REPUBLIKA E SHQIPËRISË

UNIVERSITETI "ALEKSANDËR MOISIU" DURRËS

FAKULTETI I TEKNOLOGJISË SË INFORMACIONIT

DEPARTAMENTI I S HKENCAVE KOMPJUTERIKE

Adresa: Lagja Nr.1, Rr "Taulantia", Durrës, Tel www.uamd.edu.al

PROGRAMI I STUDIMIT: Shkenca kompjuterike

LËNDA: Magazinim Informacioni dhe Kërkim Njohurish

PEDAGOGU/JA : Edjola Naka

Pjesa II – DATA MINING

Punuar nga:

ENRIKETA KRRIKU

Objektivat :

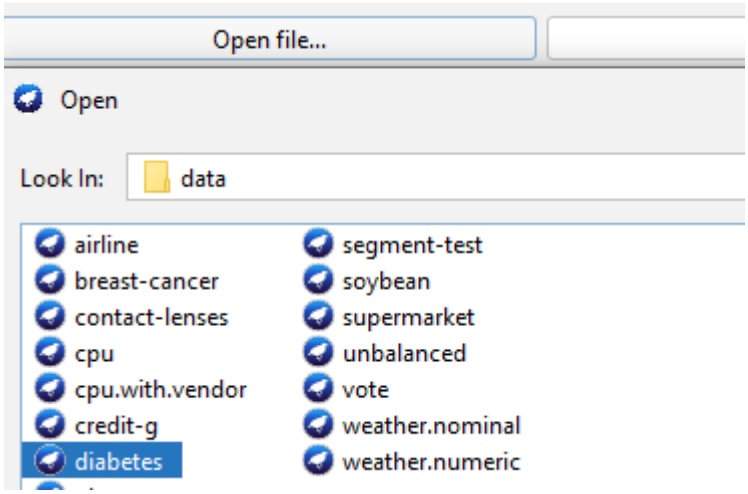
DataSet – Diabeti (marre nga Folderi data tek Weka)

Algoritmi – Decision Tree (Pemet e Vendimit)

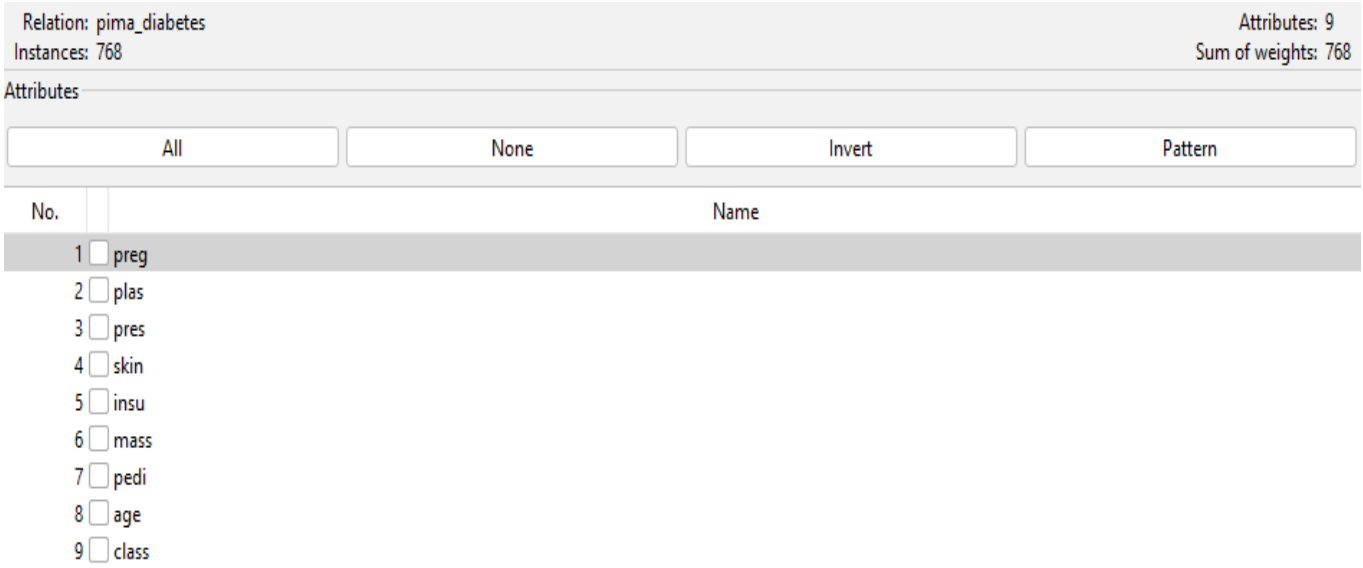
1) Njohja me Data Setin e Perdorur dhe Algoritmin
2) Ndarja e DataSetit ne 2 pjese (Trajnim/Testim)
3) Trajnimi I Algoritmit ne DataSetin Trajnues
4) Vlersimi I performances se Algoritmit ne DataSetin Test

1) Njohja me Data Setin e Perdorur dhe Algoritmin

- Ne fillim hapim programin Weka (Explorer) ku me pas uplodojme Datasetin e plote



Pasi e hapim datasetin shohim numerin e instancave(rreshtave) dhe atributeve (kolonat)



Ku kemi 768 rreshta dhe 9 kolona ku kolonat jane :

Preg – Pregrant (Shtatezane)

Plas – Glucose

Pres – Blood Pressure (Presioni gjakut)

Skin – Skin Thickness (Trashesia e lekures)

Insu – Insulin

Mass – BMI

Pdei – DiabetesPedigreeFunction

Age – Mosha

class – Klasa(na tregon se sa jane me diabet dhe jo me diabet)

Ku secila nga kolonat kane nje Minimum, nje Maksimum , nje Mesatare(Mean) dhe Devijimin standart(stdDev)

Selected attribute

Name: age Type: Numeric
Missing: 0 (0%) Distinct: 52 Unique: 5 (1%)

Statistic	Value
Minimum	21
Maximum	81
Mean	33.241
StdDev	11.76

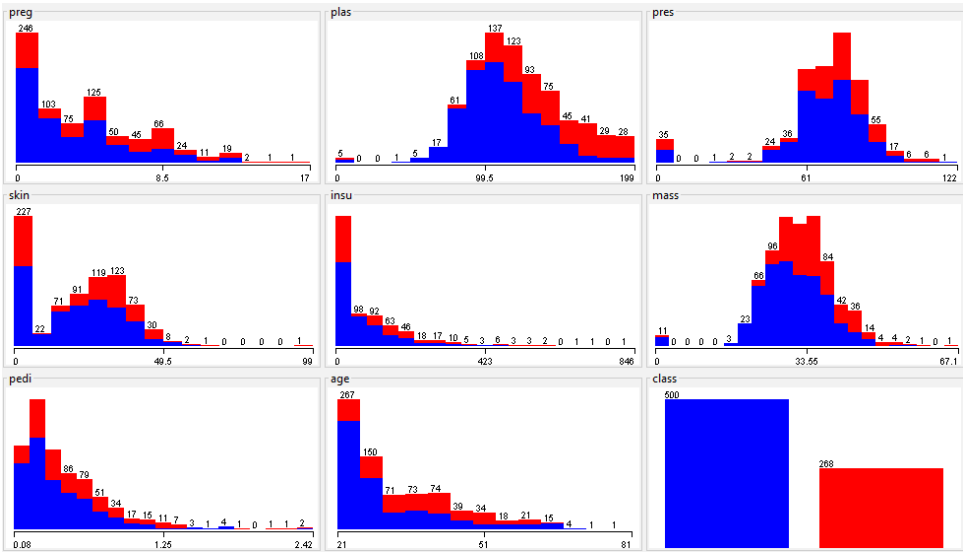
(pervec klases Diabetes I cila mban vlerat positive dhe negatie se kush eshte me diabet ose jo)

Selected attribute

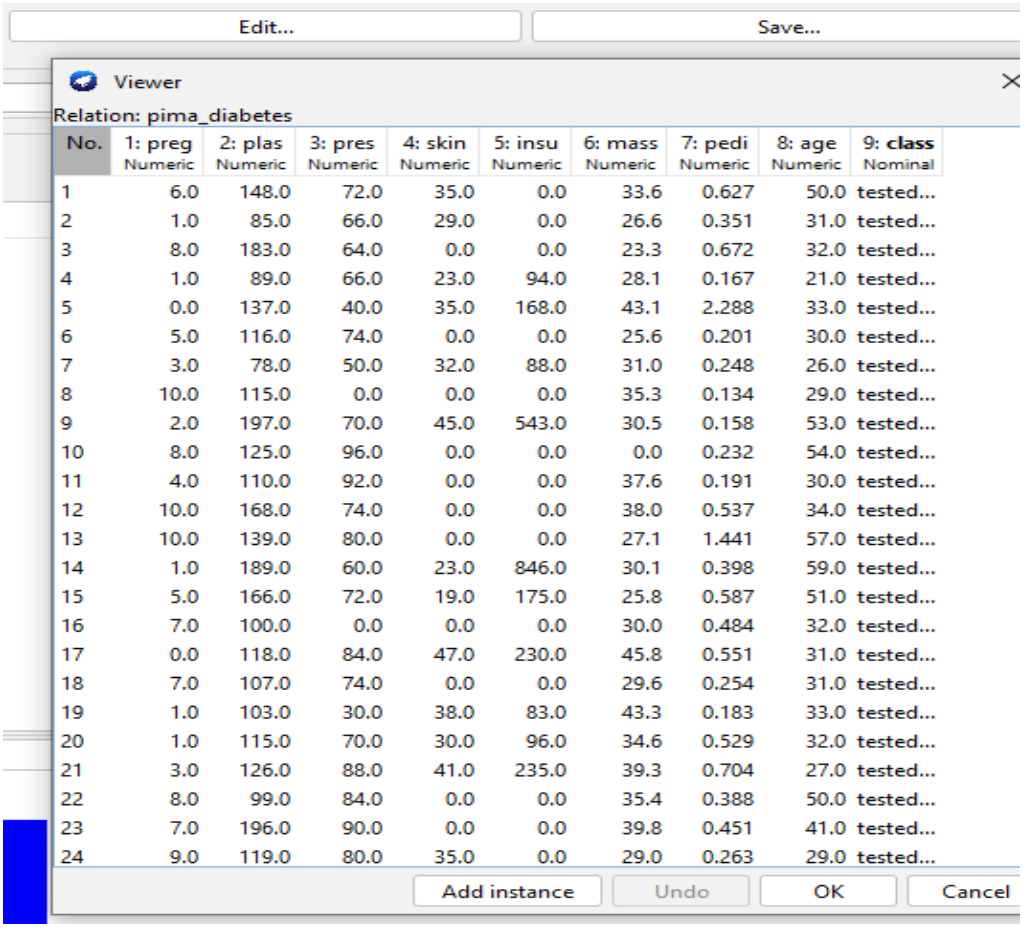
Name: class Type: Nominal
Missing: 0 (0%) Distinct: 2 Unique: 0 (0%)

No.	Label	Count	
1	tested_negative	500	500
2	tested_positive	268	268

Poshte ne figure kemi e gjithat atributet e vizualizuara



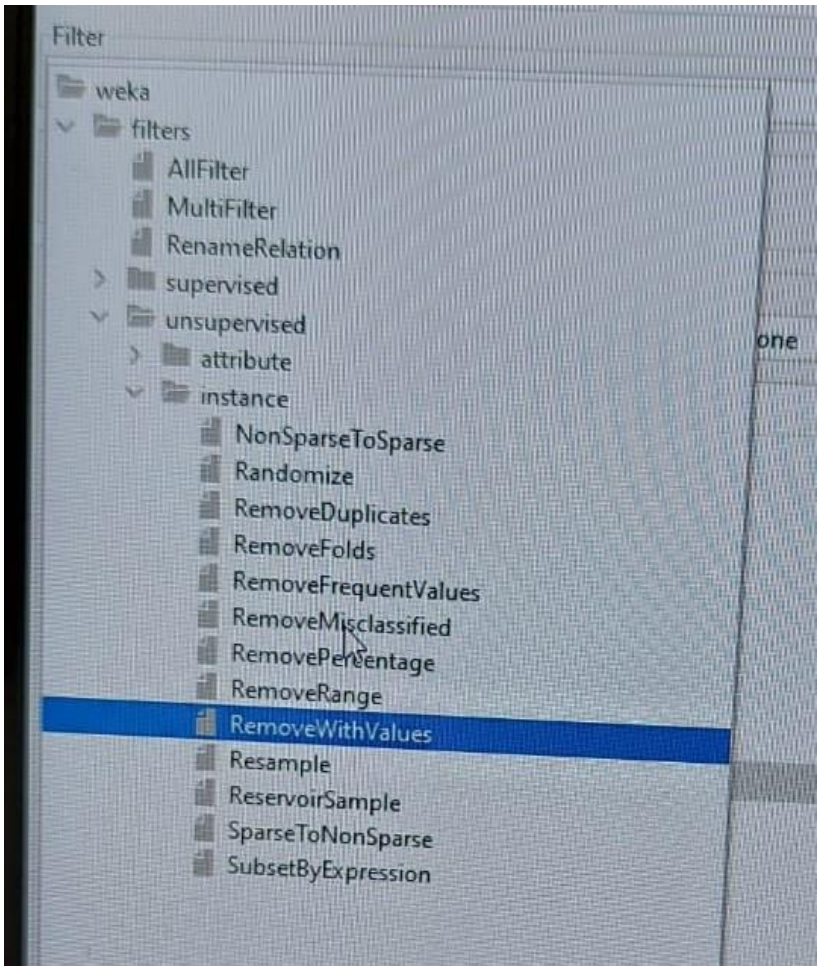
Por mund edhte ti shohim te gjithat datat per secilin rreshte e kolone nepermjet Edit :



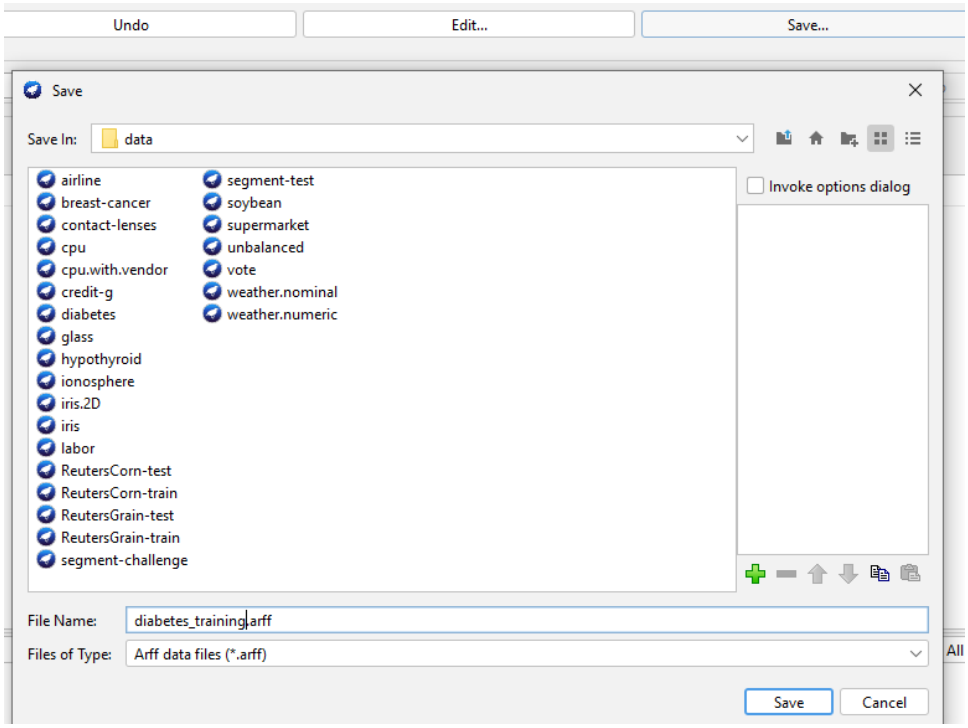
No.	1: preg Numeric	2: plas Numeric	3: pres Numeric	4: skin Numeric	5: insu Numeric	6: mass Numeric	7: pedi Numeric	8: age Numeric	9: class Nominal
1	6.0	148.0	72.0	35.0	0.0	33.6	0.627	50.0	tested...
2	1.0	85.0	66.0	29.0	0.0	26.6	0.351	31.0	tested...
3	8.0	183.0	64.0	0.0	0.0	23.3	0.672	32.0	tested...
4	1.0	89.0	66.0	23.0	94.0	28.1	0.167	21.0	tested...
5	0.0	137.0	40.0	35.0	168.0	43.1	2.288	33.0	tested...
6	5.0	116.0	74.0	0.0	0.0	25.6	0.201	30.0	tested...
7	3.0	78.0	50.0	32.0	88.0	31.0	0.248	26.0	tested...
8	10.0	115.0	0.0	0.0	0.0	35.3	0.134	29.0	tested...
9	2.0	197.0	70.0	45.0	543.0	30.5	0.158	53.0	tested...
10	8.0	125.0	96.0	0.0	0.0	0.0	0.232	54.0	tested...
11	4.0	110.0	92.0	0.0	0.0	37.6	0.191	30.0	tested...
12	10.0	168.0	74.0	0.0	0.0	38.0	0.537	34.0	tested...
13	10.0	139.0	80.0	0.0	0.0	27.1	1.441	57.0	tested...
14	1.0	189.0	60.0	23.0	846.0	30.1	0.398	59.0	tested...
15	5.0	166.0	72.0	19.0	175.0	25.8	0.587	51.0	tested...
16	7.0	100.0	0.0	0.0	0.0	30.0	0.484	32.0	tested...
17	0.0	118.0	84.0	47.0	230.0	45.8	0.551	31.0	tested...
18	7.0	107.0	74.0	0.0	0.0	29.6	0.254	31.0	tested...
19	1.0	103.0	30.0	38.0	83.0	43.3	0.183	33.0	tested...
20	1.0	115.0	70.0	30.0	96.0	34.6	0.529	32.0	tested...
21	3.0	126.0	88.0	41.0	235.0	39.3	0.704	27.0	tested...
22	8.0	99.0	84.0	0.0	0.0	35.4	0.388	50.0	tested...
23	7.0	196.0	90.0	0.0	0.0	39.8	0.451	41.0	tested...
24	9.0	119.0	80.0	35.0	0.0	29.0	0.263	29.0	tested...

2) **Ndarja e DataSetit ne 2 pjese (Trajnim/Testim)**

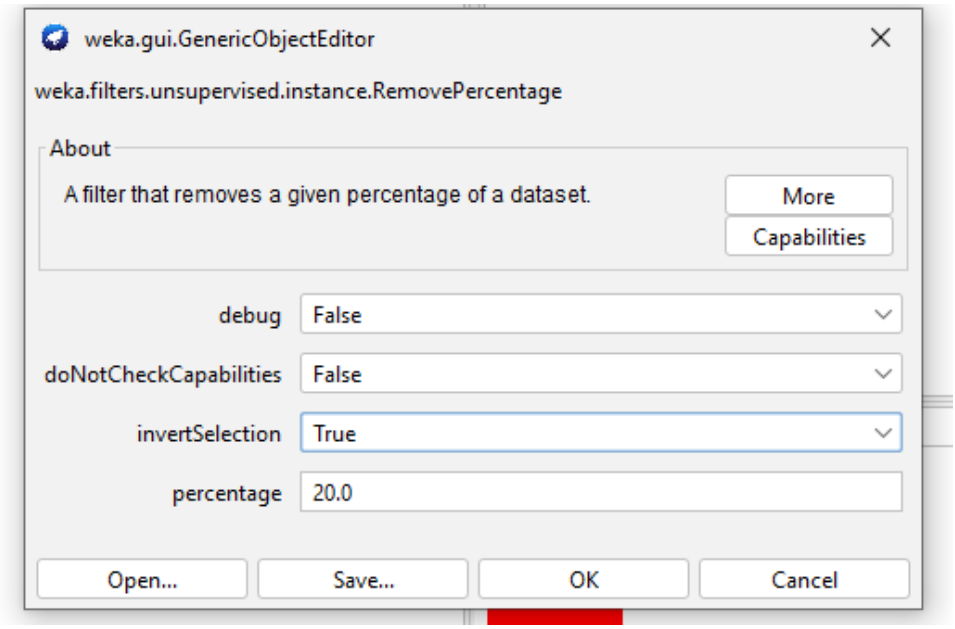
- Klikojme choose me pas filters , me pas ne unsupervized kemi 2 opsione atribute/instanca . Meqenese do te heqim nje % te datasetit qe ta perdorim per training pra zgjedhim instancen (rreshtat) pas saj zgjedhim funksionin RemovePercentage .



Me pas klikojme tek 50.0 dhe e vendosim percentage 20 qe do the thote qe do te heqim 20 % te rreshatve nga numri total I rreshtave (768) , pasi bejme aplyy shohim se nr I rreshtave nga 768 ka shkuar ne 614 , tani ne kemi datasetin per te trajnuar algoritmin. E ruajme datasetin e ri me emrin diabetes data_training.arff ne desktop



Me pas I japim Undo dhe bejme perseri te njejtin porces por bejme inverseSelection True qe te heqim 80 % nga dataseti diabetes

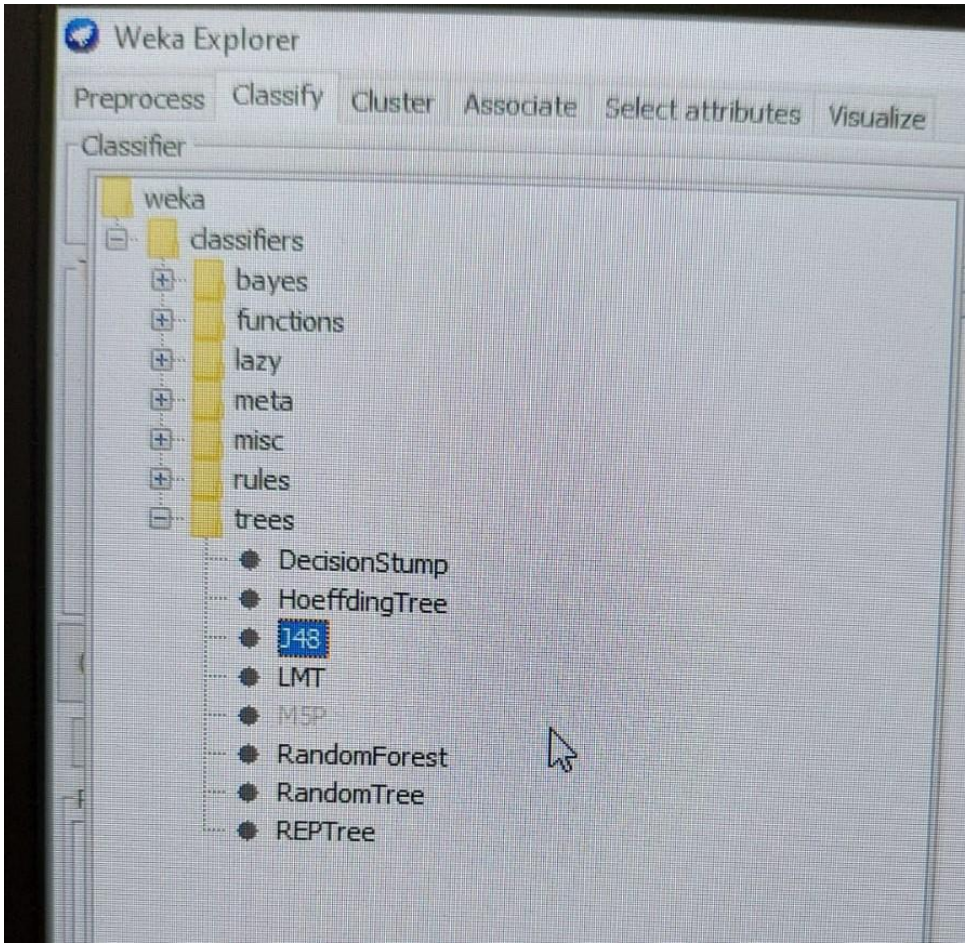


Ku na mbetet dataseti I cili do e perdorim per Testim (154 rreshta) ku perseri kete dataset e ruajme ne desktop me emrin Diabetes data_test.arff

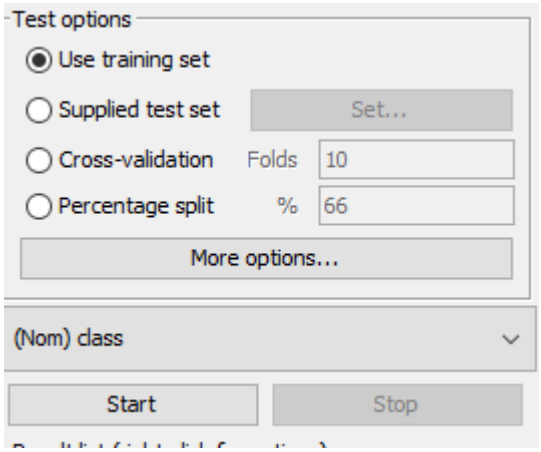
Pasi beme ndarjen e Datasetit tani kalojme ne fazen tjeter

3) **Trajnimi I Algoritmit ne DataSetin Trajnues**

Klikojme ne Classify , Choose , trees , zgjedhim J85 (Algoritmi I Pemes te Vendimit)



Ku me pas ne Test Options zgjedhim radiobutonin Use training test dhe ne more options Output predictions klikojme choose dhe zgjedhim CSV e klokojme START



Ku ne Console ne te djathte do na shfaqet performanca te Training Datasetit

```
=== Summary ===

Correctly Classified Instances         546               88.9251 %
Incorrectly Classified Instances        68               11.0749 %
Kappa statistic                          0.754
Mean absolute error                      0.1665
Root mean squared error                  0.2886
Relative absolute error                 36.6611 %
Root relative squared error             60.5583 %
Total Number of Instances              614

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.925    0.178    0.907      0.925    0.916      0.754    0.939     0.950     tested_negative
                 0.822    0.075    0.854      0.822    0.838      0.754    0.939     0.892     tested_positive
Weighted Avg.    0.889    0.142    0.889      0.889    0.889      0.754    0.939     0.930

=== Confusion Matrix ===

   a   b   <-- classified as
 370  30 |   a = tested_negative
  38 176 |   b = tested_positive
```

Shohim se kemi te deklaruar numrin e gjetheve dhe madhesine e pemes

```
Number of Leaves  :     35

Size of the tree :      69
```

Ku 88.9 % jane instanca te cilat jane llogaritur sakte dhe 11.1 % jane instanca te llogaritura gabim te instancave te Datasetit Trajnues nga 614 instanca gjithesej 546 rreshat jane llogaritu sakte dhe 68 jane llogaritur gabim

E ne fund shohim ***MATRICEN KONFUZION***

numerin e Tp(Kur rreshti eshte positiv dhe algoritmi e llogarit si positiv) - 370 rreshta

Fp (kur algoritmi e klasifikon rrestat si negativ kur eshte positiv) - 30 rreshta

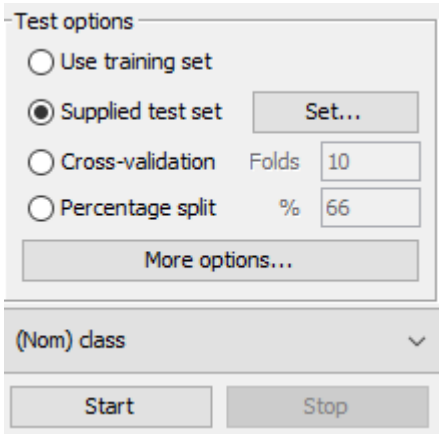
TN (kur algoritmi e klasifikon rrestat si negativ kur eshte negativ) - 176 rreshta

FN (kur algoritmi e klasifikon rrestat si positiv kur eshte negativ) - 38 rreshta

E gjithe kjo ka ndodhur per Datasetin Trajnues qe kemi trajnuar modelin tani kalojme ne fazen e Testimit

4) **Vlersimi I performances se Algoritmit ne DataSetin Test**

Ne Test Option ne te majte selectojme radiobutonin Supplied test set dhe klokojme Set, pas saj klokojme Open file dhe selektojme dataseting Test me emrin e caktuar dhe I japim close . Pas saj Kilokojme START.



Ne consloe shohim llogaritjen e Datasetit Test

```
=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.08 seconds

=== Summary ===

Correctly Classified Instances         106               68.8312 %
Incorrectly Classified Instances        48               31.1688 %
Kappa statistic                          0.3495
Mean absolute error                      0.3593
Root mean squared error                  0.4733
Relative absolute error                 77.9051 %
Root relative squared error             97.4348 %
Total Number of Instances              154

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.719    0.362    0.767      0.719    0.742      0.351    0.695     0.736     tested_negative
                 0.638    0.281    0.578      0.638    0.607      0.351    0.695     0.545     tested_positive
Weighted Avg.    0.688    0.332    0.696      0.688    0.691      0.351    0.695     0.664

=== Confusion Matrix ===

  a  b   <-- classified as
 69 27 |  a = tested_negative
 21 37 |  b = tested_positive
```

Ku 68.8 % jane instanca te cilat jane llogaritur sakte dhe 31.7 % jane instanca te llogaritura gabim te instancave te Datasetit Trajnues nga 154 instanca gjithesej 106 rreshat jane llogaritu sakte dhe 48 jane llogaritur gabim

E ne fund shohim ***MATRICEN KONFUZION***

numerin e Tp(Kur rreshti eshte positiv dhe algoritmi e llogarit si positiv) - 69 rreshta

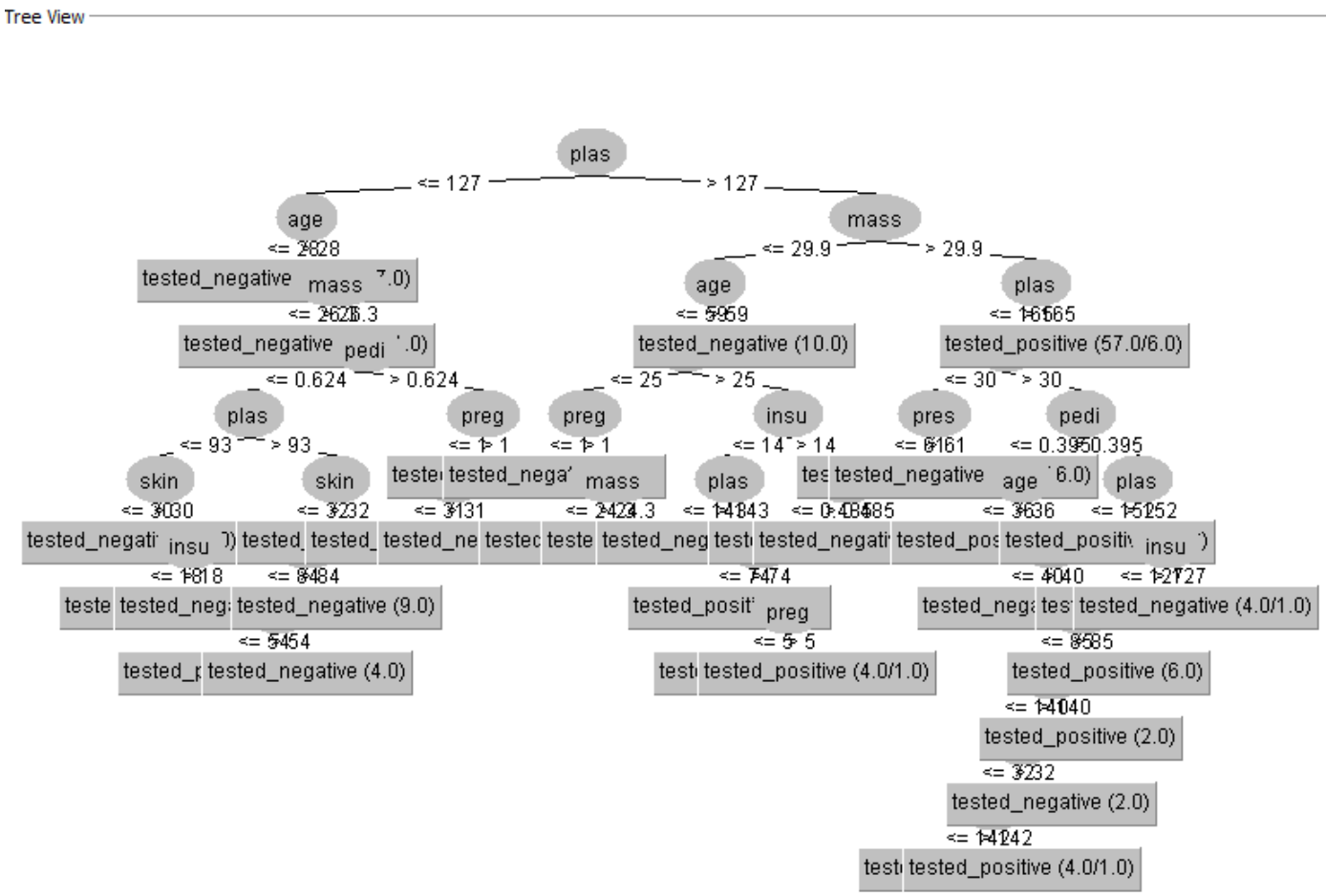
Fp (kur algoritmi e klasifikon rrestat si negativ kur eshte positiv) - 21 rreshta

TN (kur algoritmi e klasifikon rrestat si negativ kur eshte negativ) - 27 rreshta

FN (kur algoritmi e klasifikon rrestat si positiv kur eshte negativ) - 37 rreshta

E gjithe kjo ka ndodhur per Datasetin Testues qe kemi testuar modelin

Pamja e Pemes se testimit :



Rezultati :

Si fillim trajnimi I modelit kishte % te larte saktesie por gjate testimit shfaqi nje renie te larte te % te saktesise e cila ne fushen e mjeksise ky model ka probleme e nuk mund te implementohet.

(Ide per rritje ne % te saktesise se modelit : Mendoj se Data seti fillestare duhej te behej randomize)

DATA WAREHOUSE

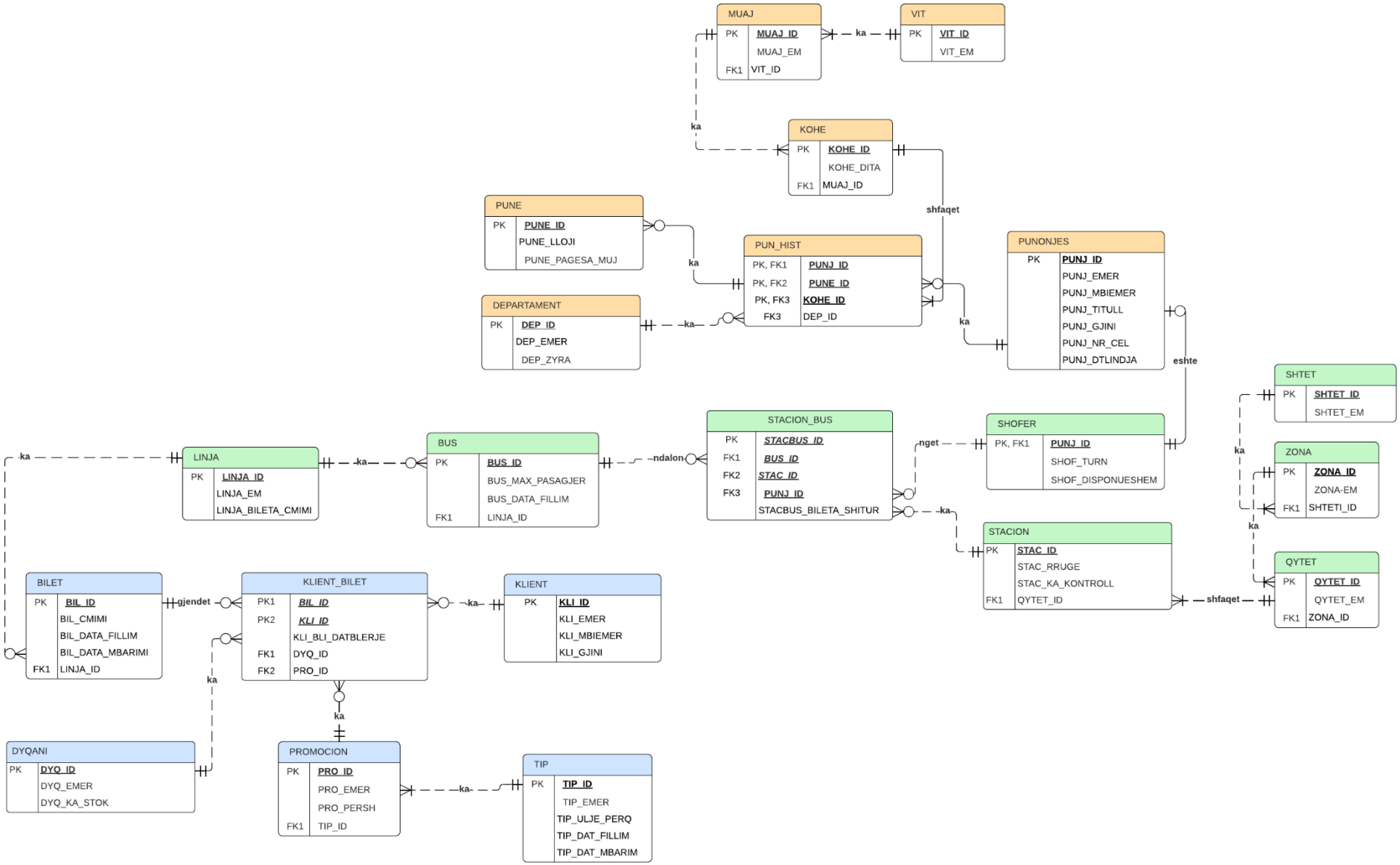
Organizata e Transportit kishte nevoje per dizenjimin e databazes per proceset e tyre. Dizenjimi Snowflake eshte bazuar ne 3 ngjarje biznesi qe ishin me me rendesi.

1) *Menaxhimi I punonjesve*
   Tek ngjarja e pare kemi tabelen PUN_HIST si tabele fakt me celes primar te perber nga entitet PUNE, PUNONJES, KOHE ku kjo e fundit eshte normalizuar dhe DEP_ID qe eshte nje celes I huaj per DEPARTAMENT. Ne kete menyre sigurohet qe nuk do kemi humbje te informacionit rreth stafit te kompanise.
2) *Transporti me autobusat*
   Tek ngjarja e dyte kemi sherbimin kryesor te kompanise qe eshte transporti me autobusa ku tabele fakt kemi STACION_BUS I cili permban celesat PK te entiteve LINJA, BUS, STACION dhe SHOFER qe trashegon PK nga tabela punonjes. Kjo skeme mundeson qe kompania te jete ne dije te mbarvatjes dhe gjendjes se sherbimit qe ofrojne.
3) *Pagese per sherbimin*
   Tek ngjarja e trete kemi skemen qe ndjek blerjen e biletave te transportit nga klientet. Tabela KLIENT_BILET eshte ajo fakt ku tabelat KLIENT, BILET, DYQANI, PROMOCIONI, TIP bejne te mundur te ruhet informacion I plote sesi klientet po nderverpojne me sherbimin dhe me ane te ciles mund te kryhet nje analize rreth fitimeve.

Skemat e 3 ngjarjeve jane te nderlidhura pasi pa nje nuk kemi tjetren. Lidhja e skemes se pare ndodh te tabela PUNONJES me tabelen SHOFER te skemes se dyt ku kemi trashegimin e celesit PK pasi SHOFER si entitet perfaqeson shoferet per te cilet nevojitet me shume informacion se pjesa tjeter e stafit. Entiteti LINJA lidhet me entitetin BILETE pasi kemi ndervaresi te psh. Cmimit te biletes ne baze te linjes qe klienti merr.

Me ane te kesaj skeme eshte perpjekur te jene mapuar realisht ngjarjet e kompanise se Transportit pa humbje te te dhenave dhe te jete sa me optimale per kryerjen e analizimeve per vendimarrje.

Ne vijim kemi skemen perfundimtare dhe kodin SQL per krijimin e databazes ne SQL Server.


MUAJ
PK MUAJ_ID
MUAJ_EM
FK1 VIT_ID
ka
VIT
PK VIT_ID
VIT_EM
ka
KOHE
PK KOHE_ID
KOHE_DITA
FK1 MUAJ_ID
shfaqet
PUNE
PK PUNE_ID
PUNE_LLOJI
PUNE_PAGESA_MUJ
ka
PUN_HIST
PK, FK1 PUNJ_ID
PK, FK2 PUNE_ID
PK, FK3 KOHE_ID
FK3 DEP_ID
DEPARTAMENT
PK DEP_ID
DEP_EMER
DEP_ZYRA
ka
ka
PUNONJES
PK PUNJ_ID
PUNJ_EMER
PUNJ_MBIEMER
PUNJ_TITULL
PUNJ_GJINI
PUNJ_NR_CEL
PUNJ_DTLINDJA
eshte
SHTET
PK SHTET_ID
SHTET_EM
ka
ZONA
PK ZONA_ID
ZONA-EM
FK1 SHTETI_ID
ka
QYTET
PK QYTET_ID
QYTET_EM
FK1 ZONA_ID
shfaqet
STACION
PK STAC_ID
STAC_RRUGE
STAC_KA_KONTROLL
FK1 QYTET_ID
ka
SHOFER
PK, FK1 PUNJ_ID
SHOF_TURN
SHOF_DISPONUESHEM
nget
STACION_BUS
PK STACBUS_ID
FK1 BUS_ID
FK2 STAC_ID
FK3 PUNJ_ID
STACBUS_BILETA_SHITUR
ndalon
BUS
PK BUS_ID
BUS_MAX_PASAGJER
BUS_DATA_FILLIM
FK1 LINJA_ID
ka
LINJA
PK LINJA_ID
LINJA_EM
LINJA_BILETA_CMIMI
ka
ka
BILET
PK BIL_ID
BIL_CMIMI
BIL_DATA_FILLIM
BIL_DATA_MBARIMI
FK1 LINJA_ID
gjendet
KLIENT_BILET
PK1 BIL_ID
PK2 KLI_ID
KLI_BLI_DATBLERJE
FK1 DYQ_ID
FK2 PRO_ID
ka
KLIENT
PK KLI_ID
KLI_EMER
KLI_MBIEMER
KLI_GJINI
ka
DYQANI
PK DYQ_ID
DYQ_EMER
DYQ_KA_STOK
ka
PROMOCION
PK PRO_ID
PRO_EMER
PRO_PERSH
FK1 TIP_ID
ka
TIP
PK TIP_ID
TIP_EMER
TIP_ULJE_PERQ
TIP_DAT_FILLIM
TIP_DAT_MBARIM

```
CREATE DATABASE BUS_SYSTEM_F;
USE BUS_SYSTEM_F

--NGJARJA 1: MENAXHIMI I PUNONJESVE
CREATE TABLE VIT(
VIT_ID INTEGER PRIMARY KEY,
VIT_EM SMALLINT NOT NULL,)

CREATE TABLE MUAJ (
MUAJ_ID INTEGER IDENTITY PRIMARY KEY,
MUAJ_EM VARCHAR(25) NOT NULL,
VIT_ID INTEGER NOT NULL REFERENCES VIT(VIT_ID),);

CREATE TABLE KOHE (
KOHE_ID DATE PRIMARY KEY,
KOHE_DITA VARCHAR(25) NOT NULL,
MUAJ_ID INTEGER NOT NULL REFERENCES MUAJ(MUAJ_ID),);

CREATE TABLE PUNE(
PUNE_ID int primary key,
PUNE_LLOJI varchar(25) NOT NULL,
PUNE_PAGESA_MUJ DECIMAL(14,4) NOT NULL,);

CREATE TABLE DEPARTAMENT(
DEP_IP int primary key,
DEP_EMER VARCHAR(30) NOT NULL,
```

```
DEP_ZYRA VARCHAR(30),)

CREATE TABLE PUNONJES(
PUNJ_ID int primary key,
PUNJ_EMER varchar(25) NOT NULL,
PUNJ_MBIEMER varchar(25) NOT NULL,
PUNJ_TITULL varchar(150) NOT NULL,
PUNJ_GJINI char(1) NOT NULL,
PUNJ_DITLINDJA date NOT NULL ,
PUNJ_NR char(13) NOT NULL,);

CREATE TABLE PUNE_HISTORIK(
PUNJ_ID INT NOT NULL,
PUNE_ID INT NOT NULL,
KOHE_ID DATE NOT NULL,
DEP_ID INT NOT NULL REFERENCES DEPARTAMENT(DEP_IP),
PRIMARY KEY (PUNJ_ID, PUNE_ID, KOHE_ID),
FOREIGN KEY (PUNJ_ID) REFERENCES PUNONJES(PUNJ_ID),
FOREIGN KEY (PUNE_ID) REFERENCES PUNE(PUNE_ID),
FOREIGN KEY (KOHE_ID) REFERENCES KOHE(KOHE_ID),
CONSTRAINT PU_HIST1 UNIQUE(PUNJ_ID, PUNE_ID, KOHE_ID));


--NGJARJA 2: SISTEMI I TRANSPORTIT
CREATE TABLE SHOFER
(PUNJ_ID  INT NOT NULL,
```

```
SHOF_TURN CHAR(1) NOT NULL,
SHOF_DISPONUESHEM CHAR(1) NOT NULL,
PRIMARY KEY (PUNJ_ID),
FOREIGN KEY (PUNJ_ID) REFERENCES PUNONJES(PUNJ_ID));

CREATE TABLE SHTET(
SHTET_ID INTEGER PRIMARY KEY,
SHTET_EM VARCHAR(25) NOT NULL,)

CREATE TABLE ZONA (
ZONA_ID INTEGER IDENTITY PRIMARY KEY,
ZONA_EM VARCHAR(25) NOT NULL,
SHTET_ID INTEGER NOT NULL REFERENCES SHTET(SHTET_ID),);

CREATE TABLE QYTET (
QYTET_ID INTEGER PRIMARY KEY,
QYTET_EM VARCHAR(25) NOT NULL,
ZONA_ID INTEGER NOT NULL REFERENCES ZONA(ZONA_ID),);

CREATE TABLE STACION (
STAC_ID SMALLINT PRIMARY KEY,
STAC_RRUGE VARCHAR(50) NOT NULL,
STAC_KA_KONTROLL BIT,
QYTET_ID INTEGER NOT NULL REFERENCES QYTET(QYTET_ID),);

CREATE TABLE LINJA(
```

```
LINJA_ID int primary key,
LINJA_EMER varchar(25) NOT NULL,
LINJA_BILETA_CMIMI DECIMAL(14,4) NOT NULL,);

CREATE TABLE BUS(
BUS_ID int primary key,
BUS_MAX_PASAGJER SMALLINT NOT NULL,
BUS_DATA_FILLIM DATE,
LINJA_ID INT NOT NULL REFERENCES LINJA(LINJA_ID));

CREATE TABLE STACION_BUS(
STACBUS_ID INTEGER PRIMARY KEY,
BUS_ID INT NOT NULL REFERENCES BUS(BUS_ID),
STAC_ID SMALLINT NOT NULL REFERENCES STACION(STAC_ID),
PUNJ_ID INT NOT NULL REFERENCES SHOFER(PUNJ_ID),
STACBUS_BILETA_SHITUR INTEGER NOT NULL, );


--NGJARJA 3: BLERJA E BILETAVE
CREATE TABLE BILETE(
BIL_ID int primary key,
BIL_CMIMI DECIMAL(14,2) NOT NULL,
BIL_DATA_FILLIMI DATE NOT NULL,
BIL_DATA_MBARIMI DATE NOT NULL,
LINJA_ID INT NOT NULL REFERENCES LINJA(LINJA_ID));
```

```
CREATE TABLE KLIENT(
KLI_ID int primary key,
KLI_EMER varchar(25) NOT NULL,
KLI_MBIEMER varchar(25) NOT NULL,
KLI_GJINI CHAR(1) NOT NULL);

CREATE TABLE TIP(
TIP_ID int primary key,
TIP_EMER varchar(25) NOT NULL,
TIP_ULJE_PERQ TINYINT NOT NULL,
TIP_DATA_FILLIMI DATE NOT NULL,
TIP_DATA_MBARIMI DATE NOT NULL,)

CREATE TABLE PROMOCION(
PRO_ID int primary key,
PRO_EMER varchar(25) NOT NULL,
PRO_PERSH varchar(25) NOT NULL,
TIP_ID INT NOT NULL REFERENCES TIP(TIP_ID));

CREATE TABLE DYQAN (
DYQ_ID INTEGER PRIMARY KEY,
DYQ_EMER VARCHAR(25) NOT NULL,
DYQ_KA_STOK BIT NOT NULL,);

CREATE TABLE KLIENT_BILET (
KLI_ID INTEGER NOT NULL REFERENCES KLIENT(KLI_ID),
```

```
BIL_ID INTEGER NOT NULL REFERENCES BILETE(BIL_ID),
KLI_BLI_DATBLERJE DATE NOT NULL,
DYQ_ID integer NOT NULL REFERENCES DYQAN(DYQ_ID),
PRO_ID INTEGER NOT NULL REFERENCES PROMOCION(PRO_ID),
PRIMARY KEY (KLI_ID, BIL_ID),
CONSTRAINT MSR1 UNIQUE(KLI_ID, BIL_ID),);
```