customer(cno,cname,mobile)

to add a column in customer table

alter table customer

add email varchar

to delete column from customer table

alter table customer

drop email varchar

Data types collection

1. list
    a. duplicate values are allowed
    b. it is represented as []
    c. it is ordered collection, hence indexing is possible.
    d. It is mutable

add in the list at the end	update student set hobbies=hobbies+['trekking'] where sid=1;
add in the list at the begining	update student set hobbies=['trekking']+hobbies where sid=1;
delete from the list	update student set hobbies-['trekking'] where sid=1;
overwrite the list 1st index position with some value	update student set hobbies[1]='trekking' where sid=1;

2. set
    a. it is a collection of unique values
    b. it is represented using {}
    c. It is unordered, and hence no indexing is possible
    d. It is mutable

add value in the set	update customer set brands=brands+{'a','b'} where cno=1;
delete value from the set	update customer set brands=brands-{'a','b'}; where cno=1;

delete all values from the set	update customer set brands={} where cno=1;
assign or overwrite value in the set	update customer set brands={‘puma’,’bata’} where cno=1;

3. map
    a. it allows to store key-value pair
    b. keys should be unique
    c. data is stored in {}
    d. values can be retrieved by using keys.

add value in the map	update student set marks=marks +{'java':98} where sid=1;
delete all values from the map	update student set marks={} where sid=1;
delete some key-value pair	update student set marks=marks -{'java','python'} where sid=1;
assign or overwrite value in the map	update student set marks[‘perl’]=100 where sid=1;

4. tuple
    a. duplicates are allowed
    b. it is ordered collection, so indexing is possible
    c. tuples are immutable
    d. it uses ()
    e. these are fixed length data
    alter table student add degree tuple<text,text,int>;

    cqlsh:iacsd0324> update student set degree=('Mtech','PU',89)
    where sid=1;

create table test1(id int, name varchar,data list<tuple<int,text>>,data2 map<text,list<text>>)

-----create custom data types

create type address(street text,zipcode text,city text)

create table suppiler(sid,sname,saddr adress)

create table customer1(sid,sname,saddr list<FROZEN<address>>)

insert into supplier(sid,sname,saddr)
values(11,’xxxx’,{‘street’:’baner’,’zipcode’:’11111’,city:’Pune’})

---add new field in the type

alter type address add bldgnm text;

---to rename the field in the type

alter type address rename bldgnm to bname

In Cassandra we can execute batch operation

begin batch

insert into customer(cno,cname,brands,mobile,billamt) values(111,'xx',{'a','b'},'2222',44444)

insert into customer(cno,cname,brands,mobile,billamt) values(112,'yy',{'x','y'},'2222',66666)

delete mobile from customer where cno=1

apply batch

create index idxname om customer(billamt)

-------to insert data in json format

insert into customer JSON '

{"cno":12,"cname":"dfsd","brands":["a","b"],"mobile":"34567",

"billamt":4567}';