREIGN KEY Constraint

To drop a FOREIGN KEY constraint, use the following SQL syntax.



ALTER TABLE ORDERS DROP FOREIGN KEY;

**Managing the constraints**

Viewing Constraints

Adding Constraints

Disabling Constraints

Constraints

Enabling Constraints

Constraints

Dropping Constraints



desc user\_constraints;

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Object Type](javascript:popupFieldHelp('22156222310565561','928044562611615','Close')) | **VIEW** | [Object](javascript:popupFieldHelp('22174606305655533','928044562611615','Close')) | **USER\_CONSTRAINTS** |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [USER\_CONSTRAINTS](javascript:ret_Column('SYS.USER_CONSTRAINTS');) | [OWNER](javascript:ret_Column('OWNER');) | Varchar2 | 30 | - | - | - | - | - | Owner of the table |
|  | [CONSTRAINT\_NAME](javascript:ret_Column('CONSTRAINT_NAME');) | Varchar2 | 30 | - | - | - | - | - | Name associated with constraint definition |
|  | [CONSTRAINT\_TYPE](javascript:ret_Column('CONSTRAINT_TYPE');) | Varchar2 | 1 | - | - | - | nullable | - | Type of constraint definition |
|  | [TABLE\_NAME](javascript:ret_Column('TABLE_NAME');) | Varchar2 | 30 | - | - | - | - | - | Name associated with table with constraint definition |
|  | [SEARCH\_CONDITION](javascript:ret_Column('SEARCH_CONDITION');) | Long | 0 | - | - | - | nullable | - | Text of search condition for table check |
|  | [R\_OWNER](javascript:ret_Column('R_OWNER');) | Varchar2 | 30 | - | - | - | nullable | - | Owner of table used in referential constraint |
|  | [R\_CONSTRAINT\_NAME](javascript:ret_Column('R_CONSTRAINT_NAME');) | Varchar2 | 30 | - | - | - | nullable | - | Name of unique constraint definition for referenced table |
|  | [DELETE\_RULE](javascript:ret_Column('DELETE_RULE');) | Varchar2 | 9 | - | - | - | nullable | - | The delete rule for a referential constraint |
|  | [STATUS](javascript:ret_Column('STATUS');) | Varchar2 | 8 | - | - | - | nullable | - | Enforcement status of constraint - ENABLED or DISABLED |
|  | [DEFERRABLE](javascript:ret_Column('DEFERRABLE');) | Varchar2 | 14 | - | - | - | nullable | - | Is the constraint deferrable - DEFERRABLE or NOT DEFERRABLE |
|  | [DEFERRED](javascript:ret_Column('DEFERRED');) | Varchar2 | 9 | - | - | - | nullable | - | Is the constraint deferred by default - DEFERRED or IMMEDIATE |
|  | [VALIDATED](javascript:ret_Column('VALIDATED');) | Varchar2 | 13 | - | - | - | nullable | - | Was this constraint system validated? - VALIDATED or NOT VALIDATED |
|  | [GENERATED](javascript:ret_Column('GENERATED');) | Varchar2 | 14 | - | - | - | nullable | - | Was the constraint name system generated? - GENERATED NAME or USER NAME |
|  | [BAD](javascript:ret_Column('BAD');) | Varchar2 | 3 | - | - | - | nullable | - | Creating this constraint should give ORA-02436. Rewrite it before 2000 AD. |
|  | [RELY](javascript:ret_Column('RELY');) | Varchar2 | 4 | - | - | - | nullable | - | If set, this flag will be used in optimizer |
|  | [LAST\_CHANGE](javascript:ret_Column('LAST_CHANGE');) | Date | 7 | - | - | - | nullable | - | The date when this column was last enabled or disabled |
|  | [INDEX\_OWNER](javascript:ret_Column('INDEX_OWNER');) | Varchar2 | 30 | - | - | - | nullable | - | The owner of the index used by the constraint |
|  | [INDEX\_NAME](javascript:ret_Column('INDEX_NAME');) | Varchar2 | 30 | - | - | - | nullable | - | The index used by the constraint |
|  | [INVALID](javascript:ret_Column('INVALID');) | Varchar2 | 7 | - | - | - | nullable | - | - |
|  | [VIEW\_RELATED](javascript:ret_Column('VIEW_RELATED');) | Varchar2 | 14 | - | - | - | nullable | - | - |
| |  |  |  | | --- | --- | --- | |  |  | 1-20 | | | | | | | | | | |

To see what all constraints are in our **EMPLOYEES** table.

select constraint\_name,constraint\_type from user\_constraints where table\_name='EMPLOYEES';



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **CONSTRAINT\_NAME** | **CONSTRAINT\_TYPE** | |
| EMPLOYEES\_FIRSTNAME\_NN | C | |
| EMPLOYEES\_EMAIL\_UN | U | |
| EMP\_GENDER\_CHECK | C | |
| EMPLOYEES\_EMPID\_PK | P | |
| 4 rows returned in 0.00 seconds |  | [CSV Export](http://127.0.0.1:8080/apex/f?p=4500:1204:928044562611615::::P1204_BROWSER_LANG:en-us) |

**C** represents Not Null and Check.

To see the Columns wise .

select constraint\_name,column\_name from user\_cons\_columns where table\_name='EMPLOYEES'



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **CONSTRAINT\_NAME** | **COLUMN\_NAME** | |
| EMPLOYEES\_FIRSTNAME\_NN | FIRSTNAME | |
| EMPLOYEES\_EMAIL\_UN | EMAILID | |
| EMP\_GENDER\_CHECK | GENDER | |
| EMPLOYEES\_EMPID\_PK | EMPID | |
| 4 rows returned in 0.02 seconds |  | [CSV Export](http://127.0.0.1:8080/apex/f?p=4500:1204:928044562611615::::P1204_BROWSER_LANG:en-us) |

**Disable Constraints**

alter table Employees disable constraint Emp\_Gender\_check;



**Enable** **Constraints**

alter table Employees enable constraint Emp\_Gender\_check;



**Drop Constrains**

alter table Employees Drop constraint Emp\_Gender\_check;



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:928044562611615::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Table altered.

0.00 seconds

**Data manipulation language (DML)**

* + **SELECT**
  + **INSERT**
  + **UPDATE**
  + **DELETE**

• A **DML** statement is executed when you:

* Add new rows to a table
* Modify existing rows in a table
* Remove existing rows from a table

• A transaction consists of a collection of **DML** statements that form a logical unit of work.

The learning objectives for this lab are:

* To know how to insert, update and delete data from within a table
* To learn how to retrieve data from a table using the SELECT statement

**INSERT**

The SQL **INSERT INTO** Statement is used to add new rows of data to a table in the database.

**Syntax:**

There are two basic syntaxes of the **INSERT INTO** statement which are shown below.

INSERT INTO TABLE\_NAME (column1, column2, column3,...columnN)]

VALUES (value1, value2, value3,...valueN);

Here, column1, column2, column3,...columnN are the names of the columns in the table into which you want to insert the data.

You may not need to specify the column(s) name in the SQL query if you are adding values for all the columns of the table. But make sure the order of the values is in the same order as the columns in the table.

The SQL **INSERT INTO** syntax will be as follows:

INSERT INTO TABLE\_NAME VALUES (value1, value2, value3,...valueN);

[Our existing **Employees** Table we put at least 10 record]

* Character data within **single quotation** (‘ ‘) mark.
* Date put either (**sysdate**) system date or new date.

Insert into Employees values(001,'Ramesh','Mishra','rameshm@gmail.com',



'M','2589636542',sysdate,500,'27-Aug-1971',15000);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

1 row(s) inserted.

0.00 seconds

insert into Employees values(002,'Suresh','Pradhan','sp007@gmail.com',



'M','8722567598','20-Jan-2009',501,'23-July-1989',18000);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

1 row(s) inserted.

0.00 seconds

insert into Employees values(003,'Anmita','Nayak','anayak@gmail.com',

'F','9435862145','18-Jan-2019',502,'13-Aug-1998',30000);



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

1 row(s) inserted.

0.00 seconds

insert into Employees values(004,'Rahul','Bajaj','rb001@gmail.com',

'M','9234856127',sysdate,503,'12-July-1995',20000);



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

1 row(s) inserted.

0.00 seconds

insert into Employees values(005,'Neha','Agrawal','nagra12@gmail.com',



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) |  |

'F','7236518925','2-Jan-2019',504,'18-Dec-1997',25000);

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

1 row(s) inserted.

insert into Employees values(006,'Hardik','Pandia','Hpcri@rediffmail.com',



'M','9643258768','5-April-2018',505,'5-march-1990',21000);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

1 row(s) inserted.

insert into Employees values(007,'Sapna','Sharma','sharmas@Yahoo.com',



'F','6958678541','10-May-2017',506,'17-Nov-1986',19000);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

1 row(s) inserted.

0.00 seconds

insert into Employees values(008,'Rohit','Kumar','rohitku@gmail.com',

'M','8452387624','28-Feb-2016',507,'16-Sep-1994',26000);



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

1 row(s) inserted.

0.00 seconds

insert into Employees values(009,'Khushi','Agrawal','kagrawal@gmail.com',



'F','2569457964','17-July-2015',508,'13-Oct-1997',23000);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

1 row(s) inserted.

0.00 seconds

insert into Employees values(010,'Ganesh','Sahu','sahuganesh@gmail.com',



'M','1348976524','29-June-2019',509,'25-Sep-1983',24000);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

1 row(s) inserted.

0.00 seconds

select \* from Employees;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **EMPID** | **FIRSTNAME** | **LASTNAME** | | **EMAILID** | **GENDER** | **MOBILENO** | **DOJ** | **DEPTID** | **DOB** | **SALARY** |
| 1 | Ramesh | Mishra | | rameshm@gmail.com | M | 2589636542 | 27-APR-20 | 500 | 27-AUG-71 | 15000 |
| 2 | Suresh | Pradhan | | sp007@gmail.com | M | 8722567598 | 20-JAN-09 | 501 | 23-JUL-89 | 18000 |
| 3 | Anmita | Nayak | | anayak@gmail.com | F | 9435862145 | 18-JAN-19 | 502 | 13-AUG-98 | 30000 |
| 4 | Rahul | Bajaj | | rb001@gmail.com | M | 9234856127 | 27-APR-20 | 503 | 12-JUL-95 | 20000 |
| 5 | Neha | Agrawal | | nagra12@gmail.com | F | 7236518925 | 02-JAN-19 | 504 | 18-DEC-97 | 25000 |
| 6 | Hardik | Pandia | | Hpcri@rediffmail.com | M | 9643258768 | 05-APR-18 | 505 | 05-MAR-90 | 21000 |
| 7 | Sapna | Sharma | | sharmas@Yahoo.com | F | 6958678541 | 10-MAY-17 | 506 | 17-NOV-86 | 19000 |
| 8 | Rohit | Kumar | | rohitku@gmail.com | M | 8452387624 | 28-FEB-16 | 507 | 16-SEP-94 | 26000 |
| 9 | Khushi | Agrawal | | kagrawal@gmail.com | F | 2569457964 | 17-JUL-15 | 508 | 13-OCT-97 | 23000 |
| 10 | Ganesh | Sahu | | sahuganesh@gmail.com | M | 1348976524 | 29-JUN-19 | 509 | 25-SEP-83 | 24000 |
| 10 rows returned in 0.00 seconds | | |  | [CSV Export](http://127.0.0.1:8080/apex/f?p=4500:1204:693772775197814::::P1204_BROWSER_LANG:en-us) |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

**Gende**r is Check constraints **M** or **F**, If we write other alphabet its display:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:693772775197814::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

**ORA-02290: check constraint (SYSTEM.EMP\_GENDER\_CHECK) violated**

|  |
| --- |
|  |

**EMPID** is **Primary Key** it cannot be duplicate. If we write same EMPID its display:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [**Results**](http://127.0.0.1:8080/apex/f?p=4500:1003:928683362370519::NO:::) | [**Explain**](http://127.0.0.1:8080/apex/f?p=4500:1003:928683362370519::NO:::) | [**Describe**](http://127.0.0.1:8080/apex/f?p=4500:1003:928683362370519::NO:::) | [**Saved SQL**](http://127.0.0.1:8080/apex/f?p=4500:1003:928683362370519::NO:::) | [**History**](http://127.0.0.1:8080/apex/f?p=4500:1003:928683362370519::NO:::) |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

**ORA-00001: unique constraint (SYSTEM.EMPLOYEES\_EMPID\_PK) violated**

|  |
| --- |
|  |

0.02 seconds

**UPDATE**

The SQL UPDATE Query is used to modify the existing records in a table. You can use the WHERE clause with the UPDATE query to update the selected rows, otherwise all the rows would be affected.

**Syntax:**

The **UPDATE** command is used to modify data in a table. The syntax for this command is:

UPDATE table name

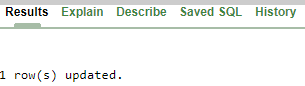
SET column1 = value1, column2 = value2...., columnN = valueN

WHERE [condition];

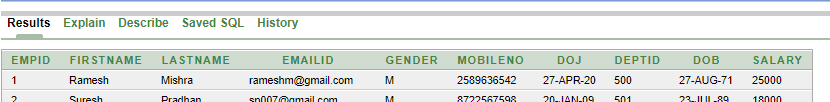


update Employees set salary=25000 where Empid=1;





select \* from Employees;



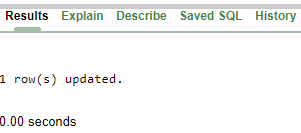
Salary 25000 EmdID - 1

If more than one attribute is to be updated in the row, separate each attribute with commas.

update Employees set MobileNo ='9435681429',



lastname='Bhoi',doj='28=July-19' where Empid =010;





select \* from Employees;



MobileNo-9435681429, lastname-Bhoi,doj-28=July-19 EmdID - 10

**DELETE**

The SQL **DELETE** Query is used to delete the existing records from a table.

You can use the **WHERE** clause with a **DELETE** query to delete the selected rows, otherwise all the records would be deleted.

**Syntax**:

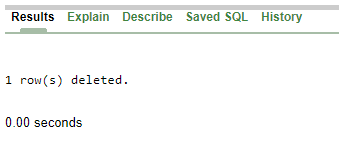
The basic syntax of the **DELETE** query with the **WHERE** clause is as follows:

DELETE FROM table name

WHERE [condition];

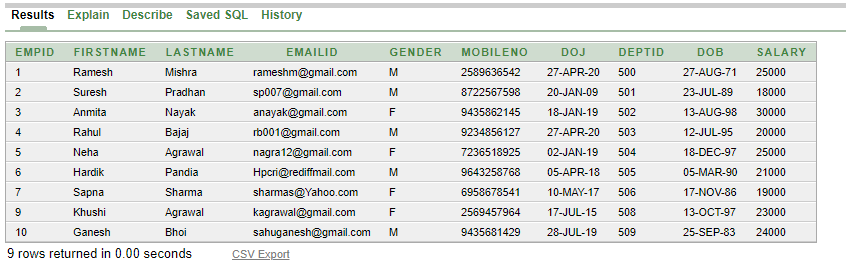


delete from Employees where deptid = '507';





select \* from Employees;



Record DEPTID 507 has DELETED

.

**SELECT**

* Is a Data Retrieval Language
* SELECT clause specifics the columns to search
* FROM clause specifics the data source
* Column search can be modified using expressions

**Syntax:**

The basic syntax of the **SELECT statement** is as follows:

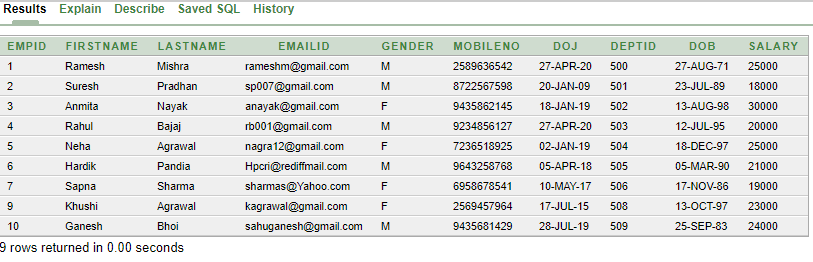
SELECT column1, column2, columnN FROM table name;

Here, column1, column2... are the fields of a table whose values you want to fetch. If you want to fetch all the fields available in the field, then you can use the following syntax.

SELECT \* FROM table\_name;

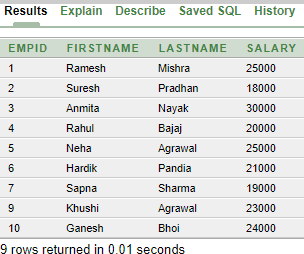


select \* from Employees;





select Empid, Firstname, Lastname, Salary from Employees;



**Concatenation Operator**

You can link columns to other columns, arithmetic expressions, or constant values to create a character expression by using the **concatenation operator (||).** Columns on either side of the operator are combined to make a single output column.

**In the example,**

FIRST NAME and LAST NAME are concatenated, and they are given the alias Employees. Notice that the employee last name and job code are combined to make a single output column.

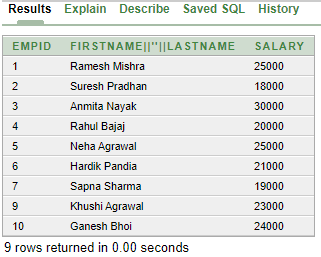
Literal Character Strings:

Literal strings of free format text can be included in the query result and are treated the same as a column in the SELECT list.

Date and character literals must be enclosed by **single quotation marks (' ')**; number literals need not be so enclosed.

select Empid, firstname || ' ' || lastname, salary from Employees;

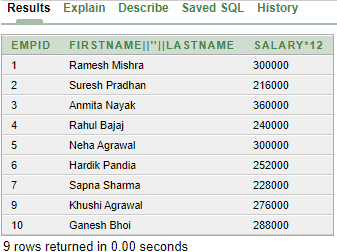




select Empid, firstname || ' ' || lastname, salary\*12 from Employees;



SALARY\*12

c

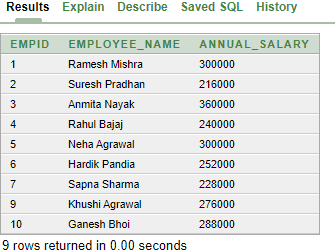
Column Aliases:

When displaying the result of a query, SQL normally uses the name of the selected column as the column heading. You can change a column heading by using a column alias.

Specify the alias after the column in the SELECT list using a space as a separator. By default, alias headings appear in uppercase. If the alias contains spaces or special characters (such as # or $), or if it is case-sensitive, enclose the alias in double quotation marks (" ").

select Empid, firstname || ' ' || lastname as Employee\_Name , salary\*12 as Annual\_Salary from Employees;

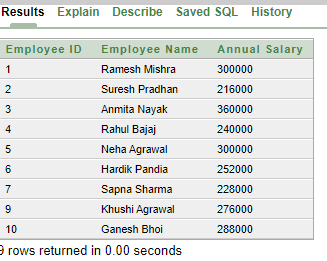




EMPLOYEE\_NAME ANNUAL\_SALARY

select Empid as "Employee ID", firstname || ' ' || lastname as "Employee Name" , salary\*12 as "Annual Salary" from Employees;





Employee ID Employee Name Annual Salary

DISTINCT keyword : Don’t show the Duplicate Rows



select distinct deptid from Employees;

**WHERE Clause**

* The **WHERE** clause is used to add condition to filter.
* Various operators can be used to pass the conditions.

If the given condition is satisfied, then only it returns a specific value from the table.

The **WHERE** clause is not only used in the **SELECT** statement, but it is also used in the **UPDATE n** **DELETE** statement.

**Syntax:**

The basic syntax of the **SELECT** statement with the **WHERE** clause is as shown below.

SELECT column1, column2, columnN

FROM table\_name

WHERE [condition]

You can specify a condition using the **comparison or logical** operators.

Comparison Operators :

|  |  |
| --- | --- |
| Symbol | Meaning |
| = | Equal to |
| < | Less than |
| <= | Less than or equal to |
| > | Greater than |
| >= | Greater than or equal to |
| <> or != | Not equal to |

We will now explore some of these conditional operators using examples.

(**In our Employees Table we have 20 records**)



select \* from Employees;



select empid, firstname, lastname, salary, deptid



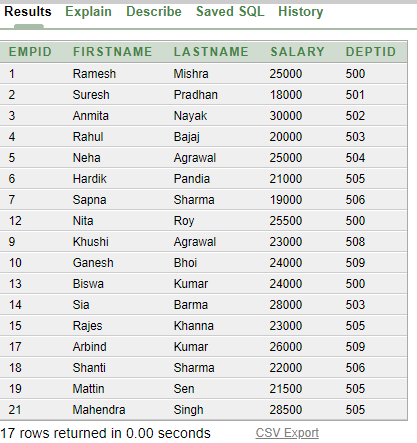
from employees where deptid = 510;



select empid, firstname, lastname, salary, deptid



from employees where deptid != 510;



select empid, firstname, lastname, salary, deptid



from employees where salary >=22000;



|  |
| --- |
| * AND |
| * OR |
| * NOT |

**Logical Operators**

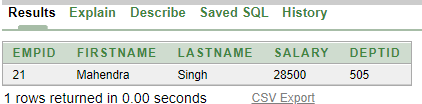
**AND**

For an action to be taken by the SQL statement, whether it be a transaction or a query, all conditions separated by the AND must be TRUE.

select empid, firstname, lastname, salary, deptid



from employees where salary >=25000 and deptid = 505;



select empid, firstname, lastname, salary, deptid



from employees where salary >=21000 and deptid = 510;



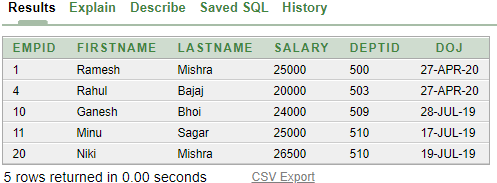
**OR**

An action to be taken by the SQL statement, whether it be a transaction or query, the only any ONE of the conditions separated by the OR must be TRUE.

select empid, firstname, lastname, salary, deptid,doj

from employees where salary >= 31000 or (doj > '10-jun-2019');





**NOT**

The NOT operator can also be used with other SQL operators, such as IN, BETWEEN, LIKE, and NULL.

select empid, firstname, lastname, salary,gender



from employees where gender not in 'M';



**Special Operators**

|  |
| --- |
| * IN |
| * BETWEEN ……AND |
| * LIKE |
| * IS NOL / IS NOT NULL |

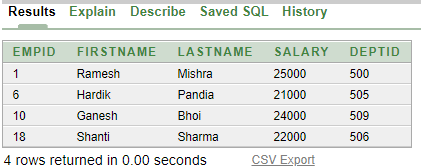
**IN**

The IN operator is used to test for values which are in a list

select empid, firstname, lastname, salary,deptid



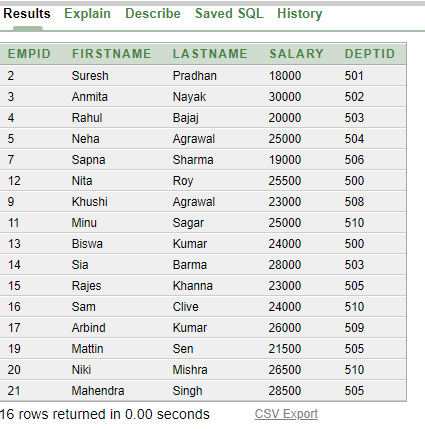
from employees where empid in (1,6,8,10,18);



select empid, firstname, lastname, salary,deptid



from employees where empid not in (1,6,8,10,18);



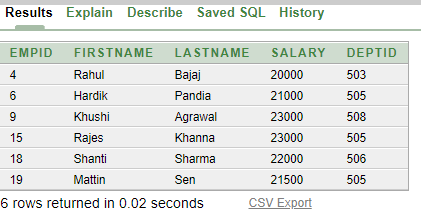
**BETWEEN**

The operator BETWEEN may be used to check whether an attribute value is within a range of values.

select empid, firstname, lastname, salary,deptid



from employees where salary between 20000 and 23000;



**LIKE**

The SQL LIKE clause is used to compare a value to similar values using wildcard operators. There are two wildcards used in conjunction with the LIKE operator.

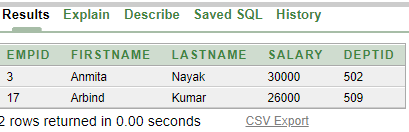
* The percent sign (%)
* The underscore (\_)

The percent sign represents zero, one or multiple characters. The underscore represents a single number or character. These symbols can be used in combinations.

select empid, firstname, lastname, salary,deptid



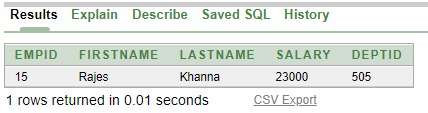
from employees where firstname like 'A%';



select empid, firstname, lastname, salary,deptid



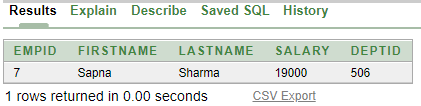
from employees where firstname like '%S';



select empid, firstname, lastname, salary,deptid



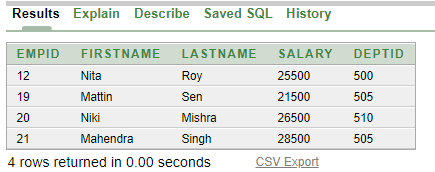
from employees where firstname like 's\_\_\_\_';



select empid, firstname, lastname, salary,deptid



from employees where salary like '%5\_\_';



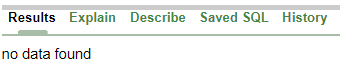
**NULL**

The NULL conditions include the IS NULL condition and the IS NOT NULL condition.

select empid, firstname, lastname, salary,deptid

from employees where deptid is null;





**ORDER BY**

The SQL **ORDER BY** clause is used to **sort the data** in ascending **(ASC)** or descending **(DESC)** order, based on one or more columns. Some databases sort the query results in an *ascending order by default*.

**Syntax:**

The basic syntax of the ORDER BY clause is as follows:

SELECT column-list

FROM table\_name

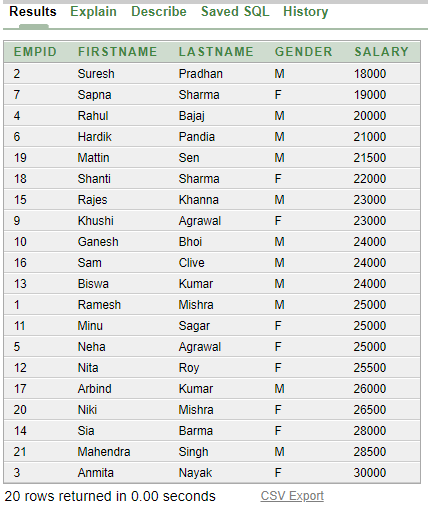
[WHERE condition]

[ORDER BY column1, column2, .. columnN] [ASC | DESC];

select empid, firstname, lastname, gender, salary



from employees order by salary;

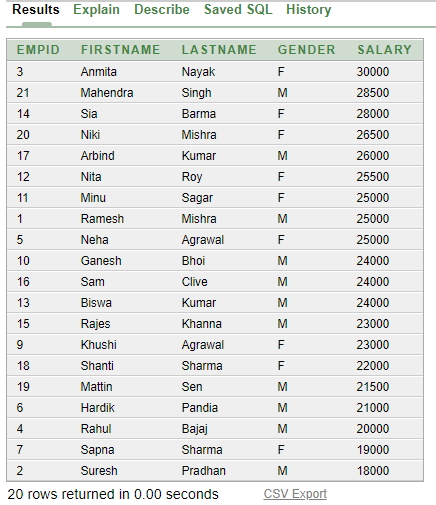


**Sorting in descending order:**

select empid, firstname, lastname, gender, salary

from employees order by salary desc;

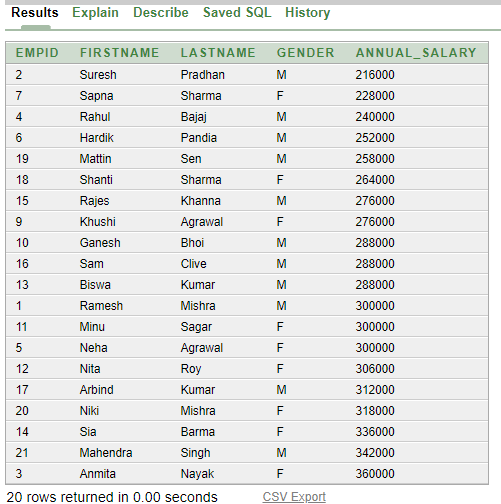




**Sorting by column alias:**

select empid, firstname, lastname, gender, salary \* 12 Annual\_Salary from employees order by Annual\_Salary;



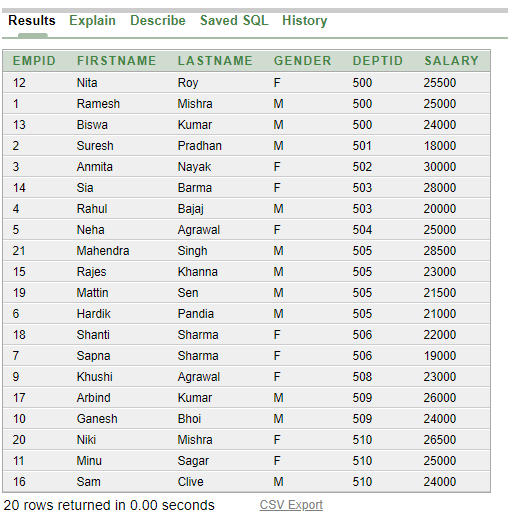


**Sorting by multiple columns:**

select empid, firstname, lastname, gender, deptid, salary

from employees order by deptid , salary desc;

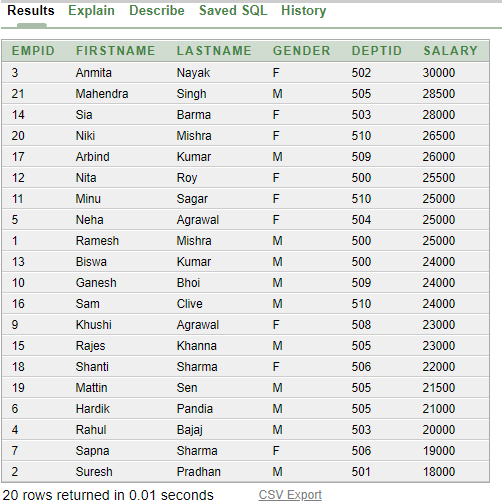




select empid, firstname, lastname, gender, deptid, salary

from employees order by salary desc,firstname ;

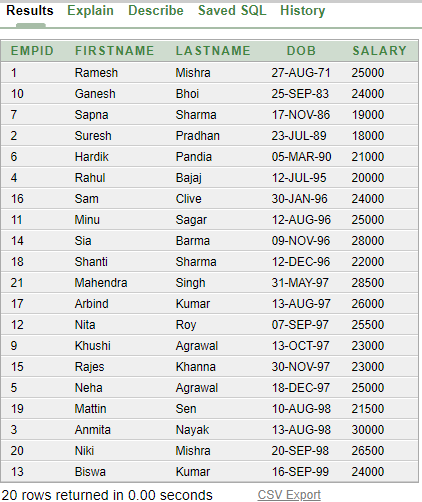




select empid, firstname, lastname,dob, salary

from employees order by dob ;

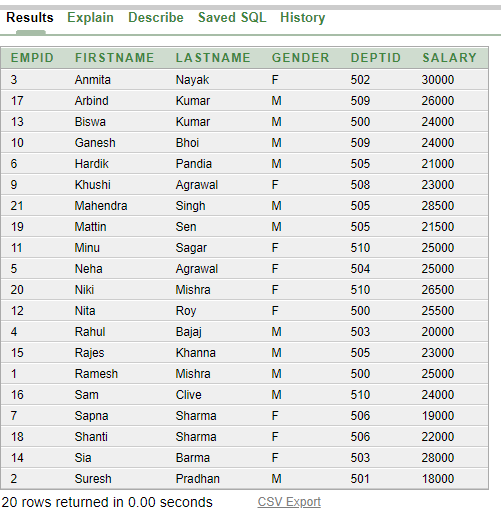




select empid, firstname, lastname, gender, deptid, salary

from employees order by firstname;

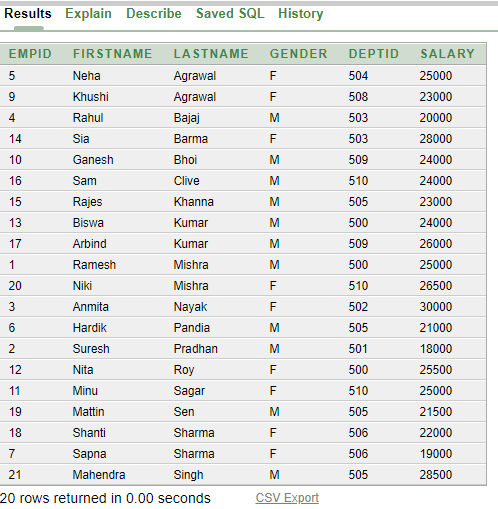




select empid, firstname, lastname, gender, deptid, salary



from employees order by 3;



**SQL Functions**

Functions are a very powerful feature of SQL. They can be used to do the following:

* + Perform calculations on data
  + Modify individual data items
  + Manipulate output for groups of rows
  + Format dates and numbers for display
  + Convert column data types
  + Accept the values through parameters
  + Return the processed value

Two Types of SQL Functions :-

* Single-row function
* Multiple-row function

Single-row function

Single-row functions are used to manipulate data items. They accept one or more arguments and return one value for each row that is returned by the query. There are different types of single-row functions.

* Character functions
* Number functions
* Date functions
* Conversion functions
* General functions

Character functions :

Accept character input and can return both character and number values. Character functions can be divided into the following:

* Case-manipulation functions

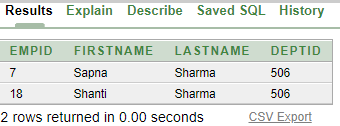
|  |  |
| --- | --- |
| **Function** | **Purpose** |
| LOWER(column|expression) | Converts alpha character values to lowercase |
| UPPER(column|expression | Converts alpha character values to uppercase |
| INITCAP(column|expression | Converts alpha character values to uppercase for the first letter of each word; all other letters in lowercase |

|  |  |
| --- | --- |
| **Function** | **Purpose** |
| LENGTH(column|expression) | Returns the number of characters in the expression |
| CONCAT(column1|expression1,  column2|expression2) | Concatenates the first character value to the second character value; equivalent to concatenation operator (||) |
| SUBSTR(column|expression,m[ ,n] | Returns specified characters from character value starting at character position m, n characters long (If m is negative, the count starts from the end of the character value. If n is omitted, all characters to the end of the string are returned.) |
| RPAD(column|expression,n, ‘string') | Pads the character value left-justified to a total width of n character positions |
| LPAD(column|expression,n,'string') | Pads the character value right-justified to a total width of n character positions |

* Character-manipulation functions

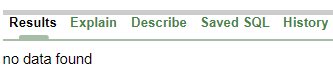
select empid , firstname , lastname , deptid from employees where lastname = 'Sharma';





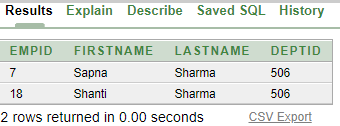
select empid , firstname , lastname , deptid from Employees where lastname = 'SHARMA';





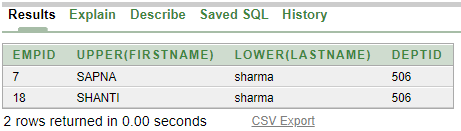
select empid , firstname , lastname , deptid from Employees where lower(lastname) = 'sharma';



37

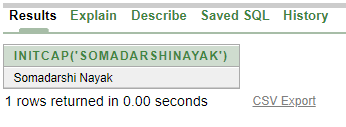
select empid , upper(firstname) , lower(lastname) , deptid from Employees where lower(lastname) = 'sharma';





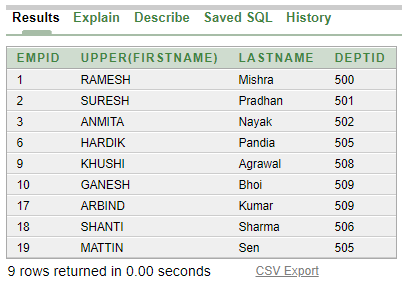


select initcap('somadarshi nayak') from dual;

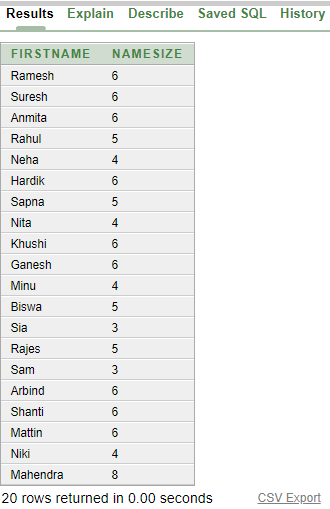


select empid , upper(firstname) , lastname , deptid from Employees where length(firstname) = 6;





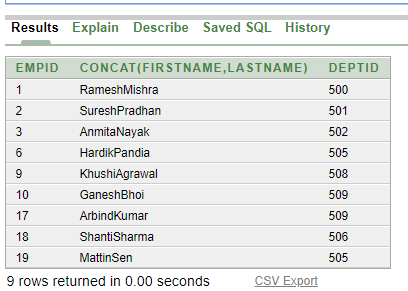
select Firstname , length(Firstname) as NAMESIZE FROM Employees;





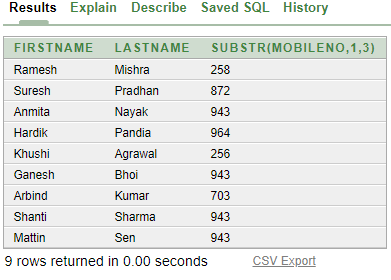
select empid , concat(firstname , lastname) , deptid from Employees where length(firstname) = 6;





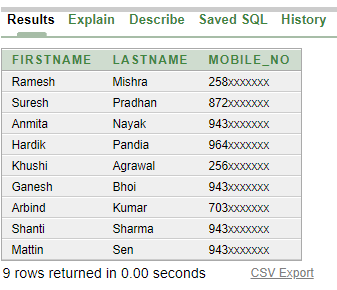
select firstname , lastname ,substr(mobileno,1,3) from Employees where length(firstname) = 6;





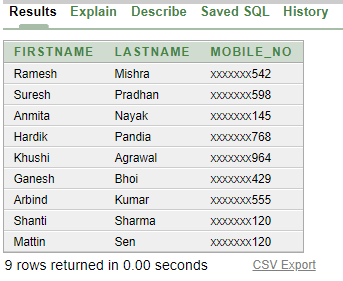
select firstname , lastname ,rpad(substr(mobileno,1,3),10,'x') as Mobile\_No from Employees where length(firstname) = 6;





select firstname , lastname ,lpad(substr(mobileno,8,3),10,'x') as Mobile\_No from Employees where length(firstname) = 6;





Q: - Write a query which generates a list of employee user IDs, using first the day of the month they were born and then the first six characters of their last name in UPPER case.

select Firstname , lastname,concat(substr(dob,1,2), upper(substr(lastname,1,6))) as User\_ID from Employees;



