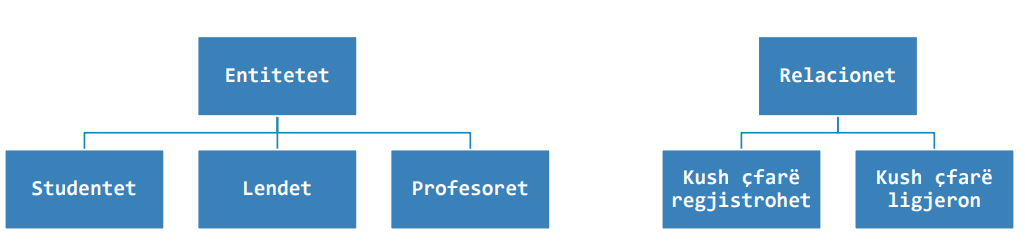
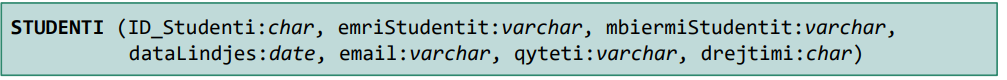
**Hyrje në SQL**

**Entitetet** (p.sh., Studentet, Lëndet, Profesoret) – janë si lloj objektesh.

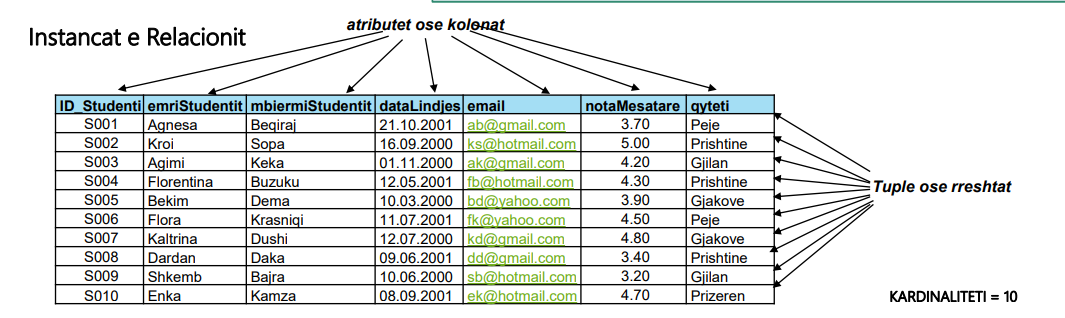
**Relacionet**(p.sh., Filani është i regjistruar në SHKI) -lidhje mes entiteteve. 

**Baza e të dhënave relacionale :** Është bashkësi e relacioneve. **Relacioni** është term matematikor për tabelë. **Tabela** përbëhet nga rreshtat dhe kolonat.

**Relacioni** përbëhet nga dy pjesë : **Skema Relacionale –** në mënyr logjike e definojnë organizimin e një relacioni , **Instanca Relacionale –** të dhënat mbrenda ati relacioni.

**Skema Relacionale :** Na mundëson ti vendosim një emër relacionit (tabelës) , gjithashtu ti vendosum emrin dhe tipin e secilës kolonë (atribut). 

**Kolona = atribute = fusha. Reshtat = tuple.**

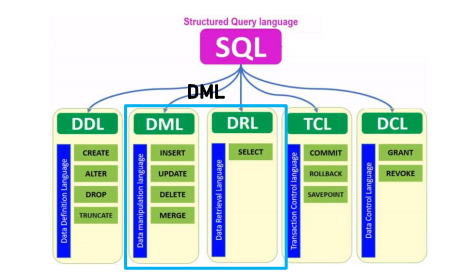
****

Secial fushë(kolone) është atomike , dmth mbajnë të dhëna të thjesha (emri dhe mbiemri duhet të jenë të ndara në kolona të veqanta).

**Kardinaliteti –** numri i rreshtave.

**SQL ( Structured Query Language) –** Është një gjuhë standarde, gjuhë programuese e nivelit të lartë për kërkimin dhe manipulimin e të dhënave.

**SQL-** Na mundëson manipulimin dhe kërkimin e të dhënave nga tabelat e Data Bazës Relacionale nga ana e përdoruesve.

Ekzistojnë disa nën gjuhë të gjuhës **SQL,** dy kryesoret janë : **DML(**Data Manipulation language) - Gjuhë për manipulimin me të dhëna ,query/INSERT/DELETE/UPDATE/SELECT tuples(rreshtat) në tabelë. **DDL(**Data Defination Language) -Gjuhë për definimin e të dhënave ,përcaktimin e skemës relacionale, krijimin e tabelave , CREATE/ALTER/DROP tabelat dhe atributet(kolonat) e trye.

**Tabela** ose **Relacioni**  në **SQL** : Është një shum e grupimit të tuples(rreshtave) të cilët kanë atribute (kolona) të specifikuara nga skema e relacionit.

Një **atribut**(ose **kolonë**) është një tip i të dhënës i cili është i pranishëm në secilën tupel(rresht) në relacion. Atibutet duhet të jen të llojit atomik.

Një **tuple**(ose **rresht**) është një input i vetëm në tabel që ka atributet e specifikuara nga skema.

Llojet atomike të të dhënave - **Karaktere:** Char(20), VarChar(50). **Numrat :** int, bigint , smallint , float. Tjera : Money, datetime.

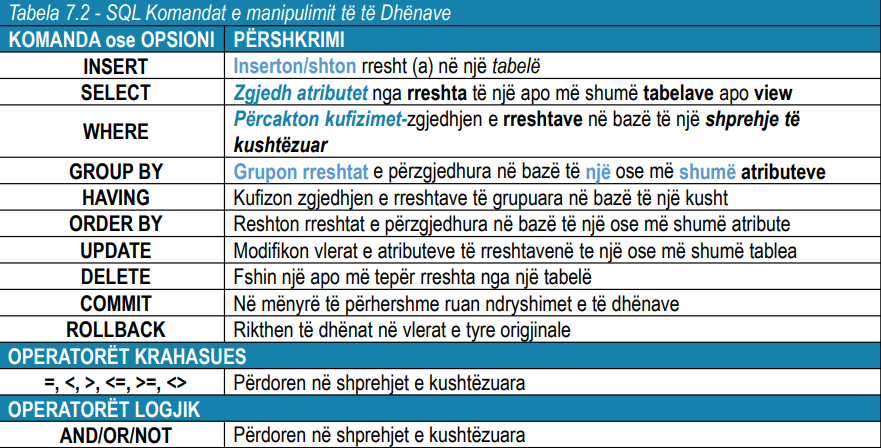
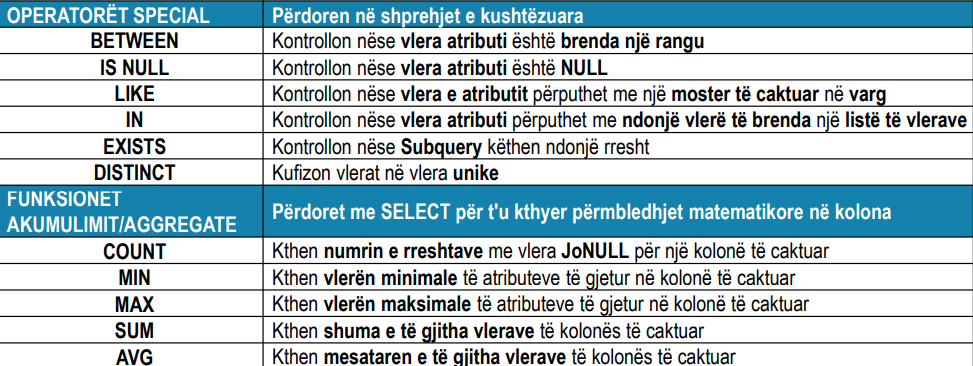
Char dhe varchar janë vargje të karaktereve të cilat marin nga 1 byte për karakter.

Dallimi mes **char** dhe **varchar** është se char zë hapsirë fike në disk pra sa i caktojmë ne madhësin edhe nëse nuk shtojm aq karaktere në rastin char(10) do zë 10 byte në disk duke i numruar edhe hapsirat boshe, ndërsa **varchar** merr nga një bajt për secilin karakter dhe 2 byte për informacionin e madhësisë së vargut.

**Skema e nje tabele** është : emri i tabelës, atributet dhe llojet e tyre. 

**Priamry key :** Një çelës është një nëngrup minimal i atributeve që vepron si një identifikues unik për tuples(rreshtat) në një relacion. Ka vetëm një qelës primar, ku çelës mund të jet vetëm një atribut(kolon) ose mund të jen disa kolona por cilsohen si një çelës i vetëm.

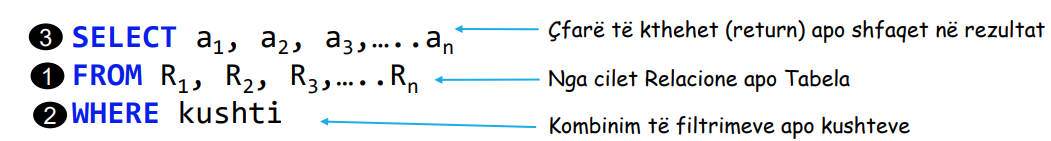
Për të definuar një qelës ai duhet ti plotësoj këto kusht : të mos përsëritet ,të ekzistoj gjithmonë dhe a mund të kemi më shumë se një qelës të ngjajshëm.

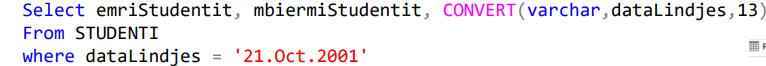
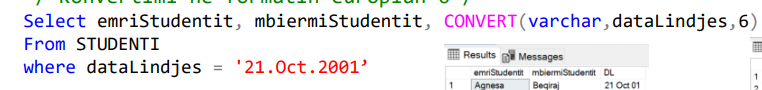


Për të shtuar dhëna në një DB , duhet të përdorim deklaratën **INSERT** ,e cila duke si më poshtë :

INSERT INTO \*Emri Tabeles\* VALUES( Vlerat);

Për të shfaqur rezultat në konzolë përdoret deklarata **SELECT** , e cila duket si më poshtë :

SELECT \* FROM \*Emri i Tabeles\*

Për të konvertuar të dhënat përdoret funksioni **CONVERT** , disa shembuj me poshtë : 

Në rast se duam që të dhënat që janë të njëjta të paraqiten vetëm njëher , përdoret deklarata **DISTINCT** , duket si më poshtë :

SELECT DISTINCT S.qyteti

FROM Studenti S

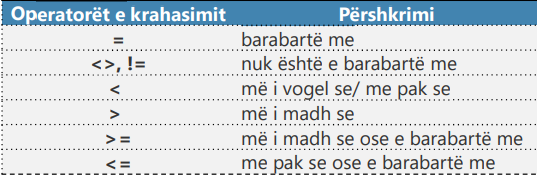
Në rast se duam që të dhënat në rezultat të dalin të rënditura atëher duhet të përdorim deklaratën ORDER BY , e cila duke si më poshtë :

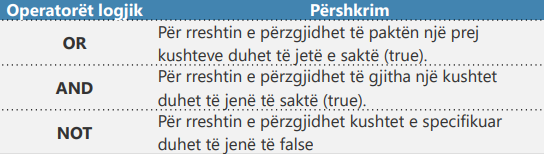
SELECT \* FROM Studenti S

ORDER BY S.emri

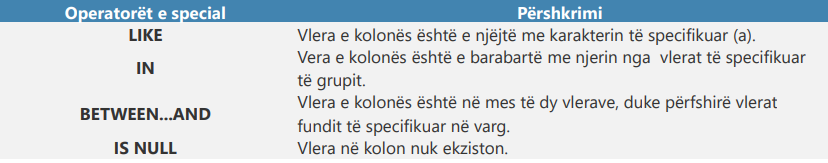
Në SQL kemi dy lloje të operatorëve : **operatorët logjik** dhe **operatorët krahasues**. Këta operator zakonisht përdoren në klauzolën WHERE dhe HAVING , për ti filtruar të dhënat sipas përzgjidhes së caktuar nga kushtet.

**Operatorët e krahasimit** përdoren për të krahasuar të dhënat në kolona të veqanta në filtrime të caktuara.



**Operatorët Logjik** : janë tre operator logjik AND , OR ,NOT , këta operatorë mundësojnë krahasimin e dy kushteve një kohësisht për të pëercaktuar outup të përzgjedhur për një rresht.

Në SQL ekzistojnë edhe **operatorët special** për krahasimin e të dhënave të cilët na japin më shum mundësin në kërkim të query SQL , ato janë : **LIKE, IN. BETWEEN, AND, IS NULL ...**



**LIKE** përdoret për krahasimin e të dhënave te tipit char ose varchar , përdoren disa simbole shtesë siq janë **%-**përputhet me disa karaktere , **\_** - përputhet me vetëm një karakterë.Shembuj të **LIKE :**

SELECT \* FROM Studenti S

WHERE S.qyteti LIKE ‘P%’

SELECT \* FROM Studenti S WHERE S.qyteti LIKE ‘\_rishtin’

**BETWEEN** përdoret kur duam ti përfshijmë dy skajet , shembull i ksaj deklarate : //të shfaqen studentet me not mesatare mes 4 dhe 5

SELECT \* FROM Studenti S

WHERE s.notaMesatare BETWEEN 4 AND 5

Deklarata **IN :**

//të shfaqen studentet me not mesatare 4.5, 4.7, 5

SELECT \* FROM Studenti S

WHERE S.notaMesatare IN(‘4.5’,’4.7’,’5’)

Deklarata **UPDATE** përdoret kur duam të përditsojmë ndonjë të dhëna në tabelë , shembull i deklaratës :

//të përditësohet qyteti i studentit me ID 01 nga pej në prishtinë

UPDATE Studentet (emri i tabelës)

SET qyteti=’Prishtinë’

WHERE id\_Studenti=’01’

Në SQL mund ti përdorim gjithashtu edhe operatorët aritmetikë si : **+ ,- ,\*, /, ^ ,%** ,shembuj të përdorimit :

--Shfaqe çmimin e CDs se 'Bajrush doda' pas aplikimt te zbitjes 5 euro

SELECT titulli, (qmimi-5) as ‘qimi pas zbritjes 5 euro’

FROM CD

WHERE titulli=’Bajrush doda’