Introduction to SQL

SQL is domain specific used in design and management of data held in RDBMS.

- can define structure of a DB
- Modify data in DB
- Specify security constraints
- Based on relational Algebra and tuple relation calculus.
- IBM developed original version of SQL called SEQUEL in 1970

Part of SQL

1- Data Definition Language
2. Data Manipulation Language
3. ~~Transation~~ Transaction control Language.

⇒ For data retrival in SQL, i/p and o/p both are relations.

⇒ No of relation i/p to query will be at least one but o/p will always be a single relation without any name unless specified.

→ Basic structure of SQL query consist of three clauses.

- Select
- from
- where

→ It should be noted that 'select' and 'from' are mandatory and if not required then not necessary to write 'where'.

→ SQL in general is not case sensitive.

→ SQL allows duplication in a i/p relation as well as in the result of SQL expressions.

Select

- is used to pick column required in result of the query out of all columns in I/p relation.

- Order in which column are represented in the select clause will be same in which column will appear in the result.
- If all columns are required then * can be used.
- may also contains arithmetic expressions involving operators like +, −, /, * etc. however it does not change to the database.

↓

Select * marks+5

Q1 Find all the details of Bank Branches ?

```
Select *
from branch
```

↓

```
Select branch_name, branch city, assets
from branch.
```

Q2 find the name of all customers having Bank A/c

```
Select
from
```

↓

```
Select Cust_name
from Depositor
```

→ Select distinct cust_name from Depositor

Q3 find each loan no along with amount ?

```
select
from
```

↓

```
Select loan NO, amount
from Loan
```

Qu find all A/c NO and balance with 6% yearly interest?

```
Select A/c NO, balance * 1.06
from Account.
```

Select with where

- where clause in SQL is used to specify a condition/predicates
- where clause can have expressions involving comparison operator <, <=, >, >=, = etc.
- SQL allows use of logical connectives, and. or, not etc.
- allows 'between' comparison operator to ~~simplify~~ simplify where clauses.

Qu all A/c NO where balance is less than 1000

```
Select A/c NO
from ACCOUNT
where balance < 1000
```

Q4 find branch name which is situated in delhi and having assets < 100000

```
select  branch_name
from    Branch
where   assets < 100000
        and b_city = Delhi
```

```
select  branch_name
from    Branch
where   branch_city = Delhi
        and assets < 100000
```

~~NOTE → and → follows Commutative law.~~

Q5 find b_name and A/c NO where balance is >= 100 and <=1000.

```
Select  B_name, A/c NO
from    Account
where   balance >= 100
        and
        balance <= 1000
```

```
Select  B_name, A/c NO
from    Account
where   balance between 100 and 1000
```

NOTE → between support boundary values.