

Practice DBMS

#Create table in the database.

```
CREATE TABLE example(  
    id SERIAL PRIMARY KEY,  
    first_name VARCHAR(30) DEFAULT 'N/A',  
    last_name VARCHAR(30) NOT NULL,  
    email_add VARCHAR(40) UNIQUE,  
    nationality CHAR(30)
```

```
);  
SELECT * FROM example;
```

#add column using alter command

```
ALTER TABLE example  
ADD COLUMN age INT;
```

```
#Alter type char length.  
ALTER TABLE example  
ALTER COLUMN first_name TYPE VARCHAR(50);
```

```
#Insert values in table.  
INSERT INTO example (first_name,last_name,email_add,nationality,age)  
VALUES('Sahil','Chavan','sahil18@gmail.com','MAH','25'),  
      ('Sagar','Mali','sm26494@gmail.com','IND','26'),  
      ('Malkeet','Rathod','malkeet.rathod@gmail.com','CAN','25'),  
      ('Anushka','Samnerkar','anusamnerkar@gmail.com','GRE','23');
```

```
#Update values in table.  
UPDATE example  
SET age='26'  
WHERE age='25';
```

```
UPDATE example  
SET nationality='NOR'  
WHERE nationality='MAH';
```

```
UPDATE example  
SET nationality='USA', age='25'  
WHERE first_name='Sahil';
```

```
UPDATE example
SET nationality='GRE'
WHERE nationality='gre';
```

```
#To show unique values in the table.
SELECT DISTINCT first_name FROM example;
```

```
#Selecting specific data.
SELECT * FROM example
WHERE age='26';
```

```
#Sorting in ASC or DESC. By default asc.
SELECT * FROM example
ORDER BY age DESC;
#can sort more than one column but set preference first.
```

```
#delete and rollback data.
begin;
delete from example
where age=23
;
```

```
rollback;
select * from example;
```

Function 1

```
create or replace function num3(a int,b int)
returns int as $$
declare
c int;
begin
c=a+b;
return c;
end;
$$
language plpgsql;

select num3(34,54);
```

Function 2

```
create or replace function num4()
returns table (salary int)
as $$
begin
return query select max(e.salary) from public.e;
end;
$$
language plpgsql
```

```
select num4()
```

Function 3

```
create or replace function num8()
returns table (first_name varchar,creditlimit numeric)
as $$
begin
return query select customers.first_name,customers.creditlimit from public.customers;
end;
$$
language plpgsql;
```