

**Punuan:**

*[Student1]* Albi Ndroqi *[Student2]* Albi Taulla

*[Student3]* Ersa Doka

*[Student4]* Iris Dizdari

**Pranoi:**

Dr. Silvana Greca

**Lënda:** Databaza nën Oracle

**Semestri I**

**Tema:** Sistem për menaxhimin e të dhënave të një metroje

**REPUBLIKA E SHQIPËRISË**

**UNIVERSITETI I TIRANËS**

**FAKULTETI I SHKENCAVE TË NATYRËS**

**Përmbajtja:**

Grupi 1

**1**. Përshkrimi i bazës së të dhënave 3

**2**. Skema ER/EER 4

**3**. Relacionet 5

**4**. Krijimi i bazës së të dhënave 7

**5**. Query 11

**6**. View 20

**7.** Indexet 23

Përshkrimi i bazës së të dhënave

Databaza eshte ndertuar mbi idene menaxhimit te te dhenave te nje sistemi metroje.Databaza ka si qëllim të mundësojë funksionalitete si marrja e informacionit për trenat e disponueshëm bazuar mes dy stacioneve, blerja e biletave dhe anullimi i tyre, marrja e informacionit mbi një biletë, etc.

Databaza lejon rezervimin e një bilete duke marrë informacion mbi Idnë e trenit dhe udhëtimin që ky tren realizon në nje orar të caktuar.Gjithashtu databaza ruan informacion mbi të gjithë stacionet e kësaj metroje,punonjesite ndryshëm që ajo mund të ketë dhe rrugët që lidhin këto stacione.

**Entiteti**: **Person,** ruan informacion mbi nje person qe ne rastin tone mund te jete nje punonjes ose nje pasagjer I thjeshtë.Informacionet janë të dhëna bazike të një personi në shoqëri,

**Atributet**:

*ID* - id identifikuese e parsonit te zakonshëm

*Emer* – emri i personit

*Mbiemer* - mbiemri i personit

*Gjinia* – gjinia e personit

*Data\_E\_Lindjes* – datëlindja e personit

*Email* – email i personit

*Shteti*- shteti ku jeton ky person

*Qyteti* – qyteton ku jeton ky person

Numer\_Telefoni – Numri i telefonit

**Entiteti**: **Pasagjer,** perfaqëson një pasagjer të thjeshtë të metrosë.

**Atributet**:

**Entiteti**: **Punonjës**, përfaqëson një punonjës të një stacioni.

**Atributet**:

*Pozicion* – pozicioni I punës së një personi

*Orët\_E\_Punës* – orët qe një punon një person cdo ditë.

*Data\_E\_Punësimit* – Data e punësimit të një punonjësi.

*SUBKLASAT:*

Shofer – Certifikata

Konduktor – Niveli\_GCSE

Inxhinier – Niveli

Menaxher – Niveli\_GCSE

**Entiteti**: **Stacionet**, ku ruhen të dhënat që përmban një stacion i metrosë.

**Atributet**:

*ID – identifikuesi I një stacioni*

*Emër – emri I qytetit*

*Qyteti – Qyteti ku ndodhet stacioni*

*Rruga – Rruga që shërben si adresa e stacionit*

**Entiteti**: **Treni**, ku ruhen të dhënat për cdo tren.

**Atributet**:

*ID – Numri identifikues I një treni*

*Modeli – Modeli I një treni*

*Ngjyra – Ngjyra e trenit*

*Numër\_Vendesh – Numri I vendeve në tren*

*Datë\_Prodhimi – Data e prodhimit të trenit*

*Shpejtësi\_Maksimale – Shpejtësia maksimale që mund të arrijë treni*

*Numër\_Vagonash – Numri pjesëve ndarëse të trenit*

**Entiteti**: **Abonim**, përfaqëson abonetë e blera nga persona të ndryshëm.

**Atributet**:

*ID* – numri identifikues i një aboneje

*Cmimi* – cmimi I abonesë

*Numri\_Udhëtimeve* – Numri i udhëtimeve që mundëson aboneja

*Klasa\_E\_Vendeve – Klasa ku do të udhëtojë pasagjeri*

**Entiteti**: **Bileta**, përfaqëson informacionin në lidhje me një biletë

**Atributet**:

*ID* – numri identifikues i një bilete

*Cmimi* – cmimi I biletës

*Data\_E\_Blerjes* – Data e blerjes së biletës

*Klasa– Klasa ku do të udhëtojë pasagjeri*

**Entiteti**: **Udhëtim**, përfaqëson itinerarin e udhëtimeve

**Atributet**:

*ID* – numri identifikues i një bilete

*Data*– Data e udhëtimit

*Koha\_E\_Nisjes*

*Koha\_E\_Mbërritjes*

***Skema ER/EER***

Map

Description automatically generated

***Relacionet***

Fillimisht do të shndërrojmë të gjithë entitetet e forta në tabela.Shohim se në rastin e entitetit **Person** ai ka dy subklasa **Punonjës** dhe **Pasagjer** për të cilat kufizimi I plotësisë është total.Pra në kalimin nga EER në relacional klasa **Person** nuk do të shndërrohet në relacion por vetëm subklasat e saj do të bëhen relacione duke trashëguar cdo atribut prej superklasës.

Skema jonë nuk përmban entitete të dobëta kështu që nuk duhet të mendojme për kalimin e celësave mes entiteteve të forta dhe të dobëta.

Atributet komplekse si kontakt zbërthehen në atribute atomike të cilat vendosen brenda relacioneve.Atributet shumëvlerëshe shndërrohen në relacione më vete dhe mbajne ID e relacionit për të cilin ato ruajnë të dhënën specifike.

Nisim me marrdhëniet e tipit 1:1.Këto marrdhënie janë **Menaxhon** mes entitetit **Menaxher** dhe **Stacion** ku cdo për një stacion kemi vetëm një menaxher dhe për një menaxher kemi vetëm një stacion të cilin ai e menaxhon,**Drejton,Përgjegjës** mes **Shofer,Konduktor**  dhe **Tren** respektivishtku për cdo **Tren** kemi vetëm një **Shofer**  dhe për cdo **Shofer v**etëm një tren që ai drejton si dhe për cdo **Konduktor** kemi vetëm një tren për të cilin ai është përgjegjës dhe për cdo **Tren** kemi një konduktor qe e menaxhon atë.

Marrdheniet e tipit 1:N jane :

**Punon** mes **Stacion** dhe  **Punonjës** ku për një stacion mund të kemi disa punonjës(N) ndërkohe për një punonjës kemi vetëm një stacion ku ai punon(1).

**I\_Perket** mes **Abonim** dhe **Pasagjer** ku per një abonim kemi një pasagjer(N) ndërsa për një pasagjer mund të kemi disa abonime në emrin e tij(N).

**Blen** mes **Pasagjer** dhe **Biletë** ku cdo pasagjer mund të bleje shumë bileta(N) ndërsa një biletë mund ti përkasi vetëm një pasagjeri(1).

**Per** mes **Udhetim** dhe **Biletë** ku për cdo udhëtim kemi shumë bileta(N) ndërsa një biletë shërben për të realizuar vetëm një udhëtim.

Per të gjithë mardhëniet e mësipërme qe te shmangim teprinë e informacionit kalojme celësin primar të relacionit(1) dhe atributet e marrdhënies brenda relacionit(N).

Marrdhënia **Udhëton** është e tipit 1:1:1:N sepse për një udhëtim,një stacion dhe një tren cfarëdo në një orar të caktuar mund të kemi vetëm një stacion(nisje/mbërritje).Për një stacion nisje, një stacion mbërritje dhe një udhëtim të caktuar mund të kemi vetëm një tren që e realizon këtë udhëtim.Ndërsa për dy stacione specifike, një tren të caktuar mund të kemi shumë udhëtime në orare të ndryshme.Për të shmangur krijimin e një relacioni për marrdhënien kalojmë celësat primarë të stacioneve dhe trenit brenda relacionit udhëtim.

Së fundmi kemi marrdhënien **Lidhen** mes dy stacioneve që është e tipit shumë me shumë pasi cdo stacion mund të lidhet me shumë stacione të tjere.Për këtë marrdhënie rekursive krijojmë një relacion më vete pasi nuk mund të kalojmë celësat primarë të asnjërit prej relacioneve tek tjetri pasi as1 nuk e identifikon unikisht këtë marrdhënie.

Përsa i përket klasave **Menaxher,Inxhinier,Shofer** dhe **Konduktor**  ato kalojne secila në një relacion më vete ku mbajnë ID e punonjësit të cilin ato referojnë dhe atributet e tyre specifike.

Bazuar në relacionet e mësiperme ndërtohet dhe modeli relacional si më poshtë :

ENTITETE -> Relacion

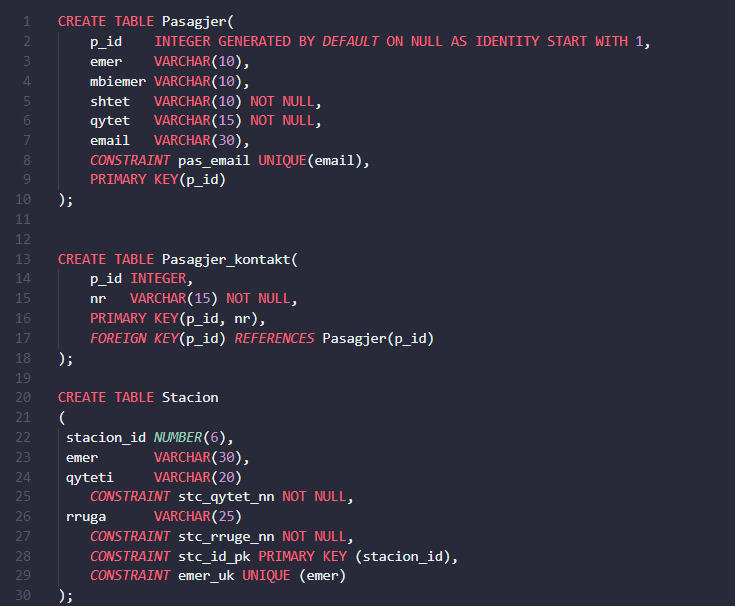
1. Stacion(ID,Emer,Qyteti,Rruga)
2. Abonim(ID,Cmimi,Klasa\_E\_Vendeve,Numri\_I\_Udhetimeve)
3. Pasagjer(ID,Emer,Mbiemer,Shteti,Qyteti,Email)
4. Pasagjer\_Kontakt(ID\_Pasagjeri, Numer\_Telfoni)
5. Punonjes(ID,Emer,Mbiemer,Data\_E\_Lindjes,Data\_E\_Punesimit,Oret\_E\_Punes,Pozicioni,Gjinia,Email,Qyteti,Shteti, Stacion\_ID)
6. Punonjes\_Kontakt(ID, Numer\_Telefoni)
7. Udhetim(ID,Data,Koha\_E\_Nisjes,Koha\_E\_Mberritjes,Stacion\_Nisje\_ID,Stacion\_Destinacion\_ID,Treni\_ID)
8. Bilete(ID,Klasa, Data\_E\_Blerjes,Cmimi,Pasagjer\_ID,Udhetim\_ID)
9. Treni(ID,Modeli,Statusi,Numer\_Vendesh,Date\_Prodhimi,Shpejtesia\_Maksimale,Numer\_Vagonash,Shofer\_ID,Konduktor\_ID)
10. Ngjyra\_E\_Trenit(ID, Ngjyra)
11. Menaxher(ID,Stacion\_ID)
12. Inxhinier(ID)
13. Shofer(ID,certifikate\_id)
14. Konduktor(ID,gcse\_anglisht\_niveli)

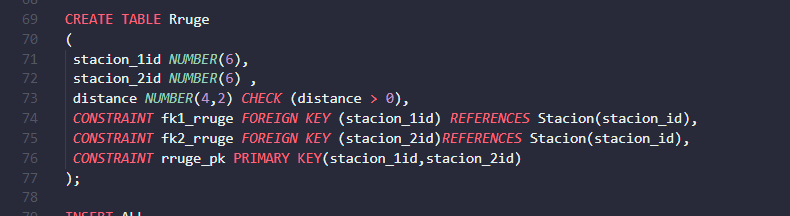
RELATIONSHIPS -> Relacion

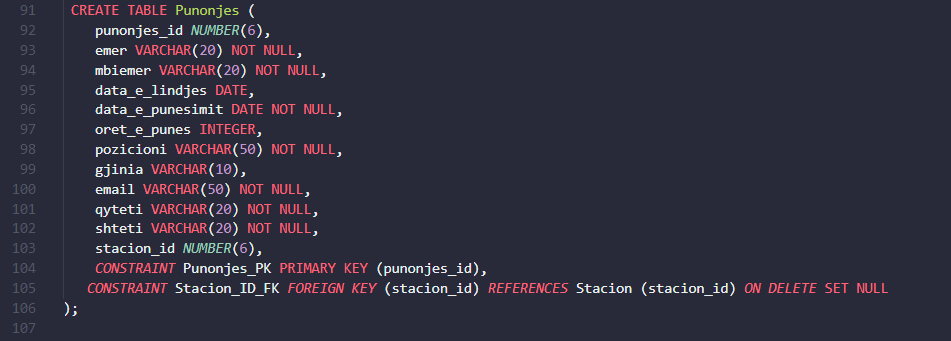
1. [Lidhen] Rruge(Stacion\_1ID,Stacion\_2ID,Distance)
2. [Abonohet] Abonim\_Pasagjeri(Abonim\_ID,Pasagjer\_ID,Data\_Skadences)

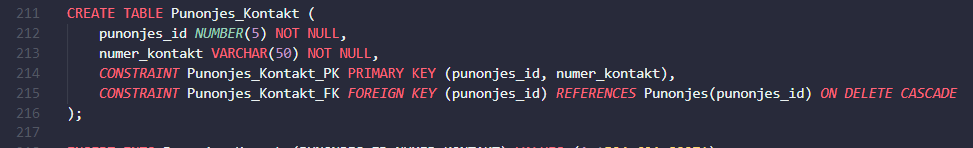
***Krijimi I Bazës Së Të Dhënave***

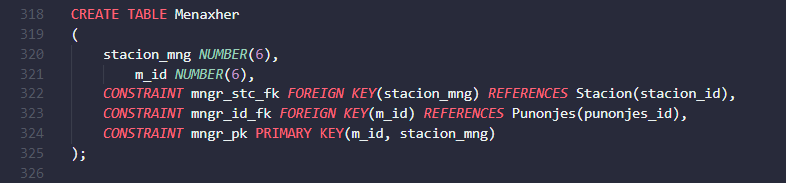
Bazuar në diagramën ER/EER dhe skemën relacionale, bëjmë shtimin e të dhënave në databazë:

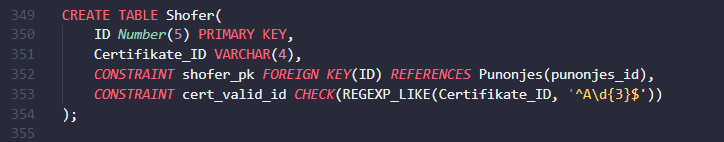


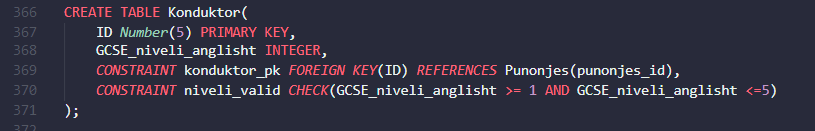


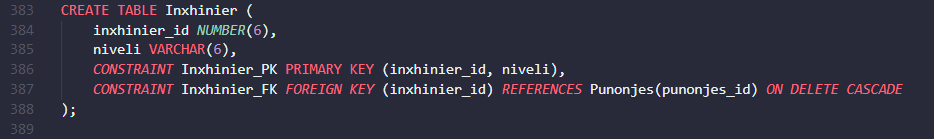


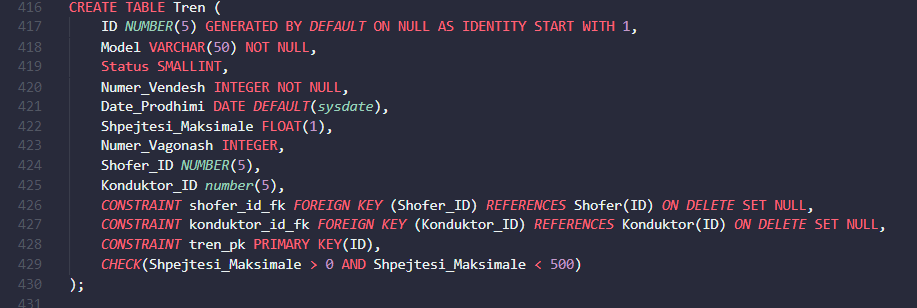


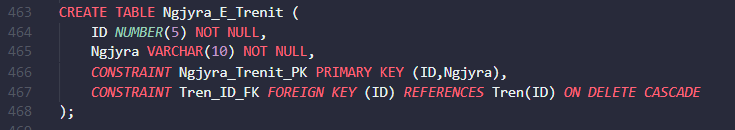


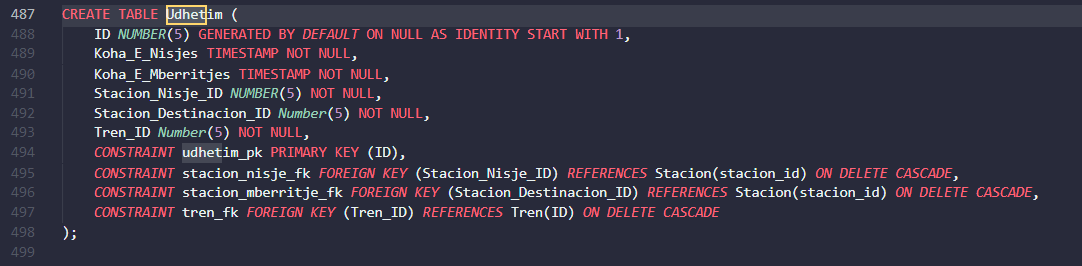


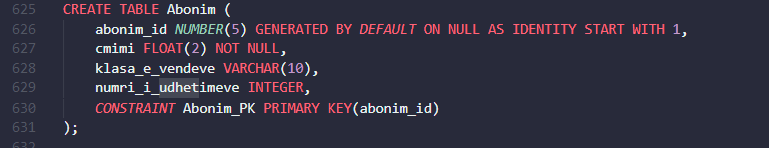


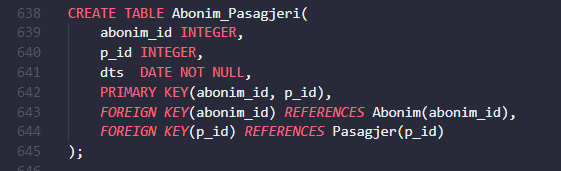


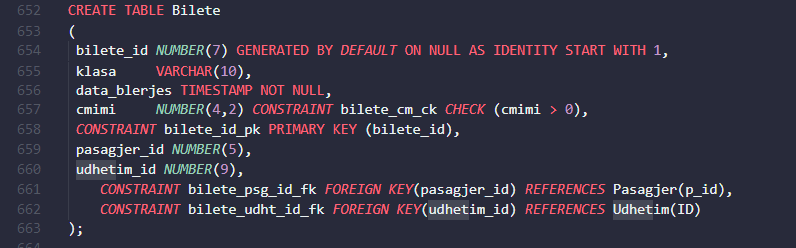












***Query***

**ALBI TAULLA**

*HISTOGRAME PER CMIMET E BILETAVE*

SELECT cmimi,COUNT(\*)

FROM Bilete

GROUP BY cmimi

ORDER BY 1;

*PUNONJESIT QE KANE DALE NE PENSION OSE JANE 6 MUAJ PARA DALJES NE PENSION*

SELECT punonjes\_id,emer,mbiemer,age

FROM(

SELECT punonjes\_id,emer,mbiemer,gjinia,data\_e\_lindjes,

CASE WHEN EXTRACT(month FROM data\_e\_lindjes) < EXTRACT(month FROM *SYS DATE*) OR

(EXTRACT(month FROM data\_e\_lindjes) = EXTRACT(month FROM *SYSDATE*)

AND (EXTRACT(day FROM data\_e\_lindjes) <= EXTRACT(day FROM *SYSDATE*)))

THEN EXTRACT(year from *sysdate*) - EXTRACT(year FROM data\_e\_lindjes)

ELSE EXTRACT(year from *sysdate*) - EXTRACT(year FROM data\_e\_lindjes) - 1 END AS  age

FROM Punonjes

)

WHERE (gjinia = 'M' AND (AGE >= 64 OR (AGE = 63 AND EXTRACT(month FROM

data\_e\_lindjes) - EXTRACT(month FROM *SYSDATE*) <=6 )))

     OR (gjinia = 'F' AND (AGE >= 62 OR (AGE = 61 AND EXTRACT(month FROM

data\_e\_lindjes) - EXTRACT(month FROM *SYSDATE*) <=6 )));

*LISTA E PUNONJESVE NE CDO STACION*

SELECT P.punonjes\_id,P.emer,p.mbiemer,P.pozicioni,S.emer

FROM Punonjes P INNER JOIN Stacion S ON P.stacion\_id = S.stacion\_id

WHERE S.stacion\_id = 1;

*UPDATIMI I SHOFERIT APO KONDUKTORIT TE NJE TRENI*

UPDATE Tren

SET Shofer\_id =17

WHERE ID = 9;

UPDATE Tren

SET Konduktor\_id = 6

WHERE ID = 10;

*UPDATIMI I KOHES SE NISJES DHE MBERRITJES PER NJE UDHETIM*

*UPDATE Udhetim*

*SET Koha\_E\_Nisjes = X,Koha\_E\_Mberritjes = Y*

*WHERE ID = 5;*

*FSHIRJA E TRENAVE ME STATUS PASIV APO NJE TRENI ME ID TE CAKTUAR*

*DELETE FROM Tren*

*WHERE Status = 0;*

*DELETE FROM Tren*

*WHERE ID = 1;*

*FSHIRJA E NJE UDHETIMI ME ID TE DHENE*

*DELETE FROM Udhetim*

*WHERE ID = 1;*

*MUAJI ME FREKUENCEN ME TE LARTE TE UDHETIMEVE NE VITIN 2014*

SELECT month,COUNT(\*) AS frequency

FROM (

    SELECT TO\_CHAR(CAST(Koha\_E\_Nisjes AS DATE),'MONTH') AS month

    FROM Udhetim

    WHERE EXTRACT(year FROM CAST(Koha\_E\_Nisjes AS DATE)) = 2014

)

GROUP BY month

ORDER BY frequency DESC

FETCH FIRST 1 ROWS ONLY;

*UDHETIMET MES DY STACIONEVE QE NUK KANE KALUAR ENDE*

SELECT ID,TO\_CHAR(Koha\_E\_Nisjes, 'MONTH DD, YYYY "at" HH24:MI') Nisja,

TO\_CHAR(Koha\_E\_Mberritjes, 'MONTH DD, YYYY "at" HH24:MI') Mberritja

FROM Udhetim

WHERE Stacion\_Nisje\_ID = 1 AND Stacion\_Destinacion\_ID =4 AND

EXTRACT(year FROM Koha\_E\_Nisjes) = EXTRACT(year FROM *SYSDATE*) AND

EXTRACT(month FROM Koha\_E\_Nisjes) >= EXTRACT(month FROM *SYSDATE*) AND

EXTRACT(day FROM Koha\_E\_Nisjes) >= EXTRACT(day FROM *SYSDATE*);

*TE GJITHA BILETAT QE KA BLERE PASAGJERI ME NUMER X(3 NE RASTIN TONE)*

SELECT P.emer, B.bilete\_id,B.klasa,TO\_CHAR(B.Data\_Blerjes, 'MONTH DD, YYYY "AT" HH24:MI'),B.cmimi

FROM Pasagjer P INNER JOIN Bilete B ON P.p\_id = B.pasagjer\_id

WHERE P.p\_id = 3;

*KONDUKTORI I CDO TRENI*

SELECT K.ID ID\_Konduktori, P.emer,P.mbiemer, T.ID ID\_Treni, T.Model

FROM (Konduktor K INNER JOIN Punonjes P ON K.ID = P.punonjes\_id) INNER JOIN Tren T ON T.Konduktor\_ID = K.ID;

**IRIS DIZDARI**

*TE GJITHA ABONIMET E NJE PASAGJERI X*

SELECT abonim\_id, P.p\_id, emer, mbiemer

FROM Abonim\_Pasagjeri A, Pasagjer P

WHERE A.p\_id = 1 AND A.p\_id = P.p\_id

*MENAXHERI I NJE STACIONI*

SELECT P.punonjes\_id, P.emer, P.mbiemer

FROM Punonjes P

WHERE P.punonjes\_id IN(

SELECT M.m\_id

FROM Menaxher M

WHERE stacion\_mng IN (

SELECT S.stacion\_id

FROM Stacion S

WHERE S.emer = 'Glendale/19th Avenue'

)

)

*STACIONET E NJE TRENI*

SELECT U.ID, U.Stacion\_Nisje\_ID, S.emer

FROM Udhetim U, Stacion S

WHERE U.Stacion\_Nisje\_D = S.stacion\_id AND U.Treni\_id = 6;

*DITA QE JANE SHITUR ME SHUME BILETA*

SELECT data\_blerjes, COUNT(data\_blerjes)

FROM Bilete GROUP BY data\_blerjes

HAVING COUNT (data\_blerjes)=(

SELECT MAX(COUNT)

FROM (

SELECT data\_blerjes, COUNT(data\_blerjes) COUNT

FROM Bilete

GROUP BY data\_blerjes))

*ANULLIMI I ABONIMIT*

DELETE Abonim\_Pasagjeri A

WHERE A.dts = '14-MAR-2018';

*INSERT RRUGE MES DY STACIONEVE*

INSERT INTO Rruge VALUES(007, 009, 2.95);

*UPDATE PASAGJER*

UPDATE Pasagjer

SET qytet = 'Florence'

WHERE p\_id = 1

**ALBI NDROQI**

*PUNONJESI QE KA PUNUAR ME SHUME ORE PER NJE VIT*

SELECT emer, mbiemer, oret\_e\_punes\*5\*50 "Ore pune ne vit"

FROM Punonjes

WHERE oret\_e\_punes = (SELECT MAX(oret\_e\_punes)

FROM Punonjes);

*STACIONET QE LIDHEN DREJPERDREJT*

SELECT s.emer , s2.emer , distance || ' km' AS "Distanca"

FROM Stacion s, Rruge, Stacion s2

WHERE (s.stacion\_id = rruge.stacion\_1id)

AND (s2.stacion\_id = rruge.stacion\_2id);

*NUMRI I KONTAKTIT TE MENAXHERIT PER NJE STACION X*

SELECT numer\_kontakt

FROM Punonjes\_Kontakt

WHERE punonjes\_id IN (SELECT m\_id

FROM Menaxher

WHERE stacion\_mng = &id\_STACIONIT);

*PASAGJERI QE E KALON NUMRIN E UDHETIMEVE*

SELECT p.emer, p.mbiemer, ab.numri\_i\_udhetimeve

FROM Pasagjer p , Abonim\_Pasagjeri a, Abonim ab

WHERE (p.p\_id = a.p\_id)

AND (ab.abonim\_id = a.abonim\_id

AND ab.numri\_i\_udhetimeve < (SELECT COUNT(\*)

FROM Bilete b

WHERE b.pasagjer\_id = p.p\_id));

*LISTA E PASAGJEREVE QE I SKADON ABONIMI PER 1, 3, 6 MUAJ*

SELECT \*

FROM Abonim\_Pasagjeri

WHERE MONTHS\_BETWEEN(dts,*SYSDATE*) = 1;

SELECT \*

FROM Abonim\_Pasagjeri

WHERE MONTHS\_BETWEEN(dts,*SYSDATE*) = 3;

SELECT \*

FROM Abonim\_Pasagjeri

WHERE MONTHS\_BETWEEN(dts,*SYSDATE*) = 6;

*PASAGJERI QE KA UDHETUAR TE PAKTEN 3 HERE BRENDA NJE JAVE TE CAKTUAR*

SELECT p\_id, emer, mbiemer

FROM Pasagjer

GROUP BY p\_id, emer,mbiemer

HAVING ((SELECT COUNT(\*)

FROM Bilete

WHERE p\_id = pasagjer\_id

AND (Data\_Blerjes BETWEEN '11-AUG-20' AND '18-AUG-20'))) =3;

*FSHIRJE UDHETIMI*

DELETE FROM Udhetim

WHERE id = &ID\_UDHETIMI;

*LISTA E TRENAVE NE STATUS PASIV*

SELECT ID, Model, Status

FROM TREN

WHERE (Status=0);

**ERSA DOKA**

*NUMRI I BILETAVE NE NJE DITE PER NJE STACION*

SELECT CAST(B.Data\_Blerjes AS DATE) AS Date\_Blerje, COUNT(\*) as Sasia\_e\_Biletave

FROM Bilete B

INNER JOIN Udhetim U ON

B.UDHETIM\_ID = U.ID

WHERE U.STACION\_NISJE\_ID = 4

GROUP BY CAST(Data\_Blerjes AS DATE)

*UDHETIMET E TE GJITHE TRENAVE NE NJE ORAR X*

SELECT ID ID\_UDHETIMIT, KOHA\_E\_NISJES

FROM UDHETIM

WHERE CAST(KOHA\_E\_NISJES AS TIME) > '08.00.00 AM' AND CAST(KOHA\_E\_NISJES AS TIME) < '10.00.00 AM'

*PASAGJERI QE KA UDHETUAR ME SHUME NE NJE VIT X*

SELECT PASAGJER\_ID, COUNT(\*) SASIA

FROM BILETE

GROUP BY PASAGJER\_ID

HAVING COUNT(\*) = (

SELECT MAX(COUNT(\*))

FROM BILETE

WHERE EXTRACT(YEAR FROM DATA\_BLERJES) = '2020'

GROUP BY PASAGJER\_ID

)

*SASIA E PUNONJESVE QE KA PUNUAR PART TIME DHE FULL TIME PER CDO STATION*

SELECT STACION\_ID, count(ORET\_E\_PUNES) as FULL\_TIME

FROM PUNONJES

where ORET\_E\_PUNES = 4

GROUP BY STACION\_ID

ORDER BY STACION\_ID ASC

SELECT STACION\_ID, count(ORET\_E\_PUNES) as FULL\_TIME

FROM PUNONJES

where ORET\_E\_PUNES = 8

GROUP BY STACION\_ID

ORDER BY STACION\_ID ASC

*UPDATE/FSHIRJE/INSERT NE ELEMENTET E TABELAVE*

UPDATE PUNONJES

SET EMAIL = 'DUMMYMAIL@GMAIL.COM'

WHERE PUNONJES\_ID = 5

UPDATE PUNONJES\_KONTAKT

SET NUMER\_KONTAKT = '06888888888'

WHERE PUNONJES\_ID = 36

UPDATE ABONIM

SET CMIMI = 70

WHERE KLASA\_E\_VENDEVE = 'Student'

INSERT INTO ABONIM(ABONIM\_ID, CMIMI, KLASA\_E\_VENDEVE, NUMRI\_I\_UDHETIMEVE) VALUES (5, 10, 'OFERTE', 100)

DELETE ABONIM WHERE KLASA\_E\_VENDEVE = 'OFERTE'

*LISTA E KONDUKTOREVE PER TOP 3 TRENAT ME TE SHPEJTE*

SELECT ID "ID E TRENIT", SHPEJTESI\_MAKSIMALE, KONDUKTOR\_ID

FROM TREN

ORDER BY SHPEJTESI\_MAKSIMALE DESC

FETCH FIRST 3 ROWS ONLY

***VIEWS***

**albi taulla**

*TE GJITHE UDHETIMET QE REALIZOHEN NE DITEN E SOTME*

CREATE VIEW Udhetimet\_E\_Dites AS

    SELECT S1.emer || ' Qyteti: ' || S1.qyteti || ' Rruga: ' || S1.rruga Stacioni1,

    S2.emer || ' Qyteti: ' || S2.qyteti || ' Rruga: ' || S2.rruga AS Stacioni2,

    U.ID Treni,TO\_CHAR(U.Koha\_E\_Nisjes, 'MONTH DD, YYYY "at" HH24:MI') Nisja,

    TO\_CHAR(U.Koha\_E\_Mberritjes, 'MONTH DD, YYYY "at" HH24:MI') Mberritja

    FROM Udhetim U

INNER JOIN Stacion S1 ON U.Stacion\_Nisje\_ID = S1.stacion\_id

INNER JOIN Stacion S2 ON U.Stacion\_Destinacion\_ID = S2.stacion\_id

    WHERE EXTRACT(year FROM U.Koha\_E\_Nisjes) = EXTRACT(year FROM *SYSDATE*)

AND

    EXTRACT(month FROM U.Koha\_E\_Nisjes) >= EXTRACT(month FROM *SYSDATE*) AND

    EXTRACT(day FROM U.Koha\_E\_Nisjes) >= EXTRACT(day FROM *SYSDATE*);

*NUMRI I UDHETIMEVE QE KA KRYER CDO TREN*

CREATE VIEW Udhetimet\_Per\_Tren AS

    SELECT T.ID,T.Model,COUNT(\*) frequency

    FROM Tren T INNER JOIN Udhetim U ON T.ID = U.Tren\_ID

    GROUP BY T.ID,T.Model

    ORDER BY frequency DESC;

***iris dizdari***

*MESATARJA E PASAGJEREVE QE UDHETOJNE ME TREN PER X MUAJ*

CREATE VIEW Bilete\_Pasagjer AS

SELECT AVG(Bilete.bilete\_id)

FROM Pasagjer, Bilete

WHERE Bilete.data\_blerjes < '01-JUL-2020'

*TE GJITHE PASAGJERET QE NUK KANE UDHETUAR ASNJEHERE ME NJE TREN X*

CREATE VIEW Pasagjer\_Tren AS

SELECT P.p\_id, P.emer, P.mbiemer

FROM Pasagjer P

WHERE P.p\_id NOT IN(

SELECT DISTINCT B.pasagjer\_id

FROM Bilete B, Udhetim U

WHERE B.udhetim\_id = U.ID AND U.Tren\_ID = 9

)

**ALBI NDROQI**

*MENAXHERET E STACIONIT 1*

CREATE VIEW Menaxher\_Stacion\_1 AS

SELECT p.emer, p.mbiemer

FROM Punonjes p

WHERE p.punonjes\_id IN(SELECT punonjes\_id

FROM Punonjes\_Kontakt

WHERE punonjes\_id IN (SELECT m\_id

FROM Menaxher

WHERE stacion\_mng = 1));

*STACIONET ME DISTANCEN MAKSIMALE*

CREATE VIEW Stacion\_Distance\_MAX AS

SELECT s1.emer AS "Emri Stacionit 1", s2.emer AS "Emri Stacionit 2"

FROM Stacion s1, Stacion s2

WHERE s1.stacion\_id = (SELECT stacion\_1id

FROM Rruge

WHERE distance = (SELECT MAX(distance)

FROM Rruge))

AND s2.stacion\_id = (SELECT stacion\_2id

FROM Rruge

WHERE distance = (SELECT MAX(distance)

FROM Rruge));

**ERSA DOKA**

*BILETAT E CDO PASAGJERI*

CREATE VIEW BILETAT\_E\_CDO\_PASAGJERI AS

SELECT PASAGJER\_ID, COUNT(\*) SASIA\_E\_BILETAVE

FROM BILETE

GROUP BY PASAGJER\_ID

ORDER BY SASIA\_E\_BILETAVE DESC

*I PUNESUARI I PARË*

CREATE VIEW I\_PUNESUARI\_I\_PARE AS

    SELECT \*

    FROM PUNONJES

    WHERE DATA\_E\_PUNESIMIT = (

        SELECT MIN(DATA\_E\_PUNESIMIT)

        FROM PUNONJES

    )

***INDEXET***

**ALBI TAULLA**

CREATE INDEX Stacion\_ID\_Index ON Stacion(stacion\_id);

CREATE INDEX Tren\_ID\_Index ON Tren(ID);

**ALBI NDROQI**

CREATE INDEX BILETE\_DB\_INDX ON Bilete(Data\_Blerjes);

**ERSA DOKA**

CREATE INDEX PUNONJES\_ID\_INDEX ON PUNONJES(PUNONJES\_ID)