**Characteristic of primary key—**

* Column assigned as primary key cannot accept null value
* A table cannot have two primary key

Note—it is not mandatory to have primary key in the table, but it is highly recommended.

**4. Foreign Key**

* It is used to relate two tables.
* It can be applied to column level and table level.

**Characteristic of foreign key--**

- A table in which foreign key is present , it is known as child table.

-A table in which actually it belongs is known as parent table.

-Foreign key must and should be defined as primary key in its parent table.

-A column assigned as a foreign key can accept duplicate and null value.

- We cannot insert a value in a column defined as foreign key which is not present in the parent table column.

|  |  |  |
| --- | --- | --- |
| Id | name | Dept id  (foreign key) |
| 111 | palak | 10 |
| 222 | jenny | 20 |

|  |  |  |
| --- | --- | --- |
| Dept id  (primary key) | address | Desg |
| 10 | mumbai | Software engg |
| 20 | mumbai | Software engg |

Child table parent table

**5. Check**

- it is used to restrict the value of a column between a range.

-its like a condition checking before saving data into a column.

-can be applied to column as well as table level.

Difference between primary key and foreign key

Primary key Foreign key

* Do not accept null and duplicate -do accept null and duplicate value

value

* Table level - table and column level
* Primary key uniquely - used to link two table
* We can have only one primary key in table -we can have multiple foreign key

**Data Type—**

* Data type selection is usually dictated by nature of data and by intended use.
* Pay close attention to expected use of attributes for sorting and retrieval purpose.

Types—

1. Numeric

* Bit 0 to 1
* Tinyint 0 to 255
* Smallint -32,768 to 32,767
* Int
* Bigint
* Decimal -10^38+1 to 10^38-1
* Numeric ‘’ ‘’
* Float
* real

1. String

* CHAR fixed length with max length of 8000 character
* VARCHAR variable length , max length 8000
* VARCHAR(max) variable length storage with provided max

character, not supported in sql

* TTEXT variable length storage with max size of 2GB data

Char datatype is of fixed memory allocation , in which memory is wasted

Varchar is variable memory allocation

1. Date and Time datatypes

* DATE stores date in format YYYY-MM-DD
* TIME HH:MI:SS
* DATETIME YYYY-MM-DD HH-MI-SS
* TIMESTAMP stores the number of seconds passed since the unix epoch
* YEAR stores year in 2-digit and 4-digit format

1. Miscellaneous datatype

-CLOB(character large object)

-BLOB(binary large object) Is use to store picture, audio & video

-XML

-JSON

**NOTE—while creating a table ,data type is mandatory to be assigned to the column but constraints are not mandatory but it is highly recommended**

**HISTORY**

-IBM was the first company to develop DBMS which follow Relational model and known as system arc.

-IBM developed a language to communicate with system arc or relational model. The language name was SEQUEL(Simple English Query Language)

-Due to its simplicity it was very popular at that time. In 1980, ANSI(American National Standard Institute ) acquired SEQUEL language and after doing some modification , it was then known as SQL and it was made standard language for RDBMS.

cv

**NOTE-- is case insensitive**

**SQL is a language and MySql is an database application**

**SQL—Structured Query Language**

* A standardized programming language which is used for storing and managing data in databases.
* SQL was the first commercial language introduced for E.F Codd’s Relational model of database.
* With SQL, you can modify, add , update or delete rows of data , retrieve subsets of information from a database and manymore
* Operations are written as statements.

**5 SQL Statement Categories—**

1. Data Definition Language(DDL) create, alter ,drop ,truncate and rename
2. Data Manipulation Language(DML)

Insert , update, delete

1. Data Control Language(DCL)
2. Transactional Control Language(TCL)
3. Data Query Language(DQL) select

* The SQL statement used to retrieve the data from database is known os DQL.
* Only one command to retrieve the data from database is:

Select

* We can retrieve data in three different ways:

1. Projection -- If we want to select some particular column from table then we use particular

SELECT column\_name from table\_name;

1. Selection -- if we want to select particular row from the table then we use selection

SELECT \*from table\_name;

1. Joins -- retrival of data from multiple table, at the same time, it is known as Joins.

SELECT \*from table\_name;

From is use to select the table from database and will put it for execution. We can pass table\_name as an argument.

Select is used to select columns from the table which is under execution and is also responsible to prepare the result set.

Select(\*) all the values .

Order of Execution—

1. From
2. select

* Clauses in SQL
* SQL statements consist of multiple clauses.
* This clause is a subprogram which accepts some arguments.

\* Where clause

- it is used to filter records.

- it is used to extract only those record that fulfil a specified condition.

- select insert

Syntax—

SELECT \*from table\_name

WHERE condition(column name=value);

Order of execution-

1. From table\_name
2. Where condition
3. Select

**Truncate v/s Delete**

**-**if we want to delete table and complete data, we use truncate

-If we want to delete a particular field of a table, we use delete command.

Note---we use drop to delete the structure of table and data also.

**EXPRESSION—**

-A statement which gives result or output is called expression.

C=a+b + is the operator, a and b are operands

* In sql operands are of two types –

1. Column
2. Literals -- actual value

select salary,salary+5000 from emp\_info;

salary and 5000 are the operands in which salary is column and 5000 is literal

literals are of three types—

1. Number
2. Character (+‘abc’) -it is case sensitive
3. Date (‘2019-08-08’)

12-08-19

**ALIAS—**

- Are used to give Temporary name to column or a table.

- It is often used to make column name readable

-->ALIAS COLUMN SYNTAX—

SELECT column\_name AS alias\_name FROM table\_name;

-->ALIAS TABLE SYNTAX—

SELECT column\_name(s) FROM table\_name AS alias\_name;

**DISTINCT—**

* It is used to return only distinct(different) values

SYNTAX--

SELECT DISTINCT column1, column 2,……. FROM table\_name;

* If more than one argument is passed to distinct, it will remove the combination of duplicate column.

**OPERATORS—**

* An opretors is reserved word or a character used primarily in an SQL statements.

TYPES—

* Arithmetic operator
* Relational/ comparison operator
* Logical operator
* Special operator

1. Additional operators—

+, -, \*, /, %

1. Relational operator—

=, !=, <>(check if two values are equal or not, if not equal then it returns true), > , < , >=, <=

1. Logical operators—

AND( allow the existence of multiple conditions in WHERE clause)

A& B are the two condition

Case 1: if both the condition are true , then output will be true,

Case 2: if A is true , then it will check the B condition. If it is False, then output will be false

Case 3: if A is false then it will not check the B condition , the output will be false

Case 4: if both A & B are false the result is false.

SYNTAX—

Select \* from table\_name where column\_name=value && column\_name=value;

OR( “ “ ),

NOT( will alter the result)

14 aug

**OR Operator**

Case1) if the first condition is true then the result of or operator is true without evaluating the second one

Case2) If the first condition is false then the result depends on second condition

Syntax: select \* from table\_name where column\_name=value or column\_name =value ;

**NOT operator- excluding**

Syntax: select\* from table\_name where not column\_name = value ;

Order of execution is

Not

And

Or

**SPECIAL Operator**

**1)IN Operator-**used to evaluate multiple of values.

Syntax: select\* from table\_name where column\_name IN(list of values);

2) **NOT IN Operator** : is a multivalue operator it will exclude multiple values

Syntax: select\* from table\_name where column\_name NOT IN(list of values);

3**) BETWEEN Operator :** is used for searching based on range of values

Syntax: select\* from table\_name where column\_name between lower limit and upper limit

4) **IS Operator :** to check null value if any record is null it will give true if record is not null it gives false.

Syntax: select\*from table\_name where column\_name is null;

5) **LIKE Operator:** used for pattern matching

Syntax: select\* from table\_name where column\_name like ‘ ‘ ;

Character are of 2 types

1. Ordinary character:normal a,b,c
2. Special Character – percentage(used when there is no limitation of characters) represents 0 or n character

Underscore- represent only one character of the string

Syntax: select \* from table\_name where column\_value like ‘% / \_ ‘ ;

6) **NOT LIKE :** select all the string which do not match with the given string

**FUNCTIONS**

Is a block of code which performs tasks

Function Attributes:it has input argument function name and return type.

In sql has some predefined function

1. Scalar Function (single row function): for every input we get corresponding output.

Divided into 2parts

1. Case Manipulation single arrow function

UPPER() -converts the value to field to uppercase

LOWER() : lowercase

INIT(cap): first char should be caps (doesnot work in my sql only works in oracle)

Grant

II) Character Manipulation

TYPES:

* **LENGTH(**):returns the length of the value in text field.
* **CONCAT():**is used to concatenate two strings.
* Select concat(arg1,arg2) from employee\_info;
* **REVERSE():** function Is used to reverse a string.
* **SUBSTR()** : it is a function used to extract a string from a given string.it accepts three arguments.we get a new string.
* Syntax: select substr(string we get a new string,position no.which specify position in a string ,length-specifies length of new string)
* **INSTR :** is used to check whether a substring is present in a given string or not.if a substring is not present I a given string then the function vl return 0;it s used to check the position.
* SYNTAX: SELECT INSTR (string ,substring) from table\_name;
* Argument: string:string in which substring has to be checked.
* **REPLACE**: used to replace a character in a given string
* Syntax: replace(string,substr1,substr2)

1. Aggregate Function(multi row function) : pass multiple input but single output

Types :

AVG()-returns average value

COUNT() -COUNTS the numbr of rows

MAX()-returns maximum value of the selected column

MIN()-returns minimum value of selected column

SUM()-returns sum of all values of selected column

Syntax:

Select avg(salary) from table\_name;

ALTER

-TO add new column

ALTER TABLE table\_name ADD column\_name column-definition;

* To drop column

ALTER TABLE table\_name DROP COLUMN column\_name;

TRUNCATE

TRUNCATE table table\_name

RENAME

-to change the table name

RENAME TABLE table\_name to new\_table\_name

INSERT

INSERT INTO table\_name(column\_name, column\_name) values(values,values);

UPDATE

UPDATE table\_name set column\_name=value;

DELETE

DELETE from table\_name;

**GROUP BY:**

Group by statement group rows have same values into summary rows.

It is often used with aggregate functions (count,max,min)

SYNTAX:

Select column\_name(s) from table\_name where condition group by column\_name(s);

Eg: select count(\*) from employee group by deptno;

**Note: groupby clause execute row by row after the execution of groupby the records are grouped. Therefore al the clause vl execute after the execution of group by clause.Group by clause vl execute group by ,where and select.**

**HAVING CLAUSE:**

It is often used with the group by clause to filter groups based on a specified condition.

In having clause we can use only a column i.e used inn group by clause.

Syntax:

Select column\_name from table where condition group by column\_name HAVING condition

Order of execution:

From

Where

Groupby

Having

Select

***NOTE: i) having clause executes after the execution of group by clause,therefore it checks the given condition after grouping.***

***ii)Since it executes after grouping having clause vl execute group by group***(for every group it vl check the condition)

***iii)Having clause can be used in columns used by group by***

***ORDER BY:***it is used to sort the records in ascending or descending order if the result set neither be in ascending or decending order. By default it is in ascending order.

SYNTAX: select \* from employee\_info order by deptid desc;

* Order of execution:
* From
* Where
* Group by
* Select
* Order by

**NOTE: orderby clause always executes after execution of select class.**

**Orderby clause should always be the last clause to be written in the syntax.by defaut sort ascending.**

**If the group by clause is used then in orderby clause v can use aggregate functions or multirow functions or only the column name which is used in the group by clause**

**SUBQUERY**

a query written inside a query.

Also known as nested query or inner query.subquery is also another way h to fetch the data from multiple tables.it should be written inside a parenthesis.

**NOTE**: Always inner query should be executed first and the output of inner query will be taken as an input for outer query,the final result vl get from outer query.

**When to use subquery in where clause**?

**Scenario1**:If the given condition contains unknown value then we can use subquery to obtain the unknown value.

**Scenario 2**: If the column to be selected and condition to be filtered for two different table we can use subquery.

Two types of subquery:

1. Single row subquery.
2. Multi row subquery.

**SINGLE ROW SUBQUERY:**

If the single value operators such as (=,<,>,<=,!=) are used

We call it as a single row subquery

**Syntax:**

Select column\_name from table where column1 **operator** (select column from table where expr1=value);

**MULTIROW SUBQUERY**

If the operator such as multi value operator (in) are used

Can return one or more rows.

Syntax: select colum1,colum2 ,… from table where column1 operator (select column from table where expr1=value);

**JOINS**

Joins is used to fetch data from two or more tables.Merging of two tables or more than two tables.

Merging of more than one table horizontally is called joins.

Minimum required condition for joining table is atleast 2.

**TYPES OF JOINS:**

1. **CROSS JOIN (cartesian)**

A record from one table is merged with each and every record of the other table.or join two tables records from a left table will be merged with each n every record of right table such joins is known as cartesian joins.

Syntax: select column\_name from table\_name cross join table\_name1

|  |  |
| --- | --- |
| A1 | B1 |
| a | 10 |
| b | 20 |
|  |  |

**Inner join**

Joining two tables with the help of join column is called as inner join.

Syntax: Select \*from table1 inner join table2 on condition.

Oracle Syntax:

Select\*from table,table where condition;

**Outer Join:**

**3 Types:**

**Left outer join , Right outer join, Full outer join.**

**Left Outer Join: gives inner joins + unmatched record of left table .**

Unmatched record- the record who do not have pair in the opposite table with respect to join

Syntax :Select\* from table1 left outer join table2 on employee\_info.deptId =department\_info.deptId;

**Right outer join: gives inner joins+ unmatched record of right table.**

**Full Outer Join:**

Full outer join gives inner join plus al the records from both the table who do not have any pair.

Syntax:

**Key in RDBMS**

**Key Attributes:** A key attribute is used to find an attribute using which we can uniquely determine the record in a table.

**Non key Attributes:** All the attributes except key attributes are known as non- key attributes.

**Prime Key Attributes :** A key attribute which is chosen to be a main attribute to determine the record uniquely in the table.

**Non Prime Key Attributes:** A key attribute which is not chosen to be a main attribute to determine the record uniquely in the table.

**Composite Key :**The combination of two or more attribute which determines the record uniquely.

**Foreign Key:**  is used to relate two tables, the table in which foreign key is present is called child table and where it belongs is called parent table n can accept null an duplicate values.

**Functional Dependency:** In a functional dependency a relation exists such that attributes determine another attributes uniquely.eg R= { x ,y}

x is determinant and unique and y is dependent.

**Types of Functional Dependency:**

1)Total F.D:

If all the attributes of a relation is determined by a key attribute is known as total f.d. eg: in a family all are dependent on father in terms of money.

2)Partial F.D: Relation is set to have partial functional dependency if

* It consists of composite key attribute
* Their exist a dependency such that the attribute will be determined by another attribute which is a part of composite key.

e.g in a family if husband and wife (a,b) -> (c,d)

3)Transitive F.D

A relation is said to be transitive functional dependency if there exists a relation such that a attribute is determined by non key attributes which interns is determined by key attribute.

a=>c

,here a is a key attribute

c=>d

**NORMALIZATION:**

**Data Redundancy:** same data is repeated many times. To overcome this problem we are using data redundancy.

**Anamoly:**The side effect that occur during performing DML operations(insert,update,delete) operation.

Normalization means thprocess of decomposing table into smaller table by removing redundancy and anamoly by identifiying dependency is called normazlization.The process of reduncing the table in normal form.

**Normal Form:** A state of table without table redundancy.

**Types of Normal Form:**

* **(1 NF) First Normal Form:** a table is set to be in first normal form if this satisfy the following condition:
* table should not have duplicate rows.
* Every cell should have a single value

1. **2NF :** tables are 2NF If follows condition:

* table should be in 1nf form.
* Table should not have partial functional dependency.(not hava composite key)
* **Note: If the table consists partial dependency functionality the attribute which are responsible will be removed from the table.**

1. **3NF:** the table is said to be in 3nf if the following condition is satisified:

* it should follow 2nf and should not have transitive functional dependency.

**Note: 1nf n 2nf and 3nf these are based on keys attributes and functional dependency of relational schema.**

**ER DIAGRAM:**  Entity relationship describe the structure of database with the help of diagram .It is a blueprint of database that can be implemented as database.