Day-3 Hands on and Functions in Sql

Name: Aathirainathan P

Date: 06-11-2024

1.Display table structure:

Handsonfuction.sq...(PUMA\pumak (61))*

```
create database mydb;
use mydb;
create table employee(
id int primary key,
name varchar(20),
age int,
email varchar(20)
);/*creatig table*/
select * from employee;
sp_help employee;/*displaying the structure of the table*/
insert into employee(id,name,age,email) values(101,'alex',30,'alex@gmail.com');
insert into employee(id,name,age,email) values(102,'bob',20,'bob@gmail.com');
insert into employee(id,name,age,email) values(103,'sunny',40,'sunny@gmail.com');
insert into employee(id,name,age,email) values(104,'stella',35,'stella@gmail.com');
```

89 %

Results | Messages

	Name	Owner	Type	Created_datetime
1	employee	dbo	user table	2024-11-06 12:30:45.733

	Column_name	Type	Computed	Length	Prec	Scale	Nullable	TrimTrailingBlanks	FixedLenNullInSource	Collation
1	id	int	no	4	10	0	no	(n/a)	(n/a)	NULL
2	name	varchar	no	20			yes	no	yes	SQL_Latin1_General_CP1_CI_AS
3	age	int	no	4	10	0	yes	(n/a)	(n/a)	NULL
4	email	varchar	no	20			yes	no	yes	SQL_Latin1_General_CP1_CI_AS

	Identity	Seed	Increment	Not For Replication
1	No identity column defined.	NULL	NULL	NULL

	RowGuidCol
1	No rowguidcol column defined.

	Data_located_on_filegroup
1	PRIMARY

	index_name	index_description	index_keys
1	PK__employee__3213E83F339E4044	clustered, unique, primary key located on PRIMARY	id

	constraint_type	constraint_name	delete_action	update_action	status_enabled	status_for_replication	constraint_keys
1	PRIMARY KEY (clustered)	PK__employee__3213E83F339E4044	(n/a)	(n/a)	(n/a)	(n/a)	id

2.Alter Table:

```
select distinct name from employee;/*using distinct constraint*/
alter table employee add  salary int;/*add one more column into table*/
select *from employee;
update employee set salary=20000 where name='alex';
update employee set salary=30000 where name='bob';
update employee set salary=35000 where name='sunny';
```

Messages

```
Commands completed successfully.

Completion time: 2024-11-06T12:36:30.8030127+05:30
```

3.Aggregate Functions:

```
select avg(salary) from employee
where department='HR';/*using avg function */
select count(id) from employee
where name='stella';/*using count function*/
select max(salary),department from employee
group by department
having department='HR';/*using group by,having with aggregate function*/
select count(name) as count_employee,salary from employee
group by salary
having salary=50000;/*using having and group by clause*/
select *from employee;
select max(salary) as min_salary,department
from employee
group by department
having department='IT';
```

89 %

Results | Messages

	(No column name)
1	45000

	(No column name)
1	2

	(No column name)	department
1	50000	HR

	count_employee	salary
1	2	50000

	id	name	age	email	salary	department
1	101	alex	30	alex@gmail.com	20000	IT
2	102	bob	20	bob@gmail.com	30000	salesforce
3	103	sunny	40	sunny@gmail.com	35000	HR
4	104	stella	35	stella@gmail.com	35000	sales
5	105	nani	40	nani@gmail.com	50000	HR
6	106	stella	35	stella@gmail.com	35000	sales
7	107	nani	35	nani@gmail.com	50000	HR

	min_salary	department
1	20000	IT

4.Rollback a transaction:

```
begin tran;
delete from employee where age=40;
rollback;/*rollback the transaction*/
insert into employee(id,name,age,email,salary,department) values(101,'aparna',20,'aparna@gmail.com',30000,'sales');
insert into employee(id,name,age,email,salary,department) values(102,'stella',20,'stella@gmail.com',40000,'HR');
select *from employee;
begin tran;
delete from employee where age=20;
```

89 %

Messages

```
Commands completed successfully.

Completion time: 2024-11-06T12:40:18.2809430+05:30
```

5. Savepoint and rollback:

```
begin tran;
delete from employee where id=102;
save transaction s1;/*create a savepoint s1*/
rollback transaction s1;
select *from employee;
begin tran;
insert into employee(id,name,age,email,salary,department) values(103,'bob',45,'bob@gmail.com',30000,'IT');
save transaction s2;/*creating the savepoint s2*/
```

Messages

```
Commands completed successfully.

Completion time: 2024-11-06T12:41:36.5246230+05:30
```

6.Union, Union all, Intersection, Except:

```
select *from employee
union
select *from emp;/*performing union with two tables*/

select *from employee
union all
select *from emp;/*performing union all with two tables*/

select *from employee
intersect
select *from emp;/*performing intersection between the tables*/

select *from emp
except
select* from employee;/*performing except  operation between two tables*/
```

Results | Messages

	id	name	age	email	salary	department
1	101	alex	30	alex@gmail.com	20000	IT
2	104	stella	35	stella@gmail.com	35000	sales
3	105	stella	20	stella@gmail.com	40000	HR
4	106	stella	35	stella@gmail.com	35000	sales
5	107	lucky	28	lucky@gmail.com	60000	salesforce
6	107	nani	35	nani@gmail.com	50000	HR
7	108	nimmi	30	nimmi@gmail.com	30000	IT
8	109	krish	30	krish@gmail.com	45000	sales

	id	name	age	email	salary	department
1	101	alex	30	alex@gmail.com	20000	IT
2	104	stella	35	stella@gmail.com	35000	sales
3	105	stella	20	stella@gmail.com	40000	HR
4	106	stella	35	stella@gmail.com	35000	sales
5	107	nani	35	nani@gmail.com	50000	HR
6	108	nimmi	30	nimmi@gmail.com	30000	IT
7	109	krish	30	krish@gmail.com	45000	sales
8	201	bob	45	bob@gmail.com	30000	IT

	id	name	age	email	salary	department
1	108	nimmi	30	nimmi@gmail.com	30000	IT

id	name	age	email	salary	department

Query executed successfully.

7.Joins:

```
select name,cid,department from employee
join
course on  employee.id=course.id;/*performing the natural join*/

select * from employee
cross join
course ;/*performing the cross join*/

/*Performing equi join*/
select *from employee,course
where employee.id=course.id;

/*Perfoming non-equi join*/
select *from employee,course
where employee.id>=course.id;

/*Perfoming self join*/
select e1.name,e1.id,e1.salary from employee e1,employee e2
where e1.salary=e2.salary and e2.name='krish';

select *from employee;
```

%

Results | Messages

name	id	salary
krish	109	45000

8.String Functions:

```
/* String functions*/
select ascii('CB'); /*return leftmost ascii value*/
select char(66); /*return ascii value to character*/
select len('Microsift sql');/*return length*/
select lower('JHON');/*convert to lowercase*/
select replace('Microsoft sql','sql','server');/*replace*/
select reverse('python');/*reverse the string */
select upper('aparna');/*converts to upper*/
select str(136.564,8,4);/*STR(number, length, decimals)*/

/* Date Functions*/
```

89 %

Results | Messages

	(No column name)
1	67

	(No column name)
1	B

	(No column name)
1	13

	(No column name)
1	jhon

	(No column name)
1	Microsoft server

	(No column name)
1	nohtyp

	(No column name)
1	APARNA

	(No column name)
1	136.5640

9.Date Functions:

```
/* Date Functions*/
select  getdate ();/*get current date and time*/
select dateadd (mm, 2, '2023-12-07');/*add months to existed date */
select datediff ( year, convert (datetime, '2006-05-06'), convert ( datetime, '2009-01-01'));/*it will return the difference of date,months,years also*/
select datepart (mm, '2008-5-22');/*return months value*/
select day ( '2023-05-30');/*return value of date of that particular day*/
select month ('2023-05-31');/*return month value*/
select year ( '2023-05-3');/*return year value*/

/*Mathematical Functions*/
select abs(-101);/*returns absolute value*/
select sin(1.5);/*returns angle in radians*/
select ceiling(14.01);/*returns the smallest or greater to the specified value*/
```

9 %

Results | Messages

	(No column name)
1	2024-11-06 12:53:31.587

	(No column name)
1	2024-02-07 00:00:00.000

	(No column name)
1	3

	(No column name)
1	5

	(No column name)
1	30

	(No column name)
1	5

	(No column name)
1	2023

10.Mathematical Functions:

```
/*Mathematical Functions*/
select abs(-101);/*returns absolute value*/
select sin(1.5);/*returns angle in radians*/
select ceiling(14.01);/*returns the smallest or greater to the specified value*/
select exp(4.5);/*returns the exponencial value*/
select floor(14.75);
select log(5.4);/*return logarithmic value*/

/*Rankig Functions*/
/* row()_number-giving consecutive numbers to rank*/
select * from employee;
```

3 %

Results | Messages

(No column name)
101

(No column name)
0.997494986604054

(No column name)
15

(No column name)
90.0171313005218

(No column name)
14

(No column name)
1.68639895357023

11.Ranking Functions:

```
/* row()_number-giving consecutive numbers to rank*/
select * from employee;
select id,name,salary,ROW_NUMBER() over(order by salary desc) as rownumber
from employee;
select * from employee;
/*rank()-used to give rank if duplicates allowed ranking will be changed based on duplicates
select id,name,salary,rank() over(order by salary) as rank
from employee;

/*dense_rank()-used to give ranks consecutively even if duplicates are allowed*/
select id,name,salary,dense_rank() over(order by salary desc) as rank
from employee;

/*ntile() function- it will divide give the rank in groups*/
select id,name,salary,ntile(2) over(order by salary) as rank
from employee;/*without condition*/

select name,salary,ntile(4) over(order by salary) as rank
from employee where salary>10000;/*with condition*/
```

98 %

Results | Messages

	id	name	age	email	salary	department
1	101	alex	30	alex@gmail.com	20000	IT
2	104	stella	35	stella@gmail.com	35000	sales
3	105	stella	20	stella@gmail.com	40000	HR
4	106	stella	35	stella@gmail.com	35000	sales
5	107	nani	35	nani@gmail.com	50000	HR
6	108	nimmi	30	nimmi@gmail.com	30000	IT
7	109	krish	30	krish@gmail.com	45000	sales

	id	name	salary	rownumber
1	107	nani	50000	1
2	109	krish	45000	2
3	105	stella	40000	3
4	106	stella	35000	4
5	104	stella	35000	5
6	108	nimmi	30000	6
7	101	alex	20000	7

Query executed successfully.

12. System Functions:

```
/*System functions*/
select host_name() as Host_name;/*return system hostname*/
select Host_ID();/*return host id*/
select SUSER_ID();
select USER_ID();
select DB_NAME();/*return database name*/

/*Aggregate funtions*/
select count(*) as total_records from employee;

select count(*) from employee where salary>30000;
```

98 %

Results | Messages

	Host_name
1	PUMA

	(No column name)
1	20396

	(No column name)
1	259

	(No column name)
1	1