



**KALINGA INSTITUTE
OF INDUSTRIAL TECHNOLOGY**

Deemed to be University U/S 3 of the UGC Act, 1956

DBMS

LAB ASSIGNMENT

NAME : Arditya Ranjan Pradhan

Roll no : 20051620

Section : CSE - 25

Date:- 10-01-2022 Aditya Ranjan Bradhan 20051620

SQL Fundamentals

Question -1

- (Q) Create a table STUDENT with the attributes like student_Name, Roll, Department, Age, Gender, Semester, city, CGPA . Enter at least 10 entries into that table as shown below and perform the following operation.

NAME	ROLL DEPT	AGE GENDER	SEM CITY	CGPA	BAL_FEE
Rahul	1 CSE	18 Male	4 BBSR	7.5	30000
Anand	2 CSE	20 Male	4 New Delhi	8.5	20000
Anjali	3 CSE	17 Female	4 Kolkata	8.46	50000
Priya	4 CSE	19 Female	4 Pune	9.5	10000
Aakash	6 CSCE	18 Male	6 Varanasi	7.5	30000
Aayush	5 CSCE	19 Male	6 Kanpur	8.5	20000
Naman	7 CSCE	20 Male	6 Jaipur	9.3	50000
Shreya	8 CSCE	18 Female	6 Allahabad	9.3	20000
Shalu	9 IT	19 Female	8 Kolkata	8.3	5000
Silajeet	10 CSCE	20 Male	6 Kolkata	9.5	25000

First we create the table .

Create table STUDENT (Name Varchar(15), Roll Number(5), Dept Varchar(25), Age Number(3), Gender Char(8), SEM Number(1), city Varchar(20), CGPA Number(4,2));

Inserting data into table .

Insert into STUDENT (Name ,Roll ,Dept ,Age ,Gender ,Sem ,city ,CGPA) values ('Rahul' ,1 , 'CSE' , 18 , 'Male' , 4 , 'BBSR' , 7.5);

Insert into STUDENT (Name ,Roll ,Dept ,Age ,Gender ,Sem ,city ,CGPA) values ('Anand' ,2 , 'CSE' , 20 , 'Male' , 4 , 'NEW Delhi' , 8.5);

Insert into STUDENT (Name ,Roll ,Dept ,Age ,Gender ,Sem ,

city, CGPA) values('Anjale', 3, 'CSE', 17, 'Female', 4, 'Kolkata', 8.4)

Insert into STUDENT (Name, Roll, Dept, AGE, Gender, Sem, City, CGPA) values('Priya', 4, 'CSE', 19, 'Female', 4, 'Pune', 9.5);

Insert into STUDENT (Name, Roll, Dept, Age, Gender, Sem, City, CGPA) values('Akash', 6, 'CSCE', 18, 'Male', 6, 'Varansi', 7.5)

Insert into STUDENT (Name, Roll, Dept, Age, Gender, Sem, City, CGPA) values('Aayush', 5, 'CSCE', 19, 'Male', 6, 'Kanpur', 8.5)

Insert into STUDENT (Name, Roll, Dept, Age, Gender, Sem, City, CGPA) values('Naman', 7, 'CSCE', 20, 'Male', 6, 'Jaipur', 9.3);

Insert into STUDENT (Name, Roll, Dept, Age, Gender, Sem, City, CGPA) values('Shreya', 8, 'CSCE', 18, 'Female', 6, 'Alibabad', 9.3);

Insert into STUDENT (Name, Roll, Dept, Age, Gender, Sem, City, CGPA) values('Shalu', 9, 'IT', 19, 'Female', 8, 'Kolkata', 8.3);

Insert into STUDENT (Name, Roll, Dept, Age, Gender, Sem, City, CGPA) values('Siljeet', 10, 'CSCE', 19, 'Male', 6, 'Kolkata', 9.5);

20051620

Aditya Ranjan Pradhan

a) Display the content of the table.

Select * from STUDENT;

NAME	ROLL DEPT	AGE GENDER	SEM CITY	CGPA
Rahul	1 CSE	18 Male	4 BBSR	7.5
Anand	2 CSE	20 Male	4 New Delhi	8.5
Anjali	3 CSE	17 Female	4 Kolkata	8.46
Priya	4 CSE	19 Female	4 Pune	9.5
Aakash	6 CSCE	18 Male	6 Varanasi	7.5
Aayush	5 CSCE	19 Male	6 Kanpur	8.5
Naman	7 CSCE	20 Male	6 Jaipur	9.3
Shreya	8 CSCE	18 Female	6 Allahabad	9.3
Shalu	9 IT	19 Female	8 Kolkata	8.3
Silajeet	10 CSCE	19 Male	6 Kolkata	9.5

10 rows selected.

b) Display the student Name and Roll for all the female students.

Select Name, Roll From STUDENT where
gender = 'Female';

NAME	ROLL
Anjali	3
Priya	4
Shreya	8
Shalu	9

c) Display the roll, age and gender for the students from CSE department.

Select Roll , Age , Gender from STUDENT where
Dept = 'CSE';

ROLL	AGE GENDER
1	18 Male
2	20 Male
3	17 Female
4	19 Female

d) Display the student Name, Roll, Dept and city for the student having CGPA greater than 8.5

Select Name, Roll, Dept, city From STUDENT
Where CGPA > 8.5;

NAME	ROLL	DEPT	CITY
Priya	4	CSE	Pune
Naman	7	CSCE	Jaipur
Shreya	8	CSCE	Allahabad
Silajeet	10	CSCE	Kolkata

e) Enter a new column Balance - Fee and update all the entries accordingly.

alter table STUDENT add BAL-FEE number (10);

Updating the table

update STUDENT set BAL-FEE = 30000 where Roll = 1;

update STUDENT set BAL-FEE = 20000 where Roll = 2;

update STUDENT set BAL-FEE = 50000 where Roll = 3;

update STUDENT set BAL-FEE = 10000 where Roll = 4;

update STUDENT set BAL-FEE = 30000 where Roll = 6;

update STUDENT set BAL-FEE = 20000 where Roll = 5;

update STUDENT set BAL-FEE = 50000 where Roll = 7;

update STUDENT set BAL-FEE = 20000 where Roll = 8;

update STUDENT set BAL-FEE = 5000 where Roll = 9;

update STUDENT set BAL-FEE = 25000 where Roll = 10;

SELECT * FROM STUDENT;

Aditya Ranjan Pradhan 20051620

NAME	ROLL DEPT	AGE GENDER	SEM CITY	CGPA	BAL_FEE
Rahul	1 CSE	18 Male	4 BBSR	7.5	30000
Anand	2 CSE	20 Male	4 New Delhi	8.5	20000
Anjali	3 CSE	17 Female	4 Kolkata	8.46	50000
Priya	4 CSE	19 Female	4 Pune	9.5	10000
Aakash	6 CSCE	18 Male	6 Varanasi	7.5	30000
Aayush	5 CSCE	19 Male	6 Kanpur	8.5	20000
Naman	7 CSCE	20 Male	6 Jaipur	9.3	50000
Shreya	8 CSCE	18 Female	6 Allahabad	9.3	20000
Shalu	9 IT	19 Female	8 Kolkata	8.3	5000
Silajeet	10 CSCE	19 Male	6 Kolkata	9.5	25000

10 rows selected.

f) Delete a student with roll 5 from the table & display the table.

Delete From STUDENT where Roll = 5;

Select * From STUDENT;

1 row deleted.

SQL> SELECT * FROM STUDENT;

NAME	ROLL DEPT	AGE GENDER	SEM CITY	CGPA	BAL_FEE
Rahul	1 CSE	18 Male	4 BBSR	7.5	30000
Anand	2 CSE	20 Male	4 New Delhi	8.5	20000
Anjali	3 CSE	17 Female	4 Kolkata	8.46	50000
Priya	4 CSE	19 Female	4 Pune	9.5	10000
Aakash	6 CSCE	18 Male	6 Varanasi	7.5	30000
Naman	7 CSCE	20 Male	6 Jaipur	9.3	50000
Shreya	8 CSCE	18 Female	6 Allahabad	9.3	20000
Shalu	9 IT	19 Female	8 Kolkata	8.3	5000
Silajeet	10 CSCE	19 Male	6 Kolkata	9.5	25000

9 rows selected.

g) Delete all the students from the table who belong to BBSR.

Delete from STUDENT where City = 'BBSR';

1 row deleted.

SQL> SELECT * FROM STUDENT;

NAME	ROLL DEPT	AGE GENDER	SEM CITY	CGPA	BAL_FEE
Anand	2 CSE	20 Male	4 New Delhi	8.5	20000
Anjali	3 CSE	17 Female	4 Kolkata	8.46	50000
Priya	4 CSE	19 Female	4 Pune	9.5	10000
Aakash	6 CSCE	18 Male	6 Varanasi	7.5	30000
Naman	7 CSCE	20 Male	6 Jaipur	9.3	50000
Shreya	8 CSCE	18 Female	6 Allahabad	9.3	20000
Shalu	9 IT	19 Female	8 Kolkata	8.3	5000
Silajeet	10 CSCE	19 Male	6 Kolkata	9.5	25000

8 rows selected.

h) Delete the student from the table whose CGPA is less than 8.

Delete from STUDENT where CGPA < 8;

1 row deleted.

SQL> SELECT * FROM STUDENT;

NAME	ROLL DEPT	AGE GENDER	SEM CITY	CGPA	BAL_FEE
Anand	2 CSE	20 Male	4 New Delhi	8.5	20000
Anjali	3 CSE	17 Female	4 Kolkata	8.46	50000
Priya	4 CSE	19 Female	4 Pune	9.5	10000
Naman	7 CSCE	20 Male	6 Jaipur	9.3	50000
Shreya	8 CSCE	18 Female	6 Allahabad	9.3	20000
Shalu	9 IT	19 Female	8 Kolkata	8.3	5000
Silajeet	10 CSCE	19 Male	6 Kolkata	9.5	25000

7 rows selected.

SQL> spool out;

Question - 2

Create a table PRODUCT with attributes like p_id, p-name, color, mfg_yr, exp_yr and price. Enter at least 8 records in the table as shown and perform the following operations.

P_ID	P_NAME	COLOR	MFG_YR	EXP_YR	PRICE	LOP
100	mobile	black	2015	2020	8499	5
101	earphone	white	2017	2020	499.99	3
102	speaker	red	2018	2022	899.49	4
103	keyboard	black	2013	2020	1299.49	7
104	monitor	blue	2017	2030	9100.49	13
105	hdd	black	2014	2030	3500.49	16
106	mouse	white	2016	2022	500.49	6
107	pen drive	green	2013	2022	300.49	9

Creating the table.

Create Table PRODUCT (p_id Number(3), P-name Varchar(15), color Varchar(10), Mgf-yr Number(4), Exp-yr Number(4), Price Number (6,2));

Inserting values into table.

Insert into PRODUCT(P_id, P-name, color, Mgf-yr, Exp-yr, Price) values (100,'mobile','black', 2015, 2020, 8400);

Insert into PRODUCT(P_id, P-name, color, Mgf-yr, Exp-yr, Price) values (101,'earphone','white', 2017, 2020, 499.99);

Insert into PRODUCT(P_id, P-name, color, Mgf-yr, Exp-yr, Price) values (102,'speaker','red', 2018, 2022, 899.49);

Insert into Product (P_id, P-name, colour, Mgf-Yr, Exp-Yr, Price) values (103, 'Keyboard', 'black', 2013, 2020, 1299.49);

Insert into PRODUCT (P_id, P-name, color, Mgf-Yr, Exp-yr, Price) values (104, 'Monitor', 'blue', 2017, 2030, 9100.49);

Insert into PRODUCT (P_id, P-name, color, Mgf-Yr, Exp-yr, Price) values (105, 'hdd', 'black', 2014, 2030, 3500.49);

Insert into PRODUCT (P_id, P-name, color, Mgf-yr, Exp-yr, Price) values (106, 'Mouse', 'white', 2016, 2022, 500.49);

Insert into PRODUCT (P_id, P-name, color, Mgf-yr, Exp-yr, Price) values (107, 'Pen drive', 'green', 2013, 2022, 300.49);

a) Display the content of the table.

Select * from PRODUCT;

P_ID	P_NAME	COLOUR	MFG_YR	EXP_YR	PRICE
100	mobile	black	2015	2020	8400
101	earphone	white	2017	2020	499.99
102	speaker	red	2018	2022	899.99
103	keyboard	black	2013	2020	1299.49
104	monitor	blue	2017	2030	9100.49
105	hdd	black	2014	2030	3500.49
106	mouse	white	2016	2022	500.49
107	pen drive	green	2013	2022	300.49

8 rows selected.

b) Display P-id, color of the product.

Select P-id, color from product;

P_ID	COLOUR
100	black
101	white
102	red
103	black
104	blue
105	black
106	white
107	green

8 rows selected.

c) Display the details of red colored product.

Select * from PRODUCT where colour='red';

P_ID	P_NAME	COLOUR	MFG_YR	EXP_YR	PRICE
102	speaker	red	2018	2022	899.99

D) Display the p-name and price for the products that were manufactured in year 2017

Select p-name, price from product where mfg_yr = 2017;

P_NAME	PRICE
earphone	499.99
monitor	9100.49

E) Add a new column lifetime to the product table and update the values for the newly added column in the table.

alter table product add lopnumber(2);

Updating values into table

Update product set LOP=5 where P_ID=100;
 Update product set LOP=3 where P_ID =101;
 Update product set LOP=4 where P_ID=102;
 Update product set LOP=7 where P_ID=103;
 update product set LOP=13 where P_ID =104;
 update product set LOP=16 where P_ID=105;
 update product set LOP=6 where P_ID=106;
 update Product set LOP=9 where P_ID=107;

Select * from PRODUCT;

P_ID	P_NAME	COLOUR	MFG_YR	EXP_YR	PRICE	LOP
100	mobile	black	2015	2020	8400	5
101	earphone	white	2017	2020	499.99	3
102	speaker	red	2018	2022	899.99	4
103	keyboard	black	2013	2020	1299.49	7
104	monitor	blue	2017	2030	9100.49	13
105	hdd	black	2014	2030	3500.49	16
106	mouse	white	2016	2022	500.49	6
107	pen drive	green	2013	2022	300.49	9

8 rows selected.

F) Display the details of the product whose lifetime is less than 10 yrs.

Select * from product where LOP < 10;

P_ID	P_NAME	COLOUR	MFG_YR	EXP_YR	PRICE	LOP
100	mobile	black	2015	2020	8400	5
101	earphone	white	2017	2020	499.99	3
102	speaker	red	2018	2022	899.99	4
103	keyboard	black	2013	2020	1299.49	7
106	mouse	white	2016	2022	500.49	6
107	pen drive	green	2013	2022	300.49	9

6 rows selected.

Q) Display the P_id, P_name and color of two products that has LOP of 4 yrs.

Select P_id, P_name, colour from product where LOP = 4;

P_ID	P_NAME	COLOUR
102	Speaker	red.

H) Delete the product details for the product from the table that was manufactured in the 2015

delete from product where mfg_yr = 2015;
select * from product;

1 row deleted.

SQL> select * from PRODUCT;

P_ID	P_NAME	COLOUR	MFG_YR	EXP_YR	PRICE	LOP
101	earphone	white	2017	2020	499.99	3
102	speaker	red	2018	2022	899.99	4
103	keyboard	black	2013	2020	1299.49	7
104	monitor	blue	2017	2030	9100.49	13
105	hdd	black	2014	2030	3500.49	16
106	mouse	white	2016	2022	500.49	6
107	pen drive	green	2013	2022	300.49	9

7 rows selected.

I) Delete the column color from the table.

`alter table product drop column COLOUR;`

`select * from product;`

P_ID	P_NAME	MFG_YR	EXP_YR	PRICE	LOP
101	earphone	2017	2020	499.99	3
102	speaker	2018	2022	899.99	4
103	keyboard	2013	2020	1299.49	7
104	monitor	2017	2030	9100.49	13
105	hdd	2014	2030	3500.49	16
106	mouse	2016	2022	500.49	6
107	pen drive	2013	2022	300.49	9

7 rows selected.

J) Delete all the records from the table without deleting the table.

`truncate table product;`

Table truncated.

K) Delete the table from the database.

`Drop Product;`

Question - 3

- Create a table EMPLOYEE with the details like Emp-ID, Emp-name, Dept, Gender, Age & city. Enter at least 10 records as shown in fig. & perform the following operations.

EMP_ID	EMP_NAME	DEPT	GENDER	AGE	CITY	SALARY
1	Kunal	CSE	Male	34	BBSR	40000
2	Anand	CSE	Male	32	BBSR	30000
3	Reema	IT	Female	27	BBSR	50000
4	Swati	IT	Female	26	Banglore	25000
5	Shiv	Bank	Male	30	Rourkela	30000
6	Abhishek	HR	Male	29	Delhi	20000
7	Rahul	CSCE	Male	25	CTC	45000
8	Pragyan	CSSE	Female	30	Puri	35000
9	Kaushal	IT	Male	32	Kolkata	55000
10	Seenu	Accounts	Female	33	Noida	25000

Creating the table .

Create Table EMPLOYEE (emp-id Number(2), Emp-Name Varchar(10), Dept Varchar(6), Age Number(2), city Varchar(10));

Inserting values into table

Insert into Employee(emp-id, emp-name, Dept, gender, Age, city) Values (1, 'Kunal', 'CSE', 'Male', 34, 'BBSR');

Insert into Employee(emp-id, emp-name, Dept, gender, Age, city) Values (2, 'Anand', 'CSE', 'Male', 32, 'BBSR');

Insert into Employee(emp-id, emp-name, Dept, gender, Age, city) Values (3, 'Reema', 'IT', 'Female', 27, 'BBSR');

Insert into Employee(Emp_id, Emp_name, dept, gender, Age, city) values (4, 'Swati', 'IT', 'Female', 26, 'Banglore');

Insert into Employee(Emp_id, Emp_name, dept, gender, Age, city) values (5, 'Shiv', 'Bank', 'Male', 30, 'Rourkela');

Insert into Employee(Emp_id, Emp_name, dept, gender, Age, city) values (6, 'Abhishek', 'HR', 'Male', 39, 'Delhi');

Insert into Employee(Emp_id, Emp_name, dept, gender, Age, city) values (7, 'Rahul', 'CSCE', 'Male', 25, 'CTC');

Insert into Employee(Emp_id, Emp_name, dept, gender, Age, city) values (8, 'Pragyan', 'CSE', 'Female', 30, 'Puri');

Insert into Employee(Emp_id, Emp_name, dept, gender, Age, city) values (9, 'Kaushal', 'IT', 'Male', 32, 'Kolkata');

Insert into Employee(Emp_id, Emp_name, dept, gender, Age, city) values (10, 'Seenu', 'Accounts', 'Female', 33, 'Noida');

a) Display the content of the table .

Select * from employee;

EMP_ID	EMP_NAME	DEPT	GENDER	AGE	CITY
1	Kunal	CSE	Male	34	BBSR
2	Anand	CSE	Male	32	BBSR
3	Reema	IT	Female	27	BBSR
4	Swati	IT	Female	26	Banglore
5	Shiv	Bank	Male	30	Rourkela
6	Abhishek	HR	Male	39	Delhi
7	Rahul	CSCE	Male	25	CTC
8	Pragyan	CSE	Female	30	Puri
9	Kaushal	IT	Male	32	Kolkata
10	Seenu	Accounts	Female	33	Noida

10 rows selected.

b) Display the details of the employees from BBSR.

Select * from employee where city = 'BBSR';

Emp-ID	Emp-Name	Dept	Gender	Age	City
1	Kunal	CSE	Male	34	BBSR
2	Anand	CSE	Male	32	BBSR
3	Reema	IT	Female	27	BBSR.

c) Display the Emp-ID, Emp-name & Dept for female employees whose age is less than 30.

Select Emp-ID, Emp-Name, Dept from Employee
Where Gender = 'Female' And Age < 30;

Emp-ID	Emp-Name	Dept
3	Reema	IT
4	Swati	IT.

d) Display the name, dept and city for all male employees.

Select Emp-name, Dept, city from Employee
where Gender = 'Male';

EMP_NAME	DEPT	CITY
Kunal	CSE	BBSR
Anand	CSE	BBSR
Shiv	Bank	Rourkela
Abhishek	HR	Delhi
Rahul	CSCE	CTC
Kaushal	IT	Kolkata

6 rows selected.

f) Display the details of the employee who belongs to either BBSR or CTC.

Select * from employee where CITY = 'BBSR' OR CITY = 'CTC';

EMP_ID	EMP_NAME	DEPT	GENDER	AGE	CITY
1	Kunal	CSE	Male	34	BBSR
2	Anand	CSE	Male	32	BBSR
3	Reema	IT	Female	27	BBSR
7	Rahul	CSCE	Male	25	CTC

g) Add a new column salary and update the values for the newly added column.

Adding new column to table

Alter table Employee Add SALARY Number(6);

Updating values into table.

update Employee set salary = 40000 where Emp-ID=1;
 update Employee set salary = 30000 where Emp-ID=2;
 update Employee set salary = 50000 where Emp-ID=3;
 update Employee set salary = 25000 where Emp-ID=4;
 update Employee set salary = 30000 where Emp-ID=5;
 update Employee set salary = 20000 where Emp-ID=6;
 update Employee set salary = 45000 where Emp-ID=7;
 update Employee set salary = 35000 where Emp-ID=8;
 update Employee set salary = 55000 where Emp-ID=9;
 update Employee set salary = 25000 where Emp-ID=10.

Select * from employee;

EMP_ID	EMP_NAME	DEPT	GENDER	AGE	CITY	SALARY
1	Kunal	CSE	Male	34	BBSR	40000
2	Anand	CSE	Male	32	BBSR	30000
3	Reema	IT	Female	27	BBSR	50000
4	Swati	IT	Female	26	Banglore	25000
5	Shiv	Bank	Male	30	Rourkela	30000
6	Abhishek	HR	Male	39	Delhi	20000
7	Rahul	CSCE	Male	25	CTC	45000
8	Pragyan	CSsE	Female	30	Puri	35000
9	Kaushal	IT	Male	32	Kolkata	55000
10	Seenu	Accounts	Female	33	Noida	25000

10 rows selected.

i) Display the details of the employee in the descending order of the salary.

Select * From Employee order by Salary Desc;

EMP_ID	EMP_NAME	DEPT	GENDER	AGE	CITY	SALARY
9	Kaushal	IT	Male	32	Kolkata	55000
3	Reema	IT	Female	27	BBSR	50000
7	Rahul	CSCE	Male	25	CTC	45000
1	Kunal	CSE	Male	34	BBSR	40000
8	Pragyan	CSsE	Female	30	Puri	35000
2	Anand	CSE	Male	32	BBSR	30000
5	Shiv	Bank	Male	30	Rourkela	30000
10	Seenu	Accounts	Female	33	Noida	25000
4	Swati	IT	Female	26	Banglore	25000
6	Abhishek	HR	Male	39	Delhi	20000

10 rows selected.

i) Display the Emp-ID , Emp-name , and Dept for the female employee whose salary is greater than or equal to 30000.

select Emp_id,emp_name,dept from Employee where gender='Female' AND salary >= 30000;

Emp-ID	Emp-Name	Dept
3	Reema	IT
8	Pragyan	CSSE

i) Delete the details of the male employee who belongs to BBSR city .

Delete from Employee where gender = 'Male' & CITY=BBSR;

EMP_ID	EMP_NAME	DEPT	GENDER	AGE	CITY	SALARY
3	Reema	IT	Female	27	BBSR	50000
4	Swati	IT	Female	26	Banglore	25000
5	Shiv	Bank	Male	30	Rourkela	30000
6	Abhishek	HR	Male	39	Delhi	20000
7	Rahul	CSCE	Male	25	CTC	45000
8	Pragyan	CSsE	Female	30	Puri	35000
9	Kaushal	IT	Male	32	Kolkata	55000
10	Seenu	Accounts	Female	33	Noida	25000

8 rows selected.

SQL> SPOOL OUT;

Constraints in SQL

Question-1

Create table ITEM-MASTER with the attributes item-no, Name , Quantity-available , category , Unit-Measure , Reorder-level , Reorder-quantity and Rate . The columns Reorder-level , Reorder-quantity and Rate should not be NULL . The Name must be unique , the column Item-no can neither be left NULL nor its value can be repeated . The column Quantity-available should have a default value 100 , if the value is not given . The column category must belong to anyone of A or B or C . That can be done with the help of check constraint . Enter at least 10 entries to the table and perform the following operation .

ITEM_NAME	QTY_AVL	C	UNIT	RDR_LVL	RDR_QTY	RATE
I01 Mobile	5000	A	pcs	500	2000	8999
I02 Headphone	1000	A	pcs	300	1000	999
I03 Charger	700	A	pcs	400	1000	399
I04 Monitor	1000	B	pcs	100	2000	10000
I05 Mouse	100	B	pcs	100	200	500
I06 Keyboard	100	B	pcs	50	200	800
I07 HDD	100	B	pcs	100	500	3000
I08 Fridge	100	C	pcs	100	500	15000
I09 TV Set	1000	C	pcs	500	5000	30000
I10 AC	1000	C	pcs	500	3000	40000

Creating the table.

```
create table item-master (item-no varchar(5) primary key, name varchar(20) unique, qty-avl number(5) default 100, category char(1) check (category in('A','B','C','D')), unit char(4), sds-lvl number(3) not null, sds-qty number(5) not null, rate number(5) not null);
```

Inserting values into tables.

```
insert into item-master (item-no, name, qty-avl, category, unit, sds-lvl, sds-qty, rate) values ('I01', 'Mobile', 500, 'A'), ('PCS', 500, 200, 8999);
```

```
insert into item-master (item-no, name, qty-avl, category, unit, sds-lvl, sds-qty, rate) values ('I02', 'Headphone', 1000, 'A'), ('PCS', 300, 100, 999);
```

```
insert into item-master (item-no, name, qty-avl, category, unit, sds-lvl, sds-qty, rate) values ('I03', 'Charger', 700, 'A'), ('PCS', 400, 1000, 399);
```

```
insert into item-master (item-no, name, qty-avl, category, unit, sds-lvl, sds-qty, rate) values ('I04', 'Monitor', 1000, 'B'), ('PCS', 100, 2000, 10000);
```

```
insert into item-master (item-no, name, qty-avl, category, unit, sds-lvl, sds-qty, rate) values ('I05', 'Mouse', 100, 'B'), ('PCS', 100, 200, 500);
```

```
insert into item-master (item-no, name, qty-avl, category, unit, sds-lvl, sds-qty, rate) values ('I06', 'Keyboard', 100, 'B'), ('PCS', 50, 200, 800);
```

```
insert into item-master (item-no, name, qty-avl, category, unit, sds-lvl, sds-qty, rate) values ('I07', 'HDD', 100, 'B'), ('PCS', 100, 1500, 3000);
```

```
insert into item-master(item-no,qty-avl,category,unit,
rdr-lvl,rdr-qty,rate) values ('I08','Fridge',100
'C','Pcs',100,500,15000);
```

```
insert into item-master(item-no,qty-avl,category,unit,
rdr-lvl,rdr-qty,rate) values ('I09','TV Set',1000,
'C','Pcs',500,5000,30000);
```

```
insert into item-master(item-no,qty-avl,category,unit,
rdr-lvl,rdr-qty,rate) values ('I10','AC',1000,'C',
'Pcs',500,3000,40000);
```

Display the item-master Table

Select * from item-master;

ITEM_NAME	QTY_AVL	C	UNIT	RDR_LVL	RDR_QTY	RATE
I01 Mobile	5000	A	pcs	500	2000	8999
I02 Headphone	1000	A	pcs	300	1000	999
I03 Charger	700	A	pcs	400	1000	399
I04 Monitor	1000	B	pcs	100	2000	10000
I05 Mouse	100	B	pcs	100	200	500
I06 Keyboard	100	B	pcs	50	200	800
I07 HDD	100	B	pcs	100	500	3000
I08 Fridge	100	C	pcs	100	500	15000
I09 TV Set	1000	C	pcs	500	5000	30000
I10 AC	1000	C	pcs	500	3000	40000

10 rows selected.

- a) Display the item-no, name & rate for all item in the table .

Select item-no, name , rate from item-master

ITEM_NAME	RATE
I01 Mobile	8999
I02 Headphone	999
I03 Charger	399
I04 Monitor	10000
I05 Mouse	500
I06 Keyboard	800
I07 HDD	3000
I08 Fridge	15000
I09 TV Set	30000
I10 AC	40000

b) Display the details of category A and rate less than 1000.

Select * from item-master where (category='A' AND rate <= 1000);

Item_No	Item_Name	QTY_AVL	C	UNIT	RDR_LVL	RDR_QTY	RATE
I02	Headphone	1000	A	PCS	300	1000	999
I03	Charger	700	A	PCS	400	1000	399

c) Display the product in descending order of quantity available for item falling under category 'B'.

Select * from item-master where category='B'
order by QTY_AVL Desc;

SQL> SELECT * FROM ITEM_MASTER WHERE CATEGORY='B' ORDER BY QTY_AVL DESC;

ITEM_NAME	QTY_AVL	C	UNIT	RDR_LVL	RDR_QTY	RATE	
I04	Monitor	1000	B	PCS	100	2000	10000
I06	Keyboard	100	B	PCS	50	200	800
I07	HDD	100	B	PCS	100	500	3000
I05	Mouse	100	B	PCS	100	200	500

d) Display the details of the item starting with "M" in descending order of their re-order level.

Select * from item-master where name like 'M%'
order by rdo_lvl desc;

Item_No	Item_Name	QTY_AVL	C	UNIT	RDR_LVL	RDR_QTY	RATE
I01	Mobile	5000	A	PCS	500	2000	8999
I05	Mouse	100	B	PCS	100	200	500
I04	Monitor	1000	B	PCS	100	2000	10000

e) Display the item in the ascending order of their names.

Select * from item-master order by name asc;

ITEM_NAME	QTY_AVL	C UNIT	RDR_LVL	RDR_QTY	RATE
I10 AC	1000	C pcs	500	3000	40000
I03 Charger	700	A pcs	400	1000	399
I08 Fridge	100	C pcs	100	500	15000
I07 HDD	100	B pcs	100	500	3000
I02 Headphone	1000	A pcs	300	1000	999
I06 Keyboard	100	B pcs	50	200	800
I01 Mobile	5000	A pcs	500	2000	8999
I04 Monitor	1000	B pcs	100	2000	10000
I05 Mouse	100	B pcs	100	200	500
I09 TV Set	1000	C pcs	500	5000	30000

10 rows selected.

Question-2

Create a table CUSTOMER with the details like unique cust-ID that can not be left blank, unique cust-name, gender, Age and city. The column "gender" must be entered for each record in the table and it must be either "M", "F" or "T". A customer must be of age more than equal to 18 yrs & less than 80 yrs. Enter at least 10 records to the table and perform the following operations.

Creating table Customer

```
create table customer (cust_id Number(4) Primary Key, cust_name Varchar(20) unique, gender Char(1) check (gender in ('M', 'F', 'T')), age number(2) check (age >= 18) AND (age < 80), city Varchar(20))
```

Inserting values into table.

```
insert into customer (cust_id, cust_name, gender, age, city) values (1, 'Aditya', 'M', 18, 'Vizag');
```

```
insert into customer (cust_id, cust_name, gender, age, city) values (2, 'Kunal', 'M', 20, 'BBSR');
```

```
insert into customer (cust_id, cust_name, gender, age, city) values (3, 'Anand', 'M', 22, 'BBSR');
```

```
insert into customer (cust_id, cust_name, gender, age, city) values (4, 'Reema', 'F', 45, 'CTC');
```

```
insert into customer (cust_id, cust_name, gender, age, city) values (5, 'Swati', 'F', 55, 'CTC');
```

insert into customer (cust_id, cust_name, gender, age, city) values (6, 'Shiv', 'M', 28, 'Noida');

insert into customer (cust_id, cust_name, gender, age, city) values (7, 'Abhishek', 'M', 29, 'Kolkata');

insert into customer (cust_id, cust_name, gender, age, city) values (8, 'Rahul', 'M', 60, 'Kanpur');

insert into customer (cust_id, cust_name, gender, age, city) values (9, 'Pragyan', 'M', 55, 'Delhi');

insert into customer (cust_id, cust_name, gender, age, city) values (10, 'Anya', 'F', 25, 'Mumbai');

a) display the content of the table.

Select * from customer;

CUST_ID	CUST_NAME	GENDER	AGE	CITY
1	Aditya	M	18	Vizag
2	kunal	M	20	bbsr
3	Anand	M	22	bbsr
4	Reema	F	45	ctc
5	Swati	F	55	ctc
6	Shiv	M	28	NOIDA
7	Abhishek	M	29	kolkata
8	rahul	M	60	kanpur
9	pragyan	M	55	delhi
10	Anya	F	25	mumbai

10 rows selected.

b) Display the details of the customer from either BBSR or CTC.

Select * from customer where city = 'bbsr' or city = 'ctc';

CUST_ID	CUST_NAME	GENDER	AGE	CITY
2	kunal	M	20	bbsr
3	Anand	M	22	bbsr
4	Reema	F	45	ctc
5	Swati	F	55	ctc

c) Display the cust-ID, cust-name and Age of all the customers.

Select cust-id, cust-name, age from customers;

CUST_ID	CUST_NAME	AGE
1	Aditya	18
2	kunal	20
3	Anand	22
4	Reema	45
5	Swati	55
6	Shiv	28
7	Abhishek	29
8	rahul	60
9	pragyan	55
10	Anya	25

10 rows selected.

d) Add a new column "Purchase_Amount" to the table and it must not be NULL. Update the values for all records in the table.

alter table customer add Purchase_Amount Integer;

Updating the values

update customer set Purchase_Amount = 2200 where cust_id = 1
 update customer set Purchase_Amount = 5200 where cust_id = 2
 update customer set Purchase_Amount = 7300 where cust_id = 3
 update customer set Purchase_Amount = 4300 where cust_id = 4
 update customer set Purchase_Amount = 5300 where cust_id = 5
 update customer set Purchase_Amount = 10000 where cust_id = 6
 update customer set Purchase_Amount = 1300 where cust_id = 7
 update customer set Purchase_Amount = 7800 where cust_id = 8
 update customer set Purchase_Amount = 3800 where cust_id = 9
 update customer set Purchase_Amount = 9700 where cust_id = 10

Select * from customer;

CUST_ID	CUST_NAME	GENDER	AGE	CITY	PURCHASE_AMOUNT
1	Aditya	M	18	Vizag	2200
2	kunal	M	20	bbsr	5200
3	Anand	M	22	bbsr	7300
4	Reema	F	45	ctc	4300
5	Swati	F	55	ctc	5000
6	Shiv	M	28	NOIDA	10000
7	Abhishek	M	29	kolkata	1300
8	rahul	M	60	kanpur	7800
9	pragyan	M	55	delhi	3800
10	Anya	F	25	mumbai	9700

10 rows selected.

e) Display the name and city for all customers made with purchase_amount ≥ 5000 .

Select cust_name, city from customer where gender = 'M' and purchase_Amount ≥ 5000 ;

CUST_NAME	CITY
kunal	bbsr
Anand	bbsr
Shiv	NOIDA
rahul	kanpur

f) Display the details of the customers whose name starts either start with 'A' or end with 'a'.

Select * from customer where cust_name like 'A%' or cust_name like '%a';

CUST_ID	CUST_NAME	GENDER	AGE	CITY	PURCHASE_AMOUNT
1	Aditya	M	18	Vizag	2200
3	Anand	M	22	bbsr	7300
4	Reema	F	45	ctc	4300
7	Abhishek	M	29	kolkata	1300
10	Anya	F	25	mumbai	9700

g) Display the details of the female customer in the descending order of their purchase_amount.

Select * from customer where gender = 'F' order by purchase_amount desc;

<u>Cust_ID</u>	<u>Cust_Name</u>	<u>Gender</u>	<u>Age</u>	<u>City</u>	<u>Purchase_Amount</u>
10	Anya	F	25	Mumbai	9700
5	Swati	F	55	CTC	5000
4	Reema	F	45	CTC	4300

Question-3

Create table EMP with the attributes EMP-ID, Emp-Name, designation, Manager-ID, Hire-Date, Salary, Commission, and Dept-ID. Enter the records in the table as shown below.

7369	SMITH	CLERK	7902	17-DEC-80	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10
7788	SCOTT	ANALYST	7566	09-NOV-81	3000		20
7839	KING	PRESIDENT		17-NOV-81	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	23-SEP-81	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	950		30
7902	FORD	ANALYST	7566	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	1300		10

Creating table EMP

```
create table EMP (emp-id number(5) primary key,
emp-name varchar2(20) not null, designation varchar(18)
not null, manager-id number(6), hire-date date,
salary number(10) not null, comission number(10),
dept-id number(4) default 10);
```

Inserting values into table

```
insert into Emp (emp-id, emp-name, designation, manager-id,
hire-date, salary, dept-id) values (7369, 'SMITH',
'CLERK', 7902, '17-DEC-80', 800, 20);
```

insert into Emp(emp_id, emp_name, designation, manager_id, hire_date, salary, comission, dept_id) values (7499, 'ALLEN', 'SALESMAN', 7698, '20-FEB-81', 1600, 300, 30);

insert into Emp(emp_id, emp_name, designation, manager_id, hire_date, salary, comission, dept_id) values (7521, 'WARD', 'SALESMAN', 7698, '22-FEB-81', 1250, 500, 30);

insert into Emp(emp_id, emp_name, designation, manager_id, hire_date, salary, comission, dept_id) values (7886, 'JONES', 'Manager', 7839, '22-APR-81', 2975, 20);

insert into Emp(emp_id, emp_name, designation, manager_id, hire_date, salary, dept_id) values (7698, 'BLAKE', 'MANAGER', 7839, '28-SEP-81', 1250, 400, 30);

insert into Emp(emp_id, emp_name, designation, manager_id, hire_date, salary, comission, dept_id) values (7654, 'MARTIN', 'Salesman', 7698, '28-SEP-81', 1250, 400, 30);

insert into Emp(emp_id, emp_name, designation, manager_id, hire_date, salary, dept_id) values (7782, 'CLARK', 'MANAGER', 7839, '09-JUN-81', 2450, 10);

insert into Emp(emp_id, emp_name, designation, manager_id, hire_date, salary, dept_id) values (7788, 'SCOTT', 'ANALYST', 7566, '09-Nov-81', 3000, 20);

insert into Emp(emp_id, emp_name, designation, manager_id, hire_date, salary, commission, dept_id) values (7844, 'TURNER', 'SALESMAN', 7698, '08-SEP-81', 1500, 0, 30);

insert into Emp(emp_id, emp_name, designation, manager_id, hire_date, salary, dept_id) values (7876, 'ADAMS', 'CLERK', 7788, '23-SEP-81', 1100, 20);

insert into EMP (emp_id, emp_name, designation, manager_id, hire_date, salary, dept_id) values
 (7900, 'JAMES', 'CLERK', 7698, '03-DEC-81', 950, 30);

insert into Emp (emp_id, emp_name, designation, manager_id, hire_date, salary, dept_id) values (7902, 'FORD', 'ANALYST', 7902, '03-DEC-81', 3000, 20);

insert into Emp (emp_id, emp_name, designation, manager_id, hire_date, salary, dept_id) values (7934, 'MILLER', 'CLERK', 7782, '23-JAN-82', 1300, 10);

A) Display the content of the table.

Select * from emp;

EMP_ID	EMP_NAME	DESIGNATION	MANAGER_ID	HIRE_DATE	SALARY	COMISSION	DEPT_ID
7369	SMITH	CLERK	7902	17-DEC-80	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	22-APR-81	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		20
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10
7788	SCOTT	ANALYST	7566	09-NOV-81	3000		20
7839	KING	PRESIDENT		17-NOV-81	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	23-SEP-81	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	950		30
7902	FORD	ANALYST	7902	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	1300		10

14 rows selected.

B) Display the details of the employee who joined on or after 01-May-81.

Select * from Emp where hire_date >= '01-MAY-81';

EMP_ID	EMP_NAME	DESIGNATION	MANAGER_ID	HIRE_DATE	SALARY	COMISSION	DEPT_ID
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		20
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10
7788	SCOTT	ANALYST	7566	09-NOV-81	3000		20
7839	KING	PRESIDENT		17-NOV-81	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	23-SEP-81	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	950		30
7902	FORD	ANALYST	7902	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	1300		10

10 rows selected.

c) Display the Emp-ID, Emp-Name and Designation of the employees who work under the Manager 7698.

Select Emp-id , Emp-name , Designation from Emp where Manager_id = 7698;

SQL> SELECT EMP_ID,EMP_NAME,DESIGNATION FROM EMP WHERE MANAGER_ID=7698 ;

EMP_ID	EMP_NAME	DESIGNATION
7499	ALLEN	SALESMAN
7521	WARD	SALESMAN
7654	MARTIN	SALESMAN
7844	TURNER	SALESMAN
7900	JAMES	CLERK

D) Display the details of the employee who either work for manager 7839 or work in dept 20

Select Emp-ID, Emp-Name, Designation from Emp where Manager_ID = 7839 OR Dept_id = 20;

EMP_ID	EMP_NAME	DESIGNATION	MANAGER_ID	HIRE_DATE	SALARY	COMISSION	DEPT_ID
7369	SMITH	CLERK	7902	17-DEC-80	800		20
7566	JONES	MANAGER	7839	22-APR-81	2975		20
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		20
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10
7788	SCOTT	ANALYST	7566	09-NOV-81	3000		20
7876	ADAMS	CLERK	7788	23-SEP-81	1100		20
7902	FORD	ANALYST	7902	03-DEC-81	3000		20

7 rows selected.

E) Display the Emp-name , Salary and Manager details for the employees who joined in dept 30 after 20 - Feb-81.

Select Emp-Name , Salary , Manager_id from Emp
Where Dept_id = 30 AND hire_date > '20-Feb-81';

EMP_NAME	SALARY	MANAGER_ID
WARD	1250	7698
MARTIN	1250	7698
TURNER	1500	7698
JAMES	950	7698

F) Display the Emp-ID and Emp-name in the ascending order of their salary .

Select Emp-id , Emp-Name . From Emp Order By Salary Asc ;

EMP_ID	EMP_NAME
7369	SMITH
7900	JAMES
7876	ADAMS
7521	WARD
7654	MARTIN
7934	MILLER
7844	TURNER
7499	ALLEN
7782	CLARK
7698	BLAKE
7566	JONES
7788	SCOTT
7902	FORD
7839	KING

14 rows selected.

SQL> spool off;

Date :- 7-02-2022 Aditya Ranjjan Pradhan 20051620

Aggregate Function in SQL

Question -1

Create a table STUDENT-2 with the attributes like Student-Name, Roll, Department, Age, Gender and Semester. The column student-name must not be left blank, Roll number must be entered and it must be unique. The age of the student must be b/w 18 & 28. For gender, one can enter either Male or Female. Add a new column CGPA to the table. The column CGPA must be entered for all the students. Enter at least 10 entries into the table as shown below & perform following operation.

NAME	ROLL	DEPT	AGE GENDER	SEM	CGPA
Shiv	I01	IT	28 Male	7	8.5
Kaushal	I02	IT	27 Male	7	9.5
Pragyan	I03	IT	25 Female	7	7.5
Seenu	C01	CSE	25 Female	8	9.5
Kunal	C02	CSE	28 Male	8	9.35
Lokesh	C03	CSE	24 Male	8	8
Amit	E01	ETC	27 Male	6	9.2
Amrita	E02	ETC	22 Female	6	7.45
Tarkeshwar	E03	EEE	26 Male	6	8.45
Neha	C04	CSE	23 Female	6	8.45

Creating the table

create table student-2 (student-name varchar(10) not null, roll char(3) unique not null, dept varchar(3), age number(2) check (age >= 18 and age <= 28), gender char(1) check (gender in ('M', 'F')) , sem number(2)); add (GPA number(3, 2));

Inserting values into table

insert into student_2 (student_name, roll, dept, age, gender, sem, cgpa) values ('Shiv', 'I01', 'IT', 28, 'M', 7, 8.5);

insert into student_2 (student_name, roll, dept, age, gender, sem, cgpa) values ('Kaushal', 'I02', 'IT', 27, 'M', 7, 9.5);

insert into student_2 (student_name, roll, dept, age, gender, sem, cgpa) values ('Pragya', 'I03', 'IT', 25, 'F', 7, 7.5);

insert into student_2 (student_name, roll, dept, age, gender, sem, cgpa) values ('Seenu', 'COI', 'CSE', 28, 'I', 8, 9.5);

insert into student_2 (student_name, roll, dept, age, gender, sem, cgpa) values ('Kunal', 'CO2', 'CSE', 28, 'M', 8, 9.35);

insert into student_2 (student_name, roll, dept, age, gender, sem, cgpa) values ('Lokesh', 'CO3', 'CSE', 24, 'M', 8.8);

insert into student_2 (student_name, roll, dept, age, gender, sem, cgpa) value ('Amit', 'EO1', 'ETC', 27, 'M', 6, 9.2);

insert into student_2 (student_name, roll, dept, age, gender, sem, cgpa) values ('Talkeshwar', 'EO3', 'EEE', 26, 'M', 6, 8.45);

insert into student_2 (student_name, roll, dept, age, gender, sem, cgpa) values ('Neha', 'CO4', 'CSE', 23, 'F', 6, 8.45);

a) Display the content of the table -

Select * from student_2;

STUDENT_NA	ROLL	DEPT	AGE	GENDER	SEM	CGPA
Shiv	I01	IT	28	M	7	8.5
Kaushal	I02	IT	27	M	7	9.5
Pragyan	I03	IT	25	F	7	7.5
Seenu	C01	CSE	25	F	8	9.5
Kunal	C02	CSE	28	M	8	9.35
Lokesh	C03	CSE	24	M	8	8
Amit	E01	ETC	27	M	6	9.2
Amrita	E02	ETC	22	F	6	7.45
Tarkeshwar	E03	EEE	26	M	6	8.45
Neha	C04	CSE	23	F	6	8.45

10 rows selected.

b) Display the roll, age and gender for the student from either CSE or IT department.

select roll,age,gender from student_2
where dept='CSE' or dept='IT';

SQL> select roll,age,gender from student_2 where dept='CSE' or dept='IT';

ROLL	AGE	GENDER
I01	28	M
I02	27	M
I03	25	F
C01	25	F
C02	28	M
C03	24	M
C04	23	F

7 rows selected.

c) Display the Student-Name , roll and department for the students whose CGPA is between 7.5 and 9.5

select student-name, roll, dept from student_2
where cgpa >= 7.5 and cgpa <= 9.5;

STUDENT_NA	ROLL	DEPT
Shiv	I01	IT
Kaushal	I02	IT
Pragyan	I03	IT
Seenu	C01	CSE
Kunal	C02	CSE
Lokesh	C03	CSE
Amit	E01	ETC
Tarkeshwar	E03	EEE
Neha	C04	CSE

 Shiv I01 IT
 Kaushal I02 IT
 Pragyan I03 IT
 Seenu C01 CSE
 Kunal C02 CSE
 Lokesh C03 CSE
 Amit E01 ETC
 Tarkeshwar E03 EEE
 Neha C04 CSE

9 rows selected.

- d) Display the student details in the descending order of their CGPA -

Select * from student-2 order by cgpa desc;

STUDENT_NA	ROLL	DEPT	AGE	GENDER	SEM	CGPA
Seenu	C01	CSE	25	F	8	9.5
Kaushal	I02	IT	27	M	7	9.5
Kunal	C02	CSE	28	M	8	9.35
Amit	E01	ETC	27	M	6	9.2
Shiv	I01	IT	28	M	7	8.5
Neha	C04	CSE	23	F	6	8.45
Tarkeshwar	E03	EEE	26	M	6	8.45
Lokesh	C03	CSE	24	M	8	8
Pragyan	I03	IT	25	F	7	7.5
Amrita	E02	ETC	22	F	6	7.45

10 rows selected.

- e) Display the Student-Name in ascending order.

Select student-name from student-2 order by student-name asc;

STUDENT_NA
Amit
Amrita
Kaushal
Kunal
Lokesh
Neha
Pragyan
Seenu
Shiv
Tarkeshwar

10 rows selected.

f) Display the total age of the students:

Select sum(age) from student-2;

SUM (AGE)

255

g) Display the average CGPA of the female student who belong to either CSE or IT.

Select avg(cgpa) from student-2 where gender = 'F' and (dept = 'CSE' or dept = 'IT');

Avg (CGPA)

8.4833333

h) Display the maximum and minimum CGPA of the student table.

Select max(cgpa), min(cgpa) from student-2;

Max (CGPA)

9.5

Min (CGPA)

7.45

i) Display the details of the youngest and oldest student of the class.

Select min(age), max(age) from student-2;

Min (Age)

22

Max (Age)

28

Question-2

Create a table Employee-2 with the details like Emp-ID, Emp-name, Dept, Gender, Age and city. Enter at least 10 records to the table as shown below and perform the following operations.

EMP_ID	EMP_NAME	DEPT	GENDER	AGE	CITY	SALARY
1	Kunal	CSE	Male	34	BBSR	40000
2	Anand	CSE	Male	32	BBSR	30000
3	Reema	IT	Female	27	BBSR	50000
4	Swati	IT	Female	26	Banglore	25000
5	Shiv	Bank	Male	30	Rourkela	30000
6	Abhishek	HR	Male	29	Delhi	20000
7	Rahul	CSCE	Male	25	CTC	45000
8	Pragyan	CSSE	Female	30	Puri	35000
9	Kaushal	IT	Male	32	Kolkata	55000
10	Seenu	Accounts	Female	33	Noida	25000

Creating the table

Create Table Employee-2 (Emp-id number(2), Emp-name varchar(16), Dept varchar(2), Gender varchar(2), Age number(2), City varchar(10)).

Inserting values into table

Insert into Employee-2(Emp-id, Emp-Name, Dept, Gender, Age, City) values (1, 'Kunal', 'CSE', 'Male', 34, 'BBSR');

Insert into Employee-2(Emp-id, Emp-Name, Dept, Gender, Age, City) values (2, 'Anand', 'CSE', 'Male', 32, 'BBSR');

Insert into Employee-2(Emp-id, Emp-Name, Dept, gender, Age, city) values (3, 'Reema', 'IT', 'Female', 27, 'BBSR');

Insert into Employee-2(Emp-id, Emp-Name, Dept, gender, Age, city) values (4, 'Swati', 'IT', 'Female', 26, 'Banglore');

Insert into Employee-2(Emp-id, Emp-Name, Dept, gender, Age, city) value (5, 'Shiv', 'Bam K', 'Male', 30, 'Rowkela');

Insert into Employee-2(Emp-id, Emp-Name, Dept, gender, Age, city) values (6, 'Abshick', 'HR', 'Male', 29, 'Delhi');

Insert into Employee-2(Emp-id, Emp-Name, Dept, gender, Age, city) values (7, 'Rahul', 'CSCE', 'Male', 25, 'CTC');

Insert into Employee-2(Emp-id, Emp-Name, Dept, gender, Age, city) values (8, 'Pragyan', 'CSSE', 'Female', 30, 'Pune');

Insert into Employee-2(Emp-id, Emp-Name, Dept, gender, Age, city) values (9, 'Kaushal', 'IT', 'Male', 32, 'Kolkata');

Insert into Employee-2(Emp-id, Emp-Name, Dept, gender, Age, city) values (10, 'Seenu', 'Accounts', 'Female', 33, 'Noida');

a) Display the content of the table.

Select * from Employee-2;

EMP_ID	EMP_NAME	DEPT	GENDER	AGE	CITY
1	Kunal	CSE	Male	34	BBSR
2	Anand	CSE	Male	32	BBSR
3	Reema	IT	Female	27	BBSR
4	Swati	IT	Female	26	Banglore
5	Shiv	Bank	Male	30	Rourkela
6	Abhishek	HR	Male	29	Delhi
7	Rahul	CSCE	Male	25	CTC
8	Pragyan	CSSE	Female	30	Puri
9	Kaushal	IT	Male	32	Kolkata
10	Seenu	Accounts	Female	33	Noida

10 rows selected.

b) Display the name & city for all the male employee and who belong to 'Delhi'

Select .Emp_name, city from Employee-2
where .gender = 'Male' AND city = 'Delhi';

Emp_Name. CITY.
Abhishek Delhi

c) Display the detail of the employee whose name starts with 'A'

Select * from Employee-2 where Emp_name like 'A%';

Emp-ID	Emp-Name	Dept	Gender	Age	City
2	Anand	CSE	Male	32	BBSR
6	Abhishek	HR	Male	29	Delhi

d) Add a new column salary and update the values.
for the newly added column.

Alter table Employee -2 Add salary Number(6);

update Employee -2 set salary = 40000 where Emp_id=1;
 update Employee -2 set salary = 30000 where Emp_id = 2;
 update Employee -2 set salary = 5000 where emp_id = 3;
 update Employee -2 set salary = 25000 where Emp_id = 4;
 update Employee -2 set salary = 30000 where Emp_id = 5;
 update Employee -2 set salary = 20000 where Emp_id = 6;
 update Employee -2 set salary = 45000 where Emp_id = 7;
 update Employee -2 set salary = 35000 where Emp_id = 8;
 update Employee -2 set salary = 55000 where Emp_id = 9;
 update Employee -2 set salary = 25000 where Emp_id = 10;

Select * from Employee -2;

EMP_ID	EMP_NAME	DEPT	GENDER	AGE	CITY	SALARY
1	Kunal	CSE	Male	34	BBSR	40000
2	Anand	CSE	Male	32	BBSR	30000
3	Reema	IT	Female	27	BBSR	50000
4	Swati	IT	Female	26	Banglore	25000
5	Shiv	Bank	Male	30	Rourkela	30000
6	Abhishek	HR	Male	29	Delhi	20000
7	Rahul	CSCE	Male	25	CTC	45000
8	Pragyan	CSSE	Female	30	Puri	35000
9	Kaushal	IT	Male	32	Kolkata	55000
10	Seenu	Accounts	Female	33	Noida	25000

10 rows selected.

e) Display the details of female employee with age ≥ 30 in descending order of salary.

Select * from Employee -2 where gender = 'Female' and Age > 30 order by salary Desc;

<u>Emp_id</u>	<u>Emp_name</u>	<u>Dept</u>	<u>Gender</u>	<u>Age</u>	<u>city</u>	<u>salary</u>
10	Seenu	Accounts	Female	33	Noida	25000

f) Display the Emp_id, Emp_name and dept for the female employee whose salary is greater than or equal to 30000.

Select Emp_id, Emp_name, dept from Employee - 2 where gender = 'Female' and salary >= 30000.

<u>Emp_id</u>	<u>Emp_name</u>	<u>Dept</u>
3	Reema	IT
8	Pragyan	CSE

g) Display the average of the male employees.

Select Avg(Age) from Employee - 2 where gender = 'Male':

Avg(Age)

30.33333

h) Display the total salary for all the female employee who work in CSE Department.

Select sum(Salary) from Employee - 2 where gender = 'Female' and Dept = 'CSE';

sum(Salary)

i) Display the maximum and minimum salary of the employees.

Select Max(Salary), Min(Salary) from Employee-2;

Max(Salary)

55000

Min(Salary)

20000

j) Display the details of employee who get highest and lowest salary.

Select * from Employee-2 where salary in (

Select Max(Salary) from Employee-2);

<u>Emp-id</u>	<u>Emp-name</u>	<u>Dept</u>	<u>Gender</u>	<u>Age</u>	<u>city</u>	<u>Salary</u>
9	Kaushal	IT	Male	32	Kolkata	5500

Select * from Employee-2 where salary in (

Select min(Salary) from Employee-2);

<u>Emp-id</u>	<u>Emp-name</u>	<u>Dept</u>	<u>Gender</u>	<u>Age</u>	<u>city</u>	<u>salary</u>
6	Abhishek	HR	Male	29	Delhi	20000

Date : 14-02-2022 Aditya Ranjan Pradhan 20051620

Group By Having Clause in SQL

Question-1

Create table Emp with the attribute Emp-ID, Emp-Name, Job, Manager, Hire-Date, Salary, commission and dept. It is mandatory to enter employee ID and the same can not be repeated. Every employee must have a name then only their details can be entered in the table. similarly, salary component cannot be left blank for any employee. Each employee belongs to either dept 10, 20, or 30. Enter the records in the table as shown below.

7369	SMITH	CLERK	7902	17-DEC-80	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10
7788	SCOTT	ANALYST	7566	09-NOV-81	3000		20
7839	KING	PRESIDENT		17-NOV-81	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	23-SEP-81	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	950		30
7902	FORD	ANALYST	7566	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	1300		10

Creating the table.

```
create table employee1 (empid char(5) primary key,  
emp-name varchar(20) not null, job varchar(10),  
manager char(6), hire-date date, salary number(6)  
not null, commission number(4), dept number(4)  
check (dept in(10, 20, 30, 40)));
```

Inserting values into table.

insert into employee (emp-id, emp-name, job, manager, hire-date, salary, dept) values (7809, 'SMITH', 'CLERK', 7902, '17-DEC-80', 800, 20);

insert into employee (emp-id, emp-name, job, manager, hire-date, salary, dept) values (7499, 'ALLEN', 'SALES MAN', 7698, '20-FEB-81', 1600, 300, 30);

insert into employee (emp-id, emp-name, job, manager, hire-date, salary, commission, dept) values (7521, 'WARD', 'SALESMAN', 7698, '22-FEB-81', 1250, 500, 30);

insert into employee (emp-id, emp-name, job, manager, hire-date, salary, dept) values (7566, 'JONES', 'Manager', 7839, '22-APR-81', 2975, 20);

insert into employee (emp-id, emp-name, job, manager, hire-date, salary, dept, commission) values (7654, 'MARTIN', 'SALESMAN', 7698, '28-SEP-81', 1250, 1400, 30);

insert into employee (emp-id, emp-name, job, manager, hire-date, salary, dept) values (7698, 'BLAKE', 'MANAGER', 7839, '01-MAY-81', 2850, 20);

insert into employee (emp-id, emp-name, job, manager, hire-date, salary, dept) values (7782, 'CLARK', 'MANAGER', 7839, '09-JUN-81', 2450, 10);

insert into employee (emp-id, emp-name, job, manager, hire-date, salary, dept) values (7788, 'SCOTT', 'ANALYST', 7566, '09-Nov-81', 3000, 20);

insert into employee (emp-id, emp-name, job, manager, hire-date, salary, dept) values (7839, 'KING', 'PRESIDENT', '17-NOV-81', 5000, 10);

insert into employee(emp-id, emp-name, job, manager, hire-date, salary, commission, dept) values (4844, 'TURNER', 'SALESMAN', 7698, '08-SEP-81', 1500, 0, 30);
 insert into employee(emp-id, emp-name, job, manager, hire-date, salary, dept) values (7876, 'ADAMS', 'CLERK', 7788, '23-SEP-81', 1100, 20);
 insert into employee(emp-id, emp-name, job, manager, hire-date, salary, dept) values (7900, 'JAMES', 'CLERK', 7698, '03-DEC-81', 950, 30);
 insert into employee(emp-id, emp-name, job, manager, hire-date, salary, dept) values ('FORD', 'Analyst', 7902, '03-Dec-81', 3000, 20);
 insert into employee(emp-id, emp-name, job, manager, hire-date, salary, dept) values (4934, 'MILLER', 'CLERK', 7782, '23-JAN-82', 1300, 10);

a) Display the content of the table.

Select * from employee.

EMP_ID	EMP_NAME	JOB	MANAGE	HIRE_DATE	SALARY	COMISSION	DEPT
7369	SMITH	CLERK	7902	17-DEC-80	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	22-APR-81	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		20
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10
7788	SCOTT	ANALYST	7566	09-NOV-81	3000		20
7839	KING	PRESIDENT		17-NOV-81	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	23-SEP-81	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	950		30
7902	FORD	ANALYST	7902	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	1300		10

14 rows selected.

b) Display the emp-ID ,Emp-name and salary for salesman only .

Select emp-id ,emp-name ,salary from employee where job = 'SALESMAN';

EMP_ID	EMP_NAME	SALARY
7499	ALLEN	1600
7521	WARD	1250
7654	MARTIN	1250
7844	TURNER	1500

c) Display the details of the employee who is getting minimum salary.

Select * from employee where salary = (select min(salary) from employee);

EMP_ID	EMP_NAME	JOB	MANAGER	HIREDATE	SALARY
7369	SMITH	CLERK	7898	17-DEC-80	800

COMMISSION	DEPT
20.	

d) Display the details of the employee who works under the manager 7898 and getting highest commission.

Select * from employee where manager id = 7898 and commission = (select max (commission) from employee);

EMP_ID	EMPNAME	JOB	MANAGER	HIREDATE	SALARY	COMMISSION	DEPT
7654	MARTIN	SALESMAN	7898	28-SEP-81	1250	1400	30

E) Display the total salary of the employee working under 7839.

Select sum(salary) from employee where manager = 7839;

sum(salary)

F) Display the dept and avg salary of the employee working in dept wise manner.

Select dept, avg(salary) from employee group by dept;

<u>Dept</u>	<u>Avg(salary)</u>
30	1310
20	2287.5
10	2916.6667

G) Display the dept and avg salary of the employees in dept wise manner only when avg salary > 2000.

Select dept, avg(salary) from employee where salary > 2000 group by dept;

<u>Dept</u>	<u>Avg(salary)</u>
20	2956.25
10	3725

H) Display the dept and avg salary of only those employees whose salary > 2000 in dept wise manner.

Select dept, avg(salary) from employee 1 group by dept having avg(salary) > 2000;

<u>Dept</u>	<u>Avg(Salary)</u>
20	2287.5
10	2916.6667.

i) Display the dept and avg salary of only those employees whose salary > 2000 in dept wise manner only when avg salary < 3000.

Select dept, avg(salary) from employee 1 where salary > 2000 group by dept having avg(salary) < 3000;

<u>DEPT</u>	<u>Avg(SALARY)</u>
20	2956.56

Question-2

Create a table STUDENT with the attribute like YOA, ROLL, section and contribution. The section can be only either CS or IT where as the group column can have the values only for group-4, group-6 or group-7. Enter at least 10 entries into the table as shown below and perform the following operations.

YOA	ROLL	SE	GR	CON
2018	1805028	cs	gr4	80.5
2018	1805077	it	gr6	87.5
2019	1805062	it	gr7	90.5
2019	1805065	cs	gr7	92.5
2017	1806777	cs	gr6	92.5
2018	1806789	cs	gr4	95.59
2019	1805053	it	gr6	99.91
2018	1805653	it	gr6	100
2019	1805657	cs	gr7	100
2019	1805650	cs	gr7	100

Creating the table

Create table studentf (YOA Number (A), ROLL NO (10), section varchar(2) check (section in ('CS', 'IT')), GRP varchar(3) check (GRP IN ('GR4', 'GR6', 'GR7')), contribution Number (5,2));

Inserting values into table

Insert into studentf (YOA, ROLL, section, GRP, contribution)
values (2018, 1805028, 'CS', 'GR4', 80.5);

Insert into studentf (YOA, ROLL, section, GRP, contribution)
values (2018, 1805077, 'IT', 'GR6', 87.5);

Insert into studentt (YOA, ROLL, Section, grp, contribution)
 values (2019, 1805062, 'IT', 'GR7', 90.5);

Insert into studentt (YOA, ROLL, Section, grp, contribution)
 values (2019, 1805065, 'CS', 'GR7', 92.5);

Insert into studentt (YOA, ROLL, Section, grp, contribution)
 values (2017, 1806777, 'CS', 'GR6', 92.5);

Insert into studentt (YOA, ROLL, Section, grp, contribution)
 values (2018, 1806789, 'CS', 'GR4', 95.59);

Insert into studentt (YOA, ROLL, Section, grp, contribution)
 values (2019, 1805053, 'IT', 'GR6', 99.91);

Insert into studentt (YOA, ROLL, Section, grp, contribution)
 values (2018, 1805653, 'IT', 'GR6', 100);

Insert into studentt (YOA, ROLL, Section, grp, contribution)
 values (2019, 1805657, 'CS', 'GR7', 100);

Insert into studentt (YOA, ROLL, Section, grp, contribution)
 values (2019, 1805650, 'CS', 'GR7', 100);

a) Display the content of the table .

Select * from studentt.

YOA	ROLL	SE	GRP	CONTRIBUTION
2018	1805028	CS	GR4	80.5
2018	1805077	IT	GR6	87.5
2019	1805062	IT	GR7	90.5
2019	1805065	CS	GR7	92.5
2017	1806777	CS	GR6	92.5
2018	1806789	CS	GR4	95.59
2019	1805053	IT	GR6	99.91
2018	1805653	IT	GR6	100
2019	1805657	CS	GR7	100
2019	1805650	CS	GR7	100

10 rows selected.

b) Display in sorted order on year, 2011 and contribution using separate select statement.

Select * from studentt order by Roll;

```
SQL> SELECT * FROM STUDENT ORDER BY YOA;
```

YOA	ROLL	SE	GRP	CONTRIBUTION
2017	1806777	CS	GR6	92.5
2018	1805077	IT	GR6	87.5
2018	1806789	CS	GR4	95.59
2018	1805653	IT	GR6	100
2018	1805028	CS	GR4	80.5
2019	1805657	CS	GR7	100
2019	1805053	IT	GR6	99.91
2019	1805650	CS	GR7	100
2019	1805065	CS	GR7	92.5
2019	1805062	IT	GR7	90.5

10 rows selected.

Select * from studentt order by YOA;

YOA	ROLL	SE	GRP	CONTRIBUTION
2018	1805028	CS	GR4	80.5
2019	1805053	IT	GR6	99.91
2019	1805062	IT	GR7	90.5
2019	1805065	CS	GR7	92.5
2018	1805077	IT	GR6	87.5
2019	1805650	CS	GR7	100
2018	1805653	IT	GR6	100
2019	1805657	CS	GR7	100
2017	1806777	CS	GR6	92.5
2018	1806789	CS	GR4	95.59

10 rows selected.

Select * from studentt order by contribution;

YOA	ROLL	SE	GRP	CONTRIBUTION
2018	1805028	CS	GR4	80.5
2018	1805077	IT	GR6	87.5
2019	1805062	IT	GR7	90.5
2019	1805065	CS	GR7	92.5
2017	1806777	CS	GR6	92.5
2018	1806789	CS	GR4	95.59
2019	1805053	IT	GR6	99.91
2018	1805653	IT	GR6	100
2019	1805657	CS	GR7	100
2019	1805650	CS	GR7	100

10 rows selected.

c) Display the number of rows in the table.

Select count(*) from student;

COUNT(*)

10

d) Display total contribution for the group-6.

Select sum(contribution) from student
where Grop = 'GR 6';

SUM(CONTRIBUTION)

379.91

e) Display the average contribution for group-7.

Select Avg(contribution) from student where Grop = 'GR 7'

Avg(CONTRIBUTION)

95.75

f) Display the maximum contribution for only group-4 students.

Select MAX(contribution) from student where Grop = 'GR 4';

MAX(contribution)

95.59

g) Display the minimum contribution for only "cs"

Select MIN(contribution) from student where section = 'CS';

min(contribution)

80.5

iii) Display the roll no of student having highest contribution.

Select roll from student where contribution =
(select max (contribution) from student);

ROLL
1805653
1805657
1805650

(i) Display the roll no and yoa of the student having smallest contribution.

Select roll, yoa from student where contribution =
(select min (contribution) from student);

ROLL	YOA
1805028	2018

(j) Display number of student in group - 7.

Select count (*) from student group by grp having grp = 'GR7';

COUNT(*)
4

(k) Display number of students in group wise manner.

Select grp, count (*) from student group by grp;

grp	COUNT(*)
GR7	4
GR6	4
GR7	2

- i) Display group, contribution (group) in group wise manner.

Select grp, sum (contribution) from student group by grp;

<u>grp</u>	<u>sum (contribution)</u>
GR7	383
GR6	379.91
GR4	176.09

- m) Display group, average contribution (group) in a group wise for only the group where the contribution > 90 .

Select grp , Avg (contribution) from student where contribution > 90 group by grp.

<u>grp</u>	<u>Avg (contribution)</u>
GR7	95.75
GR6	97.47
GR4	95.59

- n) Display section, no. of student , contribution in section wise manner .

Select section, count (*), sum (contribution) from student group by section

<u>section</u>	<u>count (*)</u>	<u>sum (contribution)</u>
IT	4	327.91
CS	6	561.09

O) Display section, no. of student; contribution in section wise manner for 2019 batch.

Select section, count(*), Avg (contribution)
Student where YOA=2019 group by section;

<u>SE</u>	<u>Count (*)</u>	<u>Avg (contribution)</u>
IT	2	95.205
CS	3	97.5

P) Display year, max contribution on year of admission.

Select YOA, MAX(contribution) from
Student group By YOA;

<u>YOA</u>	<u>MAX (contribution)</u>
2018	100
2017	92.5
2019	100