* Database:
  1. consider database as a container where an all the data is stored .The database is basically a systematic collection of the data which supposed to storage and manipulation of the data that you stored
* c>create
* u>update
* r>read
* D> delete
* DBMS:
  1. collection of programs which enables users to access database, manipulate data and represent data, A technology to store and retrive data with utmost efficiency along with appropirate security measures.
* ----------------
* TYPES OF DBMS:
* hierarchial
* -style of parent child relationship to store the data
* -it has a structure like tree with nodes
* EG: windows Registry used in window-XP
* network
* -supports many to many dbs
* -complex
* EG:RDMS
* \*RElational DBMS
* defines reationships in the form of tables also known as relations
* mySQL
* \*Object oriented DBMS
* -in the form of objects
* EG: post SQL
* -----------
* ADV of DBMS
  + - improved data sharing
    - improved Dats security
    - Reduction in data redundancy
    - minimized data inconsistency
    - interaction with user

Disadv

* -inceased costs
* -compleity of backup and recovery
* -frequent upgrade
* -------------
* DBMS which organize and store the data by following the relational model is known as RDBMS
* -Relational model is designed by E.F CODD
* -In relational model all the data is stored in the form of relations
* -a/c to codd data in r-model should be logically organized and stored in the form of tables
* ----TABLE----
* -table is a logical organization of data and consits of rows and columns
* -verical-column-used to represent the property of entity, Horizantal-row also called as tuples.
* -row is used to represent the property of all the individual entity
* -intesection of row en column generates the cell
* -cell is the smallest unit of table which is used to store the data
* ---
* RDBMS can be used to validate in two ways
* -constraints
* -datatypes
* \_\_\_\_\_\_\_\_\_\_\_
* Constraints
* >NOTNULL
* >UNIQUE
* >FOREGIN KEY
* >PRIMARY KEY
* >CHECK
* >DEFAULT
* -----------
* NOTNULL
* >conatraints restricts a column from having a NULL value. once you applies NOTNULL costraints to a column you cannot pass a null vallue to that column
* >it cannot be applicable on tale level
* -----
* UNIQUE
* >Unique constraints ensure that a field or column will only have unique values,will not have an duplicate data
* >can be apppllied at column level
* ---------
* primary key
* >this constraints uniquelyidentifes
* each record n the database
* >
* >
* prrimary key must cntaio=
* >>CHARACTERISTICS OF PRIMARY KEY
  + - combo of
* -
* -can have only primary key
* \_\_ it is not mandtory to have primarykey in the table but it is recommend
* -----------
* table in which foreign key is present is known as child table
* a table in which actually belongs<primary> 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 terminal can accept duplicates or null values

* >>
* We cannot insert a values in a column defined as a foreign key which is not present in the parent table column

|  |  |  |
| --- | --- | --- |
| Depid | dname | location |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| Eid | ename | Depid |
|  |  |  |
|  |  |  |

|  |  |
| --- | --- |
| Primary key | Foregin key |
| * Primary key is a parent key * Uniquly identified in the tabl * Cannot accept null value | * Foreign key is a child key |

7-Aug-19

* DATATYPES
* DATATYPE ----------------------------- Range

INT

* Tinyint 0-1
* Smallint 0-255
* Int -32768 - 32767
* bigint
* decimal
* numeric
* float
* real

SQL character and string Datatypes:

* character fixed length of 8000 characters
* varchar fixed length of 8000 characters
* varchar(max) variable length storage with provided max characters not supported in MYSQL
* TEXT-
* “char datatype is fixed length of memory location”

Rahul consist of 5characters but in char remaining 3 bits will be a wastage

* Varchar is a variable length in memory allocation

But In varchar it takes sufficient storage only

* SQL DATE AND TIME DATA TYPES:
  + Date -yyy-MM-dd
  + Time -HH-MI-SS
  + Datetime- YYYY-MM-DD-HH-MI-SS
  + Timestamp
  + Year

SQL MISSELLANEOUS DATA TYPES

* + Binary large is used to store Pictures,audio,video
  + CLOB- Character large objs that can hold upto 2GB
  + XML -storing xml data
  + JSON -storing JSON data

NOTE: while creating a table datatype is mandatory to assign to the column,but constraints are not mandatory, but it is highly recommended to have contrsint

IBM dvev

Or relaitona mo

Due to its simplicity it was very populara at tin

In 1980-ANSI(AMERICAN NATIONLA STANDARD INSTITIUTE),ACQUIRED SQL language as,after doing some modification then it d amd made standard language for RDbMS

* SQL is case insensitive
* Sql is a language, MySQL is an application\database

SQL

* Structured Query Language
* A standardized language which is used for storing and managing data in databases
* First commercial language introduced for E.F.Codd relational model od database
* With SQL, you can modify db`s,add, update,delete rows of data,retrives subsets os db…
* Rdb`s- MYsQL dtatabase,oracle, mysql server,Sybase
* Queries and other SQL operations-insert,add,update,delete,alter,truncate,select
* 5 SQL STATEMENT CATEGORIES
  + DDL -DATA DEFINITION LANGUAGE
    - Create,alter,truncate,rename,drop
    - Truncate table table\_name
  + DML-DATA MANIPULATION LANGUAGE
    - Insert,update,delete
    - Insert into table\_name (col\_name1,col\_name2) values (value\_1,value\_2);
  + DCL-DATA CONTROL LANGUAGE
  + TCL-TRANSACTIONAL CONTROL LANGUAGE
  + DQL-DATA QUERY LANGUAGE
    - Select-select \* from table\_name;
    - Only one command to retrieve the data
* There are three different ways to retrieve the data from the table
  + Projection
    - It is used to select the columns from the table which is under execution
    - Select col\_name,col\_name from table\_name;
  + Selection
    - Select \* from table\_name; -from used to select a table in the db and put under execution
      * We can pass table name as a argument
      * \*------selects all the columns
      * Order of execution

From

select

* + Join
    - Retrival of data from multiple of table at the same time
* ***CLAUSE***
  + Is clause is a sub program with expects some argument
* Where clause
  + Used to filter records
  + Doesn`t support for insert operation
  + Used to retrieve only those records that fulfil the specific condition
  + Select \* from table\_name where condition;
  + ***Eg: where (col\_name=value)->condition;***
  + ***ORDER OF EXECUTION -***

***FROM***

***WHERE***

***SELECT***

* Truncate- to delete all the data regarding the table.
* Delete – to delete the particular rows under condition using where clause.
* Drop – TO Delete both regards and the structure of the table.
* Expression – which gives some output
  + It consits of operator and an operand
  + *Eg: c=a+b;*
    - *a,b,c are operands, += are operators*
  + literal- it is the actual value passed in the expression
  + *eg: sal=sal+5000*
  + *literals are 3 types*
    - *numbers -64*
    - *characters -‘endoidi’-----------CASE SENSITIVE*
    - *date -‘2019-8-8’*
* ALIAS
  + Alias column
  + Alias table
* Distinct- to get the unique value from the table
  + SELECT DISTINCT C1,C2,…….. FROM TABLE\_NAME;
  + ***eg: select distinct country from customers;***
  + ***select count (DISTINCT country) from customers;***
    - ***if more than one argument is passed to distinct it will remove the combination of column duplicates***
* **OPERATORS- IS A RESERVED WORD OR A CHARACTER USED TO PERFORM AN OPERATION**
  + **Arithmetic operator**
  + **Comparison operator**
  + **Logical operator**
  + **Special operator**

***ARITHMETIC OPERATOR***

* ADD - a+b
* SUB - a-b
* MUL - a\*b
* DIV - b/a
* MOD - b%a

***COMPARISON OPERATOR***

* ***Equalto =***
* ***Not equalto !=***
* ***Not equalto <>***
* ***Greaterthan***
* ***Less than***
* ***Greaterthan or equal to >=***
* ***Less than or equal to <=***

***LOGICAL OPERATOR***

* **used to pass more than one condition**

***AND -***

***COND1 COND2 RESULT***

***<T T T>***

***T F F***

***F T F***

***F F F***

***OR -<F-F=F>***

***COND1 COND2 RESULT***

***T T T***

***T F T***

***F T T***

***F F F***

***Case-1:*  if the first cond. is true then the result of or operator is true without checking the second cond,.**

***Case-*2: if the first cond. is false then the result will depend on the second condition,**

***If the second cond.* id true it gives true, otherwise it gives false.**

***SYNTAX-* select \* from table-name where col\_name=vale or col\_name=value;**

***NOT -<T=F/F=T>***

***COND2 RESULT***

***F T***

***T F***

***SYNTAX-* select \* from table-name where not col\_name=value;**

***ORDER OF EXECUTION: 1----🡪NOT***

***2----🡪AND***

***3----🡪OR***

* ***SPECIAL OPERATORS:***

***Used to evaluate multiple values***

* + ***IN - used to apply or for multiple values***
    - ***Syntax: select \* from employee where col\_name IN(list of values);***
  + ***NOTIN - used to exclude multiple values***
    - ***Syntax: select \* from employee where col\_name NOTIN(list of values);***
  + ***BETWEEN- USED FOR searching based on the range of values***
    - ***Syntax: select \* from employee where col\_name BETWEEN LOWER\_LIMIT AND UPPER\_LIMIT;***
  + ***NOTBETWEEN- used for excluding the range of values***
  + ***ISNULL- if any record is null then it gives true, f it is not null it will give false***
    - ***Syntax- sele***
  + ***LIKE- used for pattern matching***
    - ***Select \* from employee where col\_name like “special character”***
    - ***{characters are of two types:***

***ordinary***

***Special}***

***%(percentage)- matches o or n characters***

***\_(underscore)- matches only one charater***

* + - ***Select \* from table\_name where col\_name like ‘%/\_’;***
  + ***NOTLIKE***- used to select all the string which do not match with the given string

***FUNCTION is a block of code tat performs a specific task***

***Sql has some predefined function***

***FUNCTION*** ATTRIBUTES: it has

* Input argument
* Function name
* Return type

Predefined functions

* Scalar function/single row function
* Aggregate function/multirow function

***Aggregate function***

***(or)***

***multirow function:***

***We pass only one input but the output is single***

----function()

***--------------🡪***

---------🡪 -------------------🡪

***--------------🡪***

* AVG()-returns average value
* COUNT()-counts no. of rows
* MAX()-returns maximum values of the selected column
* MIN()-returns minimum values of the selected column
* SUM()-returns sum of all the values of the selected column

***SINGLEROW FUNCTION***

------------🡪 ----------🡪R1

----function()

------------🡪 ----------🡪R2

--------🡪 ----------🡪R3

UPPER()-

Lower()-

LENGTH()-

CONCAT()-

REVERSE()-

* Case manipulation:-

Upper()-convert a col\_name in given table into uppercase

Lower()-convert a col\_name in given table into uppercase

Initcap()-start letter should be in “caps”// only applicable for oracle

* Charater manipulation:-

Length()-gives the length of the strings in a particular column

Concat()- used to concatenate two strings

* Substr():-
  + - * A function used to extract a string from a given string, it accepts 3 arguments
      * SYNTAX-select substring as a new string
      * If it is a number which spectified
* Instr():-
  + - used to check whether a given substring is present in the string or not
    - If the substring is not present in the string then it gives-0 other wise
    - It is a string in which substring has to be searched
* Replace:-
  + - It is used to replace the character with the
* Groupby:
  + Group the rows that have the same values into summary rows
  + Groupby clause executes row by rows after the execution of row by row he records are grouped therefore all the all the clauses will be excuted after the execution of group by caluse
  + Groupby statements often used with aggregate functions (count,min,max,avg,sum)
  + Syntax- select count(\*) from table-name group by Col-names;
  + OREDER oF EXECUTION:-

From

Groupby

Where

Select

* Having\_clause:
  + Is often used with the group by clause to filter groups based on a specialized condition
  + In having clause we can use only the column that is used in by group clause
  + SYNTAX:
    - * Select col\_names from table-name where condition group by col\_names having condition
  + ORDER OF Execution

From

where

groupby

having

select

* + - having clause executes after the execution of group by caluse
    - therefore it checks the given condition after grouping
    - it executes after grouping
    - having clause executes group by group
    - having clause can be used only by a column which is in groupby
* ***ORDERBY:***
  + - The order by clause used to srt the result-set in ascending or descending order
    - The order by keyword sorts the records sorts the records in ascending order by default
  + Syntax:
    - Select column,col2 from table\_name order by col1,col2…….. ASC|desc;
    - Select column,col2 from table\_name where cond order by col1,col2…….. ASC|desc;
  + Order of execution
    - From
    - Where
    - Select
    - Order by
* Innerjoin : joining two tables with the help of join column is known as inner joins
* ***Outer joins:***
* *Left outer join:*

Left outer join gives inner join + Unxmatch record of left table

The record donot have a pair in the project table

All the reecord from both the tables who donot have any pair

====================NORMALIZATION======================

* Key Attributes:

A key attribute is used to find an attribute using which we can uniquely determine the record in the table.

* Non-key attributes:

All the attributes except key attributes are known as non-key attributes.

* Prime Key Attributes:

Key Attribute which is chosen to be a main attribute to determine the record uniquely in the field.

* Non Prime Key Attributes:

Key Attribute which is not chosen to be a main attribute to determine the record uniquely in the field.

* Composite Key:

The combination of two attributes which determines a record uniquelyis called composite key

* Foreign Key:

Foreign key is used to link two tables, using a primary key in another table, with same name as that col-name

**Functional Denpendency:**

In a functional dependency a relation exist such that an attribute determines the another attribute uniquely is known as functional dependency.

R->{x,y}

x- determiner

y- Dependent

* Types of functional dependency:
* Total:

All the attributes of a relation is determined by a key attributeis known as total functional dependency

* Partial:

A relation is set to have partial functional dependency,

If it consist of composite key attributes

There exist a dependency that can be determined by another attribute which is in the composite key

* Transitive:

If a relation is said to be a transitive functional dependency if there exist a relation such that an attribute is determined by non-key attributes which intern determined by a key attributes.

Eg:

a->c

c->d

🡺a->d

repitition

Redundancy

Anomaly-the side effect that occur by performing DML operation is known as anomaly

Types of anomalies- insert anomaly, update anomaly,delete anomaly

the process of decomposing the table into a smaller table in order to remove redundancy and anomalies by identifying dependencies is known as ***NORMALIZATION***

***or***

***the process of reducing the tables into normal forms is known as normalization***

***🡪a state of a table without any redundancies and anomalies is known as normal form.***

***TYPES OF NORMAL FORMS:***

* ***1NF*** –
  + a table is said to be in the first normal form if it satisfy the following cond.

1. Should not have duplicate rows
2. Every rows in the table should be single value

* ***2NF***
  + In second normal form

1. ***No duplicates***
2. NO partial functional dependency,(composite key)

NOTE: if the table consist partial functional dependency, the attribute which are responsible will be removed from the table

* ***3NF***
  + ***The table*** is said to be 3rd normal form if the following conditions are satisfied

i.should be 2NF

***ii.*** should have transitive functional dependency

NOTE: 3nf and 2nf these are based on the key attributes and functional dependencies of Relational Schema

* ***BCNF***
* ***5NF***
* ***6NF***
* ***7NF***

Single row functions,number\_date

***E-R Diagram***

Describes the structure of database with the help of diagram which is known as E-R diagram

It is a blue print of database that can be used to implement as a database

* Component of E-R diagram

**Entity - represents table**

**Relationship - single combines all the cmponents**

**Attribute -represents**

E-r diagram

addtecxt

student

college

VIEW-

View is a logical and virtual tables that can be created on the existing table

View donot occupy memory

**For creation:**

Create view view\_name select \* from table;

**For display:**

**Select** \* from view\_name

Note:

any dml operation performed on view will be reflected in that base table

**TCL:**

**RollBack: undo**

**mysql> set autocommit=0;**

**Query OK, 0 rows affected (0.00 sec)**

**Commit:**

**hhhhh**You cannot rollback once you have got committed

**mysql> set autocommit=0;**

**Query OK, 0 rows affected (0.00 sec)**

**Savepoint:**

**Usually work with rollback**

**Stored procedures:**

**Call to procedur\_name;**

**Create->select\* from empl-> selectall-> execute all->**