

SQL QUERIES	
QUERY NAME	SYNTAX AND EXAMPLE
1. TO CREATE TABLE IN SQL DATA BASE	
create table	Syntax--> create table TN (CN DataType(DataSize) , CN DataType(DataSize).....);
	Example--> create table Velocity (Name varchar(10), Age int , Address varchar2 (10));
2. TO INSERT RECORDS OR DATA IN TABLE	
insert	Syntax--> insert TN into values ('value1', 'value2',.....);
	Example--> insert Velocity values ('Vaibhav', '25', 'Pune');
3. FETCH OR RETRIEVE ALL RECORDS FROM TABLE	
select	Syntax--> select *from TN;
	Example--> select *from Velocity;
4. FETCH OR RETRIEVE ONLY ONE COLUMN FROM TABLE	
select	Syntax--> select CN from TN;
	Example--> select Name from Velocity;
5. FETCH OR RETRIEVE MULTIPLE COLUMNS FROM TABLE	
select	Syntax--> select CN1, CN2 from TN;
	Example--> select Name, Age from Velocity;
6. FETCH OR RETRIEVE ONE ROW FROM TABLE	
select, where	Syntax--> select *from TN where CN='value';
	Example--> select *from Velocity where Name='Vaibhav';
7. FETCH DATA USING LOGICAL OPERATORS	
select, where, and/or	Syntax--> select *from TN where CN='value' and CN='value';
	Syntax--> select *from TN where CN='value' or CN='value';
	Example--> select *from Velocity where Name='Vaibhav' and Age='25';
	Example--> select *from Velocity where Name='Nilesh' or Age='25';
8. UPDATE EXISTING RECORDS WITH NEW ONE	
update, where	Syntax--> update TN set CN='value' where CN='value';
	Example--> update Velocity set address='Wakad' where name='Patil';
9. ADD NEW COLUMN IN TABLE	
alter table, add	Syntax--> alter table TN add CN Data_Type (Data_Size);
	Example--> alter table Velocity add marks int;
10. DELETE ONE COLUMN FROM TABLE	
alter table, drop	Syntax--> alter table TN drop column CN;
	Example--> alter table Velocity drop column marks;
11. MODIFY DATA TYPE OF ANY COLUMN WHERE COLUMN SHOULD HAVE ONLY NULL VALUES	
alter table, modify	Syntax--> alter table TN modify CN New DataType(DataSize);
	Example--> alter table Velocity modify marks varchar2(10);

SQL QUERIES	
QUERY NAME	SYNTAX AND EXAMPLE
12. RENAME ANY COLUMN NAME	
alter table, rename	Syntax--> alter table TN rename oldCN to NewCN;
	Example--> alter table Velocity rename marks to marks1;
13. RENAME TABLE NAME	
alter table, rename	Syntax--> alter table TN rename to newTN;
	Example--> alter table Velocity rename to JulyA;
14. DELETE ALL RECORDS FROM TABLE (EXCEPT TABLE STRUCTURE)	
delete	Syntax--> Delete from TN;
	Example--> Delete from Velocity;
15. DELETE ONLY ONE ROW FROM TABLE (EXCEPT TABLE STRUCTURE)	
delete	Syntax--> Delete from TN where CN='value';
	Example--> Delete from Velocity where name='Vaibhav';
16. DELETE ENTIRE TABLE WITH ITS STRUCTURE FROM DB	
drop table	Syntax--> Drop table TN;
	Example--> Drop table Velocity;
17. FIND MAXIMUM VALUE FROM ONE COLUMN OF TABLE	
select, max	Syntax--> select max(CN) from TN;
	Example--> select max(marks) from Velocity;
18. FIND MINIMUM VALUE FROM ONE COLUMN OF TABLE	
select, min	Syntax--> select min(CN) from TN;
	Example--> select min(Age) from Velocity;
19. FIND AVERAGE OF ALL VALUE FROM ONE COLUMN OF TABLE	
select, avg	Syntax--> select avg(CN) from TN;
	Example--> select avg(marks1) from Velocity;
20. FIND SUM OF ALL VALUE FROM ONE COLUMN OF TABLE	
select, sum	Syntax--> select sum(CN) from TN;
	Example--> select sum(marks) from Velocity;
21. TO COUNT TOTAL NUMBER OF RECORDS FROM ONE COLUMN OF TABLE	
select, count	Syntax--> select count(CN) from TN;
	Example--> select count(name) from Velocity;
22. DISTINCT- TO FETCH ONLY UNIQUE VALUES FROM ONE COLUMN	
select, distinct	Syntax--> select Distinct CN from TN;
	Example--> select Distinct marks from Velocity;
23. USE COMPARISON OPERATORS TO FETCH DATA FROM TABLE	
select, where < > <= >= = !=	Syntax--> select *from TN where CN > value;
	Example--> select *from Velocity where marks <= 65;

SQL QUERIES	
QUERY NAME	SYNTAX AND EXAMPLE
24. UNIQUE CONSTRAINTS--> FORCES COLUMN NOT TO ACCEPT DUPLICATE VALUES (ACCEPT ONLY UNIQUE VALUES)	
create table, unique	Syntax--> create table TN(CN1, CN2, CN3, CN4 Unique,.....CN-N);
	Example--> create table Velocity(Name, Age, Id Unique, Mobile Unique);
25. NOT NULL CONSTRAINTS--> FORCES COLUMN NOT TO ACCEPT NULL VALUES	
create table, not null	Syntax--> create table TN(CN1, CN2, CN3 Not Null, CN4.....CN-N);
	Example--> create table JulyA(Name, Marks, Address Not Null);
26. PRIMARY KEY CONSTRAINTS--> FORCES COLUMN NOT TO ACCEPT NULL VALUES & DUPLICATES	
create table, primary key	Syntax--> create table TN1 (CN1, CN2, CN3 Not Null, CN4, Primary Key(CN2));
	Example--> create table Mockg (UserName varchar(15), Password varchar(15), Mobile int, primary key(mobile));
27. FOREIGN KEY CONSTRAINTS--> USED TO JOIN TWO TABLES. FOREIGN KEY IS A FIELD IN SECOND TABLE THAT REFERS TO PRIMARY KEY OF FIRST TABLE	
create table, foreign key	Syntax--> create table TN2 (CN1, CN2, CN3 Not Null, CN4, Foreign Key(CN2) references TN1);
	Example--> create table part2 (Name varchar(15), Address varchar(15), Pincode int, Mobile int, foreign key(mobile) references Mockg);
28. TO SORT RECORDS IN ASCENDING OR DECENDING ORDER	
select, order by	Syntax--> select CN from TN order by CN asc;
	Syntax--> select CN from TN order by CN desc;
	Example--> select marks from Velocity order by marks asc;
	Example--> select id from Velocity order by id desc;
29. FETCH RECORDS USING BETWEEN QUERY	
select, where, between	Syntax--> select *from TN where CN between value1 and value2;
	Example--> select *from Velocity where marks between 45 and 65;
30. LIKE QUERY--> TO FETCH RECORDS WHICH STARTS WITH GIVEN ALPHABET	
select, where, like	Syntax--> select *from TN where CN like A%;
	Example--> select *from Velocity where name like A%;
31. LIKE QUERY--> TO FETCH RECORDS WHICH ENDS WITH GIVEN ALPHABET	
select, where, like	Syntax--> select *from TN where CN like %A;
	Example--> select *from Velocity where name like %A;
32. LIKE QUERY--> TO FETCH RECORDS WHICH IS HAVING GIVEN ALPHABET AT ANY PLACE	
select, where, like	Syntax--> select *from TN where CN like %A%;
	Example--> select *from Velocity where name like %A%;
33. FULL JOIN--> WILL DISPLAY ALL RECORDS FROM BOTH THE TABLES	
full join	Syntax--> select a.CN1, a.CN2, a.CN3, b.CN1, b.CN2, b.CN3 from TN1 a full join TN2 B on a.CN2 = b.CN3;
	Example--> select a.username, a.password, a.mobile, b.name, b.address, b.pincode From part1 a full join part2 b On a.mobile=b.mobile;

SQL QUERIES	
QUERY NAME	SYNTAX AND EXAMPLE
34. LEFT JOIN--> WILL DISPLAY ALL RECORDS FROM 1st TABLE & ONLY COMMON RECORDS FROM 2nd TABLE	
left join	Syntax--> select a.CN1, a.CN2, a.CN3, b.CN1, b.CN2, b.CN3 from TN1 a left join TN2 B on a.CN2 = b.CN3;
	Example--> select a.username, a.password, a.mobile, b.name, b.address, b.pincode From part1 a left join part2 b On a.mobile=b.mobile;
35. RIGHT JOIN--> WILL DISPLAY ALL RECORDS FROM 2nd TABLE & ONLY COMMON RECORDS FROM 1st TABLE	
right join	Syntax--> select a.CN1, a.CN2, a.CN3, b.CN1, b.CN2, b.CN3 from TN1 a right join TN2 B on a.CN2 = b.CN3;
	Example--> select a.username, a.password, a.mobile, b.name, b.address, b.pincode From part1 a right join part2 b On a.mobile=b.mobile;
36. INNER JOIN--> WILL DISPLAY ONLY COMMON RECORDS FROM BOTH THE TABLES	
inner join	Syntax--> select a.CN1, a.CN2, a.CN3, b.CN1, b.CN2, b.CN3 from TN1 a inner join TN2 B on a.CN2 = b.CN3;
	Example--> select a.username, a.password, a.mobile, b.name, b.address, b.pincode From part1 a inner join part2 b On a.mobile=b.mobile;