Data Management System

DBMS is a software for storing and retrieving users data while considering appropriate security measures. It consists of a group of programs which manipulate the database. The DBMS accepts the request for data from an application and instructs the operating system to provide the specific data. DBMS allows users to create their own database as per their requirement. The term "DBMS" includes the user of the database and other application programs. It provides an interface between the data and the software applications.

Few examples of database are :- Dictionary, Phone directory, Stock inventory details etc.

Advantages of DBMS

Some advantages of computerized management.

1) Improved Data Sharing :- The database management system helps to create an environment in which end users have better access to more and better managed data such access makes it possible for end users to respond quickly to change in their environment.

2. Data Sec : The more users access the database, the greater the risks of the data security. Corporation invest considerable amount of time, effort and money to ensure that data are used properly. A DBMS provides a framework for better enforcement of data privacy and security.

3. Creating backups very easy : In computerized database management system we can create backups of our data very easily but in manual database it is very very difficult

4. Data Redendancy :- Data redendancy is not possible in computerized DBMS but in data ~~not~~ redendancy or we can say it as duplication of data.

5. Data Integration : Data integration is possible only in computerized DBMS but in manual system their is no mechanism of data integrity.

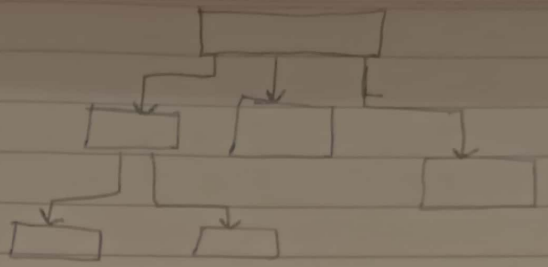
6. Time Consumption : Manual database is time consuming as compare to computerized DBMS.

7. Refrencial Integrity : In computerized DBMS refrencial integrity is possible test not in manual database.

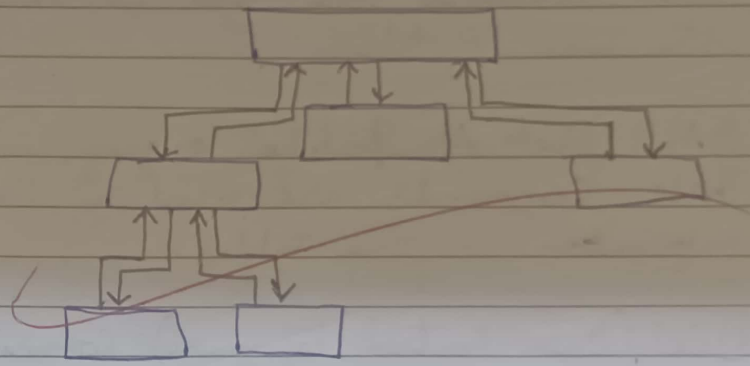
Models of Data Base

These are three ~~modul~~ ~~mod~~ models of Database

i) Hierarchical :



ii) Networking :



iii) Relational : (In table state)

Table or relation

Field/Attribute/Property

Roll No	Name	Class	Marks
101	ATiF	12th	390
102	BISM	12th	410
103	ADU	12th	342
104	Diya	12th	243
105	Arif	12th	423
106	Nasir	12th	450

Record/Tuple

SQL is a database language through which we can create or implement our data base. There are 4 types of commands used in SQL and these are:

i) DDL (Data defination language) commands:

These commands are used for changing the structure of tables.

ii) DML (Data Manipulation language) commands:
These commands are used to change the information in the tables.

iii) DCL (Data control language) commands:
These commands are used to help the user to provide the access.

iv) DTL (Data Transaction language) command:
With the help of these commands we can limit the Transaction.

Commands:

1) Creating a Table:

```
Create Table EMPP( EmpCode Varchar(5),
Name Varchar(15), Designation Varchar(15),
DoJ Date, Salary Number(5));
```

highlight whole command and then press run.

→ To show the structure of the table:

```
Desc EmPP
```

→ To insert data in the table created above:

```
Insert Into EmPP Values ('E004', 'TABISH', 'MANAGER',
'12-Jan-2019', 32600);
```

- To check the table with the data inserted in it:

```
Select * from EMPP;
```

- To show the specific columns of your table:

```
Select Name, Salary, from EMPP;
```

-> Shows only a specific data of your table
Select * from EMPP where Designation = 'Manager';

-> Shows employees having salary >= value

-> To insert less information than required without encountering any error:

Insert into table name (column name in which you want to insert the data, ...) values ("Enter the data for the columns you have written above");

Example:-

Insert into EMPP (EMPCODE, Name, Designation) values ('E008', 'Khalid', 'Operator');

-> To update the data of the table

Update EMPP Set Name = 'Tabish' where EMPCODE = 'E004';

Note: we need an unique identity in the table to update any information of the existing data.

- To edit the structure of the created table:
Alter table EMPP Modify Name VARchar(25);

- To delete any specific row:
Delete from EMPP where EMPCODE = 'E008';

- To add a new column to a existing table:
Alter table EMPP ADD Branch varchar(15);

- To add values in the above created new column:
Update EMPP Set Branch = 'Solina' where EMPCODE = 'E006' or EMPCODE = 'E008';

To Delete the column.
Alter Table (table name) Drop column (column name);
Example:
Alter table EMPP Drop column Branch;

Constraints: These are the rules which we can implement on our tables to Validate our data or we can say, to check data integrity of our data.
There are 4 types of constraints used in SQL and these are:

1. Unique: This constraint will not allow any duplicate entry in the table.

2. Check: The constraint Check ~~work~~ works where there are any conditions for the data to be entered in any column.

3. Not Null: NotNull will not allow the user to skip the column that is neccesary to be filled

4. Primary key: Primary key is that by which our row is ~~two~~ uniquely identified.

Create Table student (Roll No Number(3), Name varchar(15), Class Varchar(5) Not Null, Marks Number(3));
Insert into student Values (101, 'Vaid', '12th', 350);
Desc student;
Create Table student (Roll No Number(3) Name Varchar(15) Not Null, class varchar(5), Marks Number(3));
~~Create~~ Insert into student value (101, 'Varun', '12th', 319);
Create Table student (Roll no number (3) primary key, Name varchar (15), class varchar(5), Marks Number(3) check (Marks <= 500));
Insert into student values (104, 'Tanvar', '12th', 248);
Select * from student order by marks desc;
Select * from student ~~where~~ order by marks desc;
Insert into student value (~~114~~, ~~Dinesh Tanwar~~, ~~11th~~, ~~324~~ 334);
Select * from student where class not in ('12th', '11th');
Select Distinct (class) from student; select class, count (*) from student group by (class);
Select class, count (*), sum (marks) from student group by (class);
Select count (*) from student where class != '12th';

Aggregate function :- An aggregate function performs a calculation on a set of values and returns a single value. Except for Count (*), Aggregate functions are often used in SQL:

i) Sum(): To get the total of any column.
ii) Count(): To count the no. of posts, employees etc
iii) Min(): To get the minimum value of anything from any table
iv) Max(): To get the maximum value from any table
v) Average(): To get the average

Logic Gates : There are three logical operators namely "And", "OR" and NOT. These operators compare two conditions at a time to determine whether a row can be selected for the output, we can use logical operators in the "where" clause, which allows us to combine more than one condition.

i) And: If you want to select rows that must satisfy all the given conditions, you can use the logical operator, 'And'

Example :- Select * from Employe where Designation = 'operator' And Salary >= 15000;

ii) OR: If you want to select rows that should satisfy at least one of the given condition, you can use the logical operator, 'OR'
Example :- Select * from employe where Designation = 'operator' OR Salary >= 15000;

iii) NOT/NOT IN: If you want to find rows that do not satisfy a condition you can use the logical operator, 'NOT'. NOT results in the reverse of a condition that is, if a condition is satisfied, then the row is not returned
Example :- Select * from employe where Designation NOT IN ('Sweeper', 'Manager', 'operator')

→ Init cap : Shows the first letter of the selected list in capitals.

Example : Select Init cap (Name), Init cap (Designation), Salary from Employe;

Name	Designation	Salary
Hinna	Manager	30,000
UZair	Operator	15,000
Mansha	Assistant	18,000
Tajamul	Operator	14,000

→ lower and upper: "lower" shows the whole selected list in the lower case but in "upper" The selected list will be displayed in the upper case form.
Example: Select upper (Name), lower (address) init cap (Designation) from Employee;

	Name (upper)	Address (lower)	Designation (init cap)
1	HINNA	ram bagh	Manager
2	UZair	nowgam	Operator
3	Mansha	Solina	Assistant
4.	Tajamul	baghat	Operator

→ Concat :- Concat function is a SQL String function that provides to concatenate two or more than two character expressions into a single string.
Example:- Select Name, Address, concat (Name, address), Substr (concat, (Name, address), 1 length (Name) from Employee;

→ Length: To get the number of characters in a string in SQL.

→ RTRIM:- The RTRIM function removes all space characters from the right hand side of a ~~strong~~ string.

Example: Select Rollno, Name, Class, RTRIM (class, 'm') from Student;

→ LTRIM:→ The LTRIM function removes leading spaces ~~from~~ from a string (from the left end).

Numeric Commands

- SQRT :- Returns the square root of a given value in the argument. An expression which is a numeric value data type.

Example : Select Name, Address, salary, SQRT (salary) from ~~Salary~~ employe;

Select Name, class, marks, SQRT (marks), Round (SQRT (marks); 0) from students;

Select Name, class, Marks, SQRT (Marks, Round (SQRT (marks); 0) from student;

Select Name, Class, Marks; SQRT (Marks), floor (SQRT (marks)) from student;

- Power : Power() function returns the value of a number raised to another, where both of the numbers are passed as arguments. The SQL Distinct command along with the SQL power() function can be used to retrieve only unique data depending on a specified expression;

Example : Select Name, Address, Salary, Power (Salary) from Employe; (2)

→ MOD() :- MOD() function is used to get the remainder from a division.

Example : Select Name, Address, Salary, Mod (salary) from Employe;

Referential Integrity Constraints

Referential integrity requires that a foreign key must have a matching primary key or it must be null. This constraint is specified between two tables (parent and child); it maintains the corresponding between rows in these tables. It means the reference from a row in one table to another table must be valid.

Foreign Key

Admission block

Roll No.	Name	Parentage	Class	DoB
101	Abid	Mr Ishtiyaq	12th	1-Jan 1997
102	Talib	Mr Amin Bhat	12th	21-Jan 1997
103	Aiman	Mr Gh. Mohd	12th	30 Mar 1992
104	Naiya	Mr. Riyaz Dar	12th	12 May 1992

Book ID	Book Name	Author	DOI	Roll no
B013	Monk mind	Jay Shetty	2-April 2020	102
C004	C++	Ravi Kadre	3-Feb 2018	104

Aliases: Aliases are used to give a table, or a column in a table, a temporary name. Aliases are often used to make column names more readable.

Example: Select Empcode ID, Name Employee, Designation Post, Salary Pay from Employee;

Create table Stud 1 (Rollno, Number(3), Name ~~to~~ varchar (15) Class varchar (5));

Insert into stud 1 values (104, 'Tanveer');

Select * from Stud

Union
Select * from Stud 1;

Select * from stud

Intersect
Select * from Stud 1;

Select * from Stud

Minus
Select * from Stud 1;

Joining of Tables.

Select * from Stud;
Select * from Stud;

Select S. Roll no, S1. Roll no. S. Name, S1. Name S. class, S1. Class from Stud S, Stud 1 S1 where S. Roll no = S1. Roll no:

Select * from employe;

Select E1. Name, E1. Designation, E1. DoJ, S1. Name, S1. Class from Employe E1, Stud1 S1;

Select S. Roll no, S1, Roll no, S. Name, S1. Name S1 Class, S1 class ~~fro~~ from Stud S, Stud1 S1 Where S, Roll no(+) = S1. Roll no;

Superintendent ITI AFCPL (PVT.) Rambagh. Srinagar Reg. No: PU01000105

Checked & verified by me on the date .../10/2022