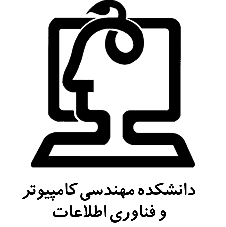
****

**گزارش پروژه آزمایشگاه پایگاه­داده**

گروه 2

**موضوع پروژه**

سیستم راه آهن

**ارائه دهنده­ها**

مهدیس صفری ، الهه رنجبری

تابستان 99

|  |  |
| --- | --- |
| فهرست مطالب | صفحه |

[**1- سیستم راه­آهن** 2](#_Toc52292809)

[**2- دسترسی ها** 2](#_Toc52292810)

[**3- تعریف کلی پروژه** 2](#_Toc52292811)

[**4- ERD** 5](#_Toc52292812)

[**5- پیادهسازی** 5](#_Toc52292813)

[**5-1- رویه ها:** 20](#_Toc52292814)

[**5-2- آغازگرها:** 26](#_Toc52292815)

[**5-3- توابع:** 28](#_Toc52292816)

[**5-4- پرس و جو:** 30](#_Toc52292817)

## **1- سیستم راه­آهن**

در این سیستم اطلاعات قطارها، ایستگاههای قطار، سفرها، خدمه قطار به ازای هر سفر و همچنین مسافران هر سفر موجود است.

کاربرانی از جمله مسافران، کارمندان راهآهن، مدیر راهآهن، رییس ایستگاه قطار و گروه برنامه­ریز سفرهای ریلی به سامانه دسترسی دارند که هر کدام سطح دسترسی­های متفاوتی که در ادامه توضیح داده خواهد شد دارند.

## **2- دسترسی­ها**

سطح دسترسی مسافران در حد مشاهده اطلاعات سفرها و اطلاعات خدمه مرتبط و رزرو یا کنسلی بلیط است.

دسترسی گروه برنامهریز سفرهای ریلی به مشاهده و همچنین تدوین و تغییر در جدولهای اطلاعات سفرها و خدمه مرتبط است.

دسترسی رئیس ایستگاههای قطار به جدول اطلاعات قطارها است.

کارمندان شرکت راهآهن به اطلاعات قطارها دسترسی دارند و میتوانند تغییرات لازم را اعمال کنند.

اما مدیر شرکت را آهن به تمام جداول دسترسی داشته و توانایی ایجاد تغییر در آنها را دارد.

## **3- تعریف کلی پروژه**

میتوان قطار جدید اضافه یا از رده خارج کرد. ایستگاههای جدیدی که ساخته میشوند افزوده شوند. سفرهای

برنامهریزی شده و تمام اطلاعات مربوط به آنها از جمله قطار مربوطه، گروه خدمات مرتبط با آن، ساعت و

تاریخ و مبدا و مقصد اضافه شوند. برای مسافران بلیط رزرو و یا کنسل کرد.

**جدول­ها:**

**1- قطار**

* شماره قطار primary key
* رنگ بدنه
* ظرفیت
* نوع : کوپه­ای / اتوبوسی
* سرعت : سریعالسیر / معمولی

**2- ایستگاه قطار**

* شماره ایستگاه primary key
* نام ایستگاه
* نام رییس ایستگاه
* شهر
* استان

**3- اطلاعات سفر**

* شماره سفر primary key
* شماره قطار foreign key
* مبدا foreign key
* مقصد foreign key
* ساعت حرکت
* ساعت رسیدن به مقصد
* تاریخ حرکت)فرض بر این است هر سفر حداکثر تا پایان همان روز به اتمام میرسد(
* قیمت بلیط
* ظرفیت باقی مانده
* تعداد درخواست

**4-ایستگاه سفر**

* شماره سفر
* شماره ایستگاه
* وضعیت )مبدا / مقصد(

شماره سفر، شماره ایستگاه Primary key/ foreign key

**5-اطلاعات خدمه قطار)کاربر نیست(**

* شماره خدمات سفر primary key
* شماره سفر foreign key
* نام لوکوموتیوران
* شرکت خدماترسانی
* نام رییس قطار

**6-بلیط**

* شماره سفر foreign key
* نام
* نام خانوادگی
* کد ملی primary key
* سن
* جنسیت

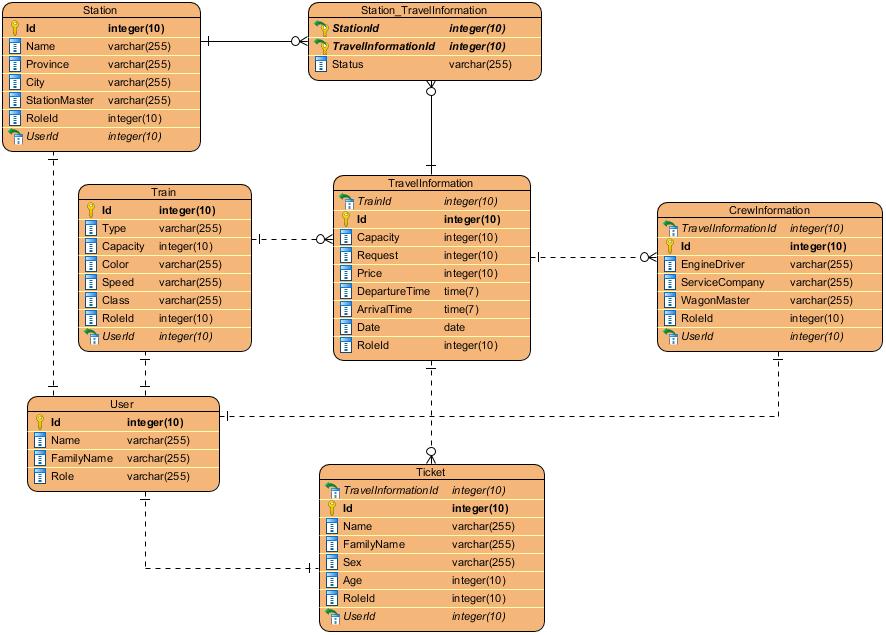
**7-کاربر**

* نام
* نام خانوادگی
* کد ملی primary key

**8-نقش**

* کد ملی کاربر foreign key
* شناسه نقش primary key
* عنوان نقش

## **4- ERD**



## **5- پیاده­سازی**

کدهای مربوط به ایجاد و مقداردهی جداول به شرح زیر است.

commit;

create table train(

id integer primary key,

type varchar(255),

capacity integer,

color varchar(255),

speed varchar(255),

class varchar(255))

create table travel(

id integer primary key,

train\_id integer REFERENCES train (id),

departure\_time time,

arrival\_time time,

date date,

capacity integer,

price integer,

request integer)

create table travel\_station(

travel\_id integer,

station\_id integer,

status varchar(255))

create table station(

id integer primary key,

name varchar(255),

province varchar(255),

city varchar(255),

station\_master varchar(255))

create table crew\_information(

id integer primary key,

travel\_id integer REFERENCEs travel(id),

engine\_driver varchar(255),

service\_company varchar(255),

wagon\_master varchar(255)

create table ticket(

id integer primary key,

travel\_id integer REFERENCEs travel(id),

name varchar(255),

family\_name varchar(255),

age integer,

sex varchar(255))

create table users( id integer primary key,

name varchar(255),

family\_name varchar(255),

role varchar(255))

insert into train values(

1000, 'coupe train', 250, 'black', '100km/h', 'A'),

(1001, 'coupe train', 300, 'black', '100km/h', 'A'),

(1002, 'bus train', 150, 'brown', '60km/h', 'C'),

(1003, 'coupe train', 250, 'red', '100km/h', 'A'),

(1004, 'coupe train', 180, 'black', '80km/h', 'B'),

(1005, 'bus train', 200, 'yellow', '100km/h', 'A'),

(1006, 'coupe train', 250, 'black', '80km/h', 'B'),

(1007, 'bus train', 300, 'white', '60km/h', 'C')

insert into travel values(

2000, 1000, '12:30:00', '22:35:00', '09/17/2020', 300, 1000000, 295),

(2001, 1001, '12:30:00', '22:35:00', '09/17/2020', 250, 8500000, 245),

(2002, 1000, '11:30:00', '18:40:00', '09/18/2020', 245, 9500000, 245),

(2003, 1002, '11:30:00', '21:55:00', '09/18/2020', 150, 700000, 145),

(2004, 1001, '13:40:00', '20:42:00', '09/18/2020', 250, 9500000, 255),

(2005, 1003, '09:35:00', '19:42:00', '09/18/2020', 250, 9550000, 235),

(2006, 1004, '11:15:00', '23:02:00', '09/19/2020', 180, 10500000, 177),

(2007, 1005, '16:35:00', '23:45:00', '09/19/2020', 200, 10000000, 200),

(2008, 1000, '14:15:00', '22:35:00', '09/19/2020', 250, 10000000, 250),

(2009, 1006, '13:35:00', '21:45:00', '09/20/2020', 250, 7000000, 251),

(2010, 1007, '08:16:00', '18:45:00', '09/20/2020', 300, 6000000, 280),

(2011, 1002, '09:25:00', '20:00:00', '09/20/2020', 150, 6500000, 150),

(2012, 1005, '13:20:00', '23:05:00', '09/20/2020', 200, 7000000, 200)

insert into station values(

3000, 'tehranStation', 'tehran', 'tehran', 'sadr'),

(3001, 'golab', 'shiraz', 'shiraz', 'rahimi'),

(3002, 'zayande rood', 'esfahan', 'shiraz', 'razi'),

(3003, 'emam reza', 'khorasan razavi', 'mashhad', 'siri'),

(3004, 'dolat abad', 'yazd', 'yazd', 'talebi'),

(3005, 'kavir', 'kerman', 'kerman', 'safari')

insert into travel\_station values(

2000, 3000, 'begining'),

(2000, 3001, 'destination'),

(2001, 3000, 'begining'),

(2001, 3003, 'destination'),

(2002, 3001, 'begining'),

(2002, 3002, 'destination'),

(2003, 3002, 'begining'),

(2003, 3003, 'destination'),

(2004, 3004, 'begining'),

(2004, 3000, 'destination'),

(2005, 3005, 'begining'),

(2005, 3001, 'destination'),

(2006, 3000, 'begining'),

(2006, 3002, 'destination'),

(2007, 3001, 'begining'),

(2007, 3004, 'destination'),

(2008, 3000, 'begining'),

(2008, 3005, 'destination'),

(2009, 3003, 'begining'),

(2009, 3001, 'destination'),

(2010, 3003, 'begining'),

(2010, 3001, 'destination'),

(2011, 3001, 'begining'),

(2011, 3002, 'destination'),

(2012, 3004, 'begining'),

(2012, 3000, 'destination')

insert into crew\_information values(

4000, 2000, 'baravarde', 'raja', 'kermani'),

(4001, 2001, 'sabai', 'raja', 'mana'),

(4002, 2002, 'rad', 'safir', 'jaberi'),

(4003, 2003, 'kaleydari', 'raja', 'javadi'),

(4004, 2004, 'mehrab', 'raja', 'rafat'),

(4005, 2005, 'sadr', 'raja', 'hani'),

(4006, 2006, 'sori', 'safir', 'fakhari'),

(4007, 2007, 'raha', 'raja', 'fadai'),

(4008, 2008, 'bagheri', 'raja', 'atri'),

(4009, 2009, 'tavana', 'safir', 'bozorgnia'),

(4010, 2010, 'ranjbari', 'raja', 'kamali'),

(4011, 2011, 'bahar', 'raja', 'shirazi'),

(4012, 2012, 'rafat', 'safir', 'sharifzade')

insert into ticket values(

5000, 2000, 'reza', 'rajai', 20, 'male'),

(5001, 2000, 'sina', 'souri', 35, 'male'),

(5002, 2000, 'mahdi', 'jaberi', 22, 'male'),

(5003, 2000, 'fateme', 'rezaei', 40, 'female'),

(5004, 2000, 'elahe', 'ranjbari', 12, 'female'),

(5005, 2000, 'mehrab', 'hakimi', 32, 'male'),

(5006, 2000, 'zahra', 'shirazi', 68, 'female'),

(5007, 2000, 'ali', 'zand', 77, 'male'),

(5008, 2000, 'kamran', 'ershadbakhsh', 31, 'male'),

(5009, 2000, 'hadi', 'safiei', 45, 'male'),

(5010, 2000, 'hoori', 'sabouri', 49, 'female'),

(5011, 2000, 'hamed', 'karkhane', 53, 'male'),

(5012, 2000, 'yaghma', 'mahroo', 18, 'female'),

(5013, 2000, 'reza', 'maleki', 12, 'male'),

(5014, 2000, 'haniye', 'sari', 26, 'female'),

(5015, 2001, 'mahdis', 'safari', 26, 'female'),

(5016, 2001, 'hassan', 'saafat', 41, 'male'),

(5017, 2001, 'saeed', 'rahro', 22, 'male'),

(5018, 2001, 'haniye', 'rahavi', 33, 'female'),

(5019, 2001, 'hoda', 'habibi', 41, 'female'),

(5020, 2001, 'rahele', 'keshavarz', 47, 'female'),

(5021, 2001, 'habib', 'borna', 66, 'male'),

(5022, 2001, 'kiana', 'siri', 58, 'female'),

(5023, 2001, 'vahid', 'vahdat', 17, 'male'),

(5024, 2001, 'mona', 'mohammadi', 16, 'female'),

(5025, 2002, 'mohsen', 'mohammadi', 16, 'male'),

(5026, 2002, 'sara', 'laknahoor', 46, 'female'),

(5027, 2002, 'arshiya', 'babai', 64, 'male'),

(5028, 2002, 'tiyam', 'moafi', 73, 'male'),

(5029, 2002, 'tina', 'sadat', 25, 'female'),

(5030, 2002, 'mahmood', 'mahmoodi', 23, 'male'),

(5031, 2003, 'ashkan', 'mahmoodi', 23, 'male'),

(5032, 2003, 'sadaf', 'saberi', 41, 'female'),

(5033, 2003, 'mahmood', 'mirzaei', 35, 'male'),

(5034, 2003, 'elnaz', 'zahiri', 55, 'female'),

(5035, 2003, 'emaad', 'bozorgi', 49, 'male'),

(5036, 2004, 'ziba', 'bozorgmehr', 26, 'female'),

(5037, 2004, 'behnam', 'hosseyni', 48, 'male'),

(5038, 2004, 'zakiye', 'baradar', 70, 'female'),

(5039, 2004, 'behzad', 'mohaf', 37, 'male'),

(5040, 2004, 'mansour', 'kabiri', 35, 'male'),

(5041, 2004, 'elham', 'bozorgzade', 55, 'female'),

(5042, 2005, 'radin', 'jaberi', 25, 'male'),

(5043, 2005, 'katrin', 'javadi', 55, 'female'),

(5044, 2005, 'mehrab', 'rafat', 46, 'male'),

(5045, 2005, 'sadr', 'hani', 35, 'male'),

(5046, 2005, 'sori','fakhari', 58, 'female'),

(5047, 2005, 'raha','fadai', 43, 'female'),

(5048, 2005, 'mana', 'sabai', 19, 'female'),

(5049, 2006, 'radin', 'jaberi', 35, 'male'),

(5050, 2006, 'katrin', 'javadi', 52, 'female'),

(5051, 2006, 'mehrab', 'rafat', 61, 'male'),

(5052, 2006, 'sadr', 'hani', 26, 'male'),

(5053, 2006, 'sori','fakhari', 33, 'female'),

(5054, 2006, 'raha','fadai', 45, 'female'),

(5055, 2006, 'mana', 'sabai', 17, 'female'),

(5056, 2007, 'radin', 'jaberi', 25, 'male'),

(5057, 2007, 'katrin', 'javadi', 55, 'female'),

(5058, 2007, 'mehrab', 'rafat', 46, 'male'),

(5059, 2007, 'sadr', 'hani', 35, 'male'),

(5060, 2007, 'sori','fakhari', 58, 'female'),

(5061, 2008, 'sahar', 'jaberi', 28, 'female'),

(5062, 2008, 'katrin', 'javadi', 58, 'female'),

(5063, 2008, 'mehrab', 'rafat', 43, 'male'),

(5064, 2008, 'sadr', 'hani', 38, 'male'),

(5065, 2008, 'sori','fakhari', 29, 'female'),

(5066, 2008, 'raha','fadai', 80, 'female'),

(5067, 2008, 'mana', 'sabai', 17, 'female'),

(5068, 2009, 'radin', 'jaberi', 36, 'male'),

(5069, 2009, 'katrin', 'javadi', 25, 'female'),

(5070, 2009, 'mehrab', 'rafat', 71, 'male'),

(5071, 2009, 'sadr', 'hani', 62, 'male'),

(5072, 2009, 'sori','fakhari', 51, 'female'),

(5073, 2010, 'radin', 'jaberi', 28, 'male'),

(5074, 2010, 'katrin', 'javadi', 36, 'female'),

(5075, 2010, 'mehrab', 'rafat', 41, 'male'),

(5076, 2010, 'sadr', 'hani', 66, 'male'),

(5077, 2010, 'sori','fakhari', 28, 'female'),

(5078, 2010, 'raha','fadai', 20, 'female'),

(5079, 2010, 'mana', 'sabai', 29, 'female'),

(5080, 2011, 'radin', 'jaberi', 25, 'male'),

(5081, 2011, 'katrin', 'javadi', 53, 'female'),

(5082, 2011, 'mehrab', 'rafat', 43, 'male'),

(5083, 2011, 'sadr', 'hani', 38, 'male'),

(5084, 2011, 'sori','fakhari', 28, 'female'),

(5085, 2011, 'raha','fadai', 41, 'female'),

(5086, 2011, 'mana', 'sabai', 39, 'female'),

(5087, 2012, 'radin', 'jaberi', 28, 'male'),

(5088, 2012, 'katrin', 'javadi', 25, 'female'),

(5089, 2012, 'mehrab', 'rafat', 46, 'male'),

(5090, 2012, 'sadr', 'hani', 35, 'male'),

(5091, 2012, 'sori','fakhari', 38, 'female'),

(5092, 2012, 'raha','fadai', 53, 'female'),

(5093, 2012, 'mana', 'sabai', 83, 'female'),

(5094, 2012, 'mana', 'sabai', 22, 'female')

insert into users values(

6000, 'elmira', 'jaberi', 'manager'),

(6001, 'ali', 'keramati', 'staff1'),

(6002, 'kasra', 'rajabi', 'staff2'),

(6003, 'linda', 'babaei', 'staff3'),

(6004, 'saman', 'sadr', 'station\_master1'),

(6005, 'golab', 'rahimi', 'station\_master2'),

(6006, 'roodabe', 'razi', 'station\_master3'),

(6007, 'reza', 'siri', 'station\_master4'),

(6008, 'emaad','talebi', 'station\_master5'),

(6009, 'sina','salari', 'programmer1'),

(6010, 'saba','rahmati', 'programmer2'),

(6011, 'mana','mahdavi', 'programmer3'),

(6012, 'asra','kaviri', 'programmer4'),

(6013, 'sahar','rbiei', 'programmer5'),

(6014, 'kave', 'rahavi', 'customer1'),

(6015, 'reza', 'rajai', 'customer2'),

(6016, 'sina', 'souri', 'customer3'),

(6017, 'mahdi', 'jaberi', 'customer4'),

(6018, 'fateme', 'rezaei', 'customer5'),

(6019, 'elahe', 'ranjbari', 'customer6'),

(6020, 'mehrab', 'hakimi', 'customer7'),

(6021, 'zahra', 'shirazi', 'customer8'),

(6022, 'ali', 'zand', 'customer9'),

(6023, 'kamran', 'ershadbakhsh', 'customer10'),

(6024, 'hoori', 'sabouri', 'customer11')

کدهای مربوط به ایجاد نقش­ها و تعیین دسترسی­ها به شرح زیر است.

create sequence auto\_id

owned by ticket.id;

alter table ticket

alter column id set default nextval('auto\_id');

create sequence auto\_id\_users

owned by users.id;

alter table users

alter column id set default nextval('auto\_id\_users');

create sequence auto\_id\_station

owned by station.id;

alter table station

alter column id set default nextval('auto\_id\_station');

create sequence auto\_id\_travel

owned by travel.id;

alter table travel

alter column id set default nextval('auto\_id\_travel');

create sequence auto\_id\_crew\_information

owned by crew\_information.id;

alter table crew\_information

alter column id set default nextval('auto\_id\_crew\_information');

create sequence auto\_id\_train

owned by train.id;

alter table train

alter column id set default nextval('auto\_id\_ train);

CREATE ROLE manager

SUPERUSER

login

PASSWORD 'manager';

CREATE ROLE staffs

BYPASSRLS;

CREATE ROLE staff1

login

PASSWORD 'staff';

CREATE ROLE staff2

login

PASSWORD 'staff';

CREATE ROLE staff3

login

PASSWORD 'staff';

GRANT staffs to staff1;

GRANT staffs to staff2;

GRANT staffs to staff3;

CREATE ROLE station\_masters

BYPASSRLS;

CREATE ROLE station\_master1

login

PASSWORD 'station\_master';

CREATE ROLE station\_master2

login

PASSWORD 'station\_master';

CREATE ROLE station\_master3

login

PASSWORD 'station\_master';

CREATE ROLE station\_master4

login

PASSWORD 'station\_master';

CREATE ROLE station\_master5

login

PASSWORD 'station\_master';

GRANT station\_masters to station\_master1;

GRANT station\_masters to station\_master2;

GRANT station\_masters to station\_master3;

GRANT station\_masters to station\_master4;

GRANT station\_masters to station\_master5;

CREATE ROLE programmers

BYPASSRLS;

CREATE ROLE programmer1

login

PASSWORD 'programmer1';

CREATE ROLE programmer2

login

PASSWORD 'programmer2';

CREATE ROLE programmer3

login

PASSWORD 'programmer3';

CREATE ROLE programmer4

login

PASSWORD 'programmer4';

CREATE ROLE programmer5

login

PASSWORD 'programmer5';

GRANT programmers to programmer1;

GRANT programmers to programmer2;

GRANT programmers to programmer3;

GRANT programmers to programmer4;

GRANT programmers to programmer5;

CREATE ROLE customers

BYPASSRLS;

CREATE ROLE customer1

login

PASSWORD 'customer';

CREATE ROLE customer2

login

PASSWORD 'customer';

CREATE ROLE customer3

login

PASSWORD 'customer';

CREATE ROLE customer4

login

PASSWORD 'customer';

CREATE ROLE customer5

login

PASSWORD 'customer';

CREATE ROLE customer6

login

PASSWORD 'customer';

CREATE ROLE customer7

login

PASSWORD 'customer';

CREATE ROLE customer8

login

PASSWORD 'customer';

CREATE ROLE customer9

login

PASSWORD 'customer';

CREATE ROLE customer10

login

PASSWORD 'customer';

CREATE ROLE customer11

login

PASSWORD 'customer';

GRANT customers to customer1;

GRANT customers to customer2;

GRANT customers to customer3;

GRANT customers to customer4;

GRANT customers to customer5;

GRANT customers to customer6;

GRANT customers to customer7;

GRANT customers to customer8;

GRANT customers to customer9;

GRANT customers to customer10;

GRANT customers to customer11;

GRANT ALL

ON ALL Tables

IN SCHEMA public

TO manager;

GRANT insert, update, delete, select

ON train

TO staffs;

GRANT select

ON crew\_information, station, ticket, users, travel, travel\_station

TO staffs;

GRANT insert, delete

ON ticket

TO staffs;

GRANT update

ON travel

TO staffs;

GRANT USAGE ON SEQUENCE auto\_id TO staffs;

GRANT insert, update, delete, select

ON station

TO station\_masters;

GRANT select

ON travel

TO station\_masters;

grant select

on travel\_station

to station\_masters;

grant select

on users

to station\_masters;

GRANT select, insert, update, delete

ON crew\_information, travel, travel\_station

TO programmers;

grant select

on train

to programmers;

grant select

on station

to programmers;

GRANT USAGE ON SEQUENCE auto\_id\_travel TO programmers;

GRANT USAGE ON SEQUENCE auto\_id\_crew\_information TO programmers;

GRANT select

ON crew\_information, travel, travel\_station

TO customers;

set role manager;

grant select

on station

to customers;

grant select, delete, update

on users

to customers;

ALTER TABLE station ENABLE ROW LEVEL SECURITY;

CREATE POLICY account\_station

ON station

TO station\_masters

USING (current\_user = (select u.role

from station s inner join users u on s.station\_master= u.family\_name

where u.role like 'station\_master%'));

ALTER TABLE users ENABLE ROW LEVEL SECURITY;

CREATE POLICY account\_users

ON users

TO customers

USING (role = current\_user);

CREATE POLICY account\_master

ON users

TO station\_masters

USING (role = current\_user);

CREATE POLICY account\_staffs

ON users

TO staffs

USING (role = current\_user);

### **5-1- رویه­ها:**

**Reserve ticket:**

در این بخش staff برای مشتریان بلیط رزرو می­کند.

دو تابع req\_cnt و cap\_cnt تعداد درخواست ها و ظرفیت برای آن سفر را می­یابند که در رویه­ی reserve\_ticket استفاده می­شود.

create function req\_cnt(travel\_id int)

returns int

language plpgsql

as $$

declare req integer;

begin

select request into req from travel where travel\_id = travel.id;

return req;

end;

$$;

create function cap\_cnt(travel\_id int)

returns int

language plpgsql

as $$

declare cap integer;

begin

select capacity into cap from travel where travel\_id = travel.id;

return cap;

end;

$$;

create procedure reserve\_ticket(travel\_id int, name varchar(255),family\_name varchar(255), age int, sex varchar(255))

language plpgsql as $$ begin

if (req\_cnt(travel\_id) < cap\_cnt(travel\_id)) then

insert into ticket(travel\_id , name, family\_name, age, sex) values(travel\_id , name, family\_name, age, sex);

else

raise notice'Capacity is full';

end if;

commit;

end;

$$

**Cancel ticket:**

create procedure cancel\_ticket(ticket\_id int)

language plpgsql as $$ begin

DELETE FROM ticket

WHERE id = ticket\_id;

commit;

end;

$$

**Hire new staff:**

در این رویه مدیر یک staff جدید استخدام می­کند.

create procedure hire\_staff(name varchar(255), family\_name varchar(255))

language plpgsql as $$

DECLARE staff\_cnt integer;

DECLARE role\_name Text;

begin

staff\_cnt = (select COUNT(users.id) from users where users.role like 'staff%');

staff\_cnt = staff\_cnt + 1;

role\_name := CONCAT('staff', (staff\_cnt::Text));

insert into users(name, family\_name, role)values(name, family\_name, role\_name);

execute 'CREATE ROLE "'||role\_name||'" login PASSWORD ''staff''';

execute 'GRANT staffs to "'||role\_name||'"';

commit;

end;

$$

**fired up staff:**

در این رویه مدیر یک staff را اخراج می­کند.

create procedure fireup\_staff(sid integer)

language plpgsql as $$

DECLARE role\_name Text;

begin

role\_name = (select users.role from users where users.id = sid);

delete from users

where users.id = sid and users.role like 'staff%';

execute 'DROP ROLE "'||role\_name||'"';

commit;

end;

$$

**Hire new programmer:**

در این رویه مدیر یک programmer جدید استخدام می­کند.

create procedure hire\_programmer(name varchar(255), family\_name varchar(255))

language plpgsql as $$

DECLARE programmer\_cnt integer;

DECLARE role\_name Text;

begin

programmer\_cnt = (select COUNT(users.id) from users where users.role like 'programmer%');

programmer\_cnt = programmer\_cnt + 1;

role\_name := CONCAT('programmer', (programmer\_cnt::Text));

insert into users(name, family\_name, role)values(name, family\_name, role\_name);

execute 'CREATE ROLE "'||role\_name||'" login PASSWORD '' programmer ''';

execute 'GRANT programmers to "'||role\_name||'"';

commit;

end;

$$

**fired up programmer:**

در این رویه مدیر یک programmer را اخراج می­کند.

create procedure fireup\_programmer(pid integer)

language plpgsql as $$

DECLARE role\_name Text;

begin

role\_name = (select users.role from users where users.id = pid);

delete from users

where users.id = pid and users.role like 'programmer%';

execute 'DROP ROLE "'||role\_name||'"';

commit;

end;

$$

**Add Station and Hire new station Master:**

در این رویه مدیر یک ایستگاه جدید اضافه می­کند و یک مدیر ایستگاه برای آن استخدام می­کند.

create procedure station\_and\_master(name varchar(255), family\_name varchar(255), station\_name varchar(255), sProvince varchar(255), sCity varchar(255))

language plpgsql as $$

DECLARE station\_master\_cnt integer;

DECLARE role\_name Text;

begin

station\_master\_cnt = (select COUNT(users.id) from users where users.role like 'station\_master%');

station\_master\_cnt = station\_master\_cnt + 1;

role\_name := CONCAT('station\_master', (station\_master\_cnt::Text));

insert into users(name, family\_name, role)values(name, family\_name, role\_name);

execute 'CREATE ROLE "'||role\_name||'" login PASSWORD ''station\_master''';

execute 'GRANT station\_masters to "'||role\_name||'"';

insert into station(name, province, city, station\_master) values(station\_name, sProvince, sCity, family\_name);

commit;

end;

$$

**Change station\_master:**

در این بخش مدیر، مسزول یک ایستگاه خاص را عوض می­کند.

create procedure change\_station\_master(mid integer, nname varchar(255), nfamily\_name varchar(255))

language plpgsql as $$

DECLARE mfamily\_name varchar(255);

begin

mfamily\_name = (select family\_name from users where id = mid);

update users

set name = nname

where id = mid;

update users

set family\_name = nfamily\_name

where id = mid;

update station

set station\_master = nfamily\_name

where station\_master = mfamily\_name;

commit;

end;

$$

**Add Travel and it’s crew:**

در این بخش programmer یک سفر جدید به همرا خدمه مرتبط با آن را به جدول سفرها اضافه می­کند.

create procedure add\_travel(train\_id integer, departure\_time time, arrival\_time time, date date, price integer, engine\_driver varchar(255), service\_company varchar(255), wagon\_master varchar(255))

language plpgsql as $$

begin

insert into travel(train\_id, departure\_time, arrival\_time, date, price) values(train\_id, departure\_time, arrival\_time, date, price);

insert into crew\_information(engine\_driver, service\_company, wagon\_master) values(engine\_driver, service\_company, wagon\_master);

commit;

end;

$$

**Add latency to travel:**

در این بخش programmer به یک سغر خاص تاخیر میزند.

create procedure add\_latency (tid integer, latency interval)

language plpgsql as $$

begin

update travel

set departure\_time = departure\_time + latency

where id = tid;

update travel

set arrival\_time = arrival\_time + latency

where id = tid;

commit;

end;

$$

### **5-2- آغازگرها:**

**Increment requests after each ticket reserve:**

این آغازگر به ازای هر رزرو بلیط یک عدد به میزان درخواست­های آن سفر افزوده می­کند.

create function incr\_req()

returns trigger

language plpgsql

as $$

begin

update travel

set request = request + 1 where new.travel\_id = travel.id;

RETURN NEW;

end;

$$;

CREATE TRIGGER request\_inc\_trigger

AFTER insert

ON ticket

FOR EACH ROW

EXECUTE PROCEDURE incr\_req();

**Decrement requests after each ticket cancellation:**

این آغازگر به ازای هر کنسلی بلیط یک عدد از تعداد درخواست­های آن سفر کم می­کند.

create function decr\_req()

returns trigger

language plpgsql

as $$

begin

update travel

set request = request - 1 where old.travel\_id = travel.id;

RETURN NEW;

end;

$$;

CREATE TRIGGER request\_decr\_trigger

AFTER delete

ON ticket

FOR EACH ROW

EXECUTE PROCEDURE decr\_req();

**Set capacity and request after travel insertion:**

این آغازگر به ازای هر سفری که برنامه ریز اضافه می­کند. تعداد درخواست را برابر صفر و تعداد ظرفیت آن سفر را به طور خودکار برابر با ظزفیت قطار مربوط به آن قرار می­دهد.

create function insert\_cap()

returns trigger

language plpgsql

as $$

begin

update travel

set request = 0

where id = new.id;

update travel

set capacity = (select capacity from train where train.id = new.train\_id)

where id = new.id;

RETURN NEW;

end;

$$;

CREATE TRIGGER set\_req\_cap

AFTER insert

ON travel

FOR EACH ROW

EXECUTE PROCEDURE insert\_cap();

**Set relevant travel id after crew\_information insertion:**

این آغازگر به ازای هر گروه خدمع ای که افزوده می­شود آیدی سفر مربوط به آن ها را set می­کند.

create function insert\_travel\_id\_in\_crew()

returns trigger

language plpgsql

as $$

begin

update crew\_information

set travel\_id = (SELECT travel.id,

FROM travel

ORDER BY travel.id DESC

LIMIT 1)

where id = new.id;

RETURN NEW;

end;

$$;

CREATE TRIGGER set\_tid\_crew

AFTER insert

ON crew\_information

FOR EACH ROW

EXECUTE PROCEDURE insert\_travel\_id\_in\_crew ();

### **5-3- توابع:**

1. دریافت اطلاعات قطارهای مطابق با نوع ورودی

CREATE OR REPLACE FUNCTION public.get\_train\_info(IN t character varying)

RETURNS TABLE (id integer, type character varying, capacity integer, color character varying,

Speed character varying(255) , class character varying) AS $func$

BEGIN

RETURN table

SELECT \*

FROM train

where train.type = t;

END

$func$ LANGUAGE plpgsql;

1. دریافت شناسه سفر با مبدا و مقصد ورودی

CREATE OR REPLACE FUNCTION public.get\_travel\_id(IN o character varying, IN d character varying)

RETURNS TABLE (id integer) AS $func$

BEGIN

RETURN QUERY

SELECT distinct ts1.tid

FROM (select ts.travel\_id as tid, ts.station\_id as sid, ts.status, s.name as sname

from travel\_station ts inner join station s on ts.station\_id = s.id) ts1 inner join

(select ts.travel\_id as tid, ts.station\_id as sid, ts.status, s.name as sname

from travel\_station ts inner join station s on ts.station\_id = s.id) ts2

on ts1.tid = ts2.tid

where((ts1.status='begining' and ts2.status='destination' and ts1.sname = o and ts2.sname = d) or

(ts2.status='begining' and ts1.status='destination' and ts2.sname = o and ts1.sname = d));

END

$func$ LANGUAGE plpgsql;

1. دریافت اطلاعات سفر با ورود تاریخ حرکت و مبدا و مقصد سفر

CREATE OR REPLACE FUNCTION public.get\_travel\_info(IN t\_id integer, IN o character varying, IN d character varying)

RETURNS TABLE (tid integer,

train\_id integer,

capacity integer,

request integer,

date date,

price integer,

departure\_time time,

arrival\_time time

) AS

$func$

BEGIN

RETURN QUERY

SELECT t.id, t.train\_id ,t.capacity, t.request,

t.date ,t.price ,t.departure\_time ,t.arrival\_time

FROM travel t inner join (SELECT public.get\_travel\_id(o, d)) x on t.id = x.get\_travel\_id

where t.id = t\_id;

END

$func$ LANGUAGE plpgsql;

1. دریافت اطلاعات خدمه یک سفر

CREATE OR REPLACE FUNCTION public.get\_crew\_info(tid integer)

RETURNS TABLE (id integer, engine\_driver varchar(255), service\_company varchar(255), wagon\_master varchar(255)) AS $func$

BEGIN

RETURN QUERY

SELECT id, engine\_driver, service\_company, wagon\_master

FROM crew\_information

Where crew\_information.travel\_id = tid;

END

$func$ LANGUAGE plpgsql;

### **5-4- پرس و جو:**

1. در جدول سفرها ، سفرهای بین دو ایستگاه خاص را بر اساس قیمت رتبه بندی کند. این رتبه بندی باید در هر نوع قطار باشد.

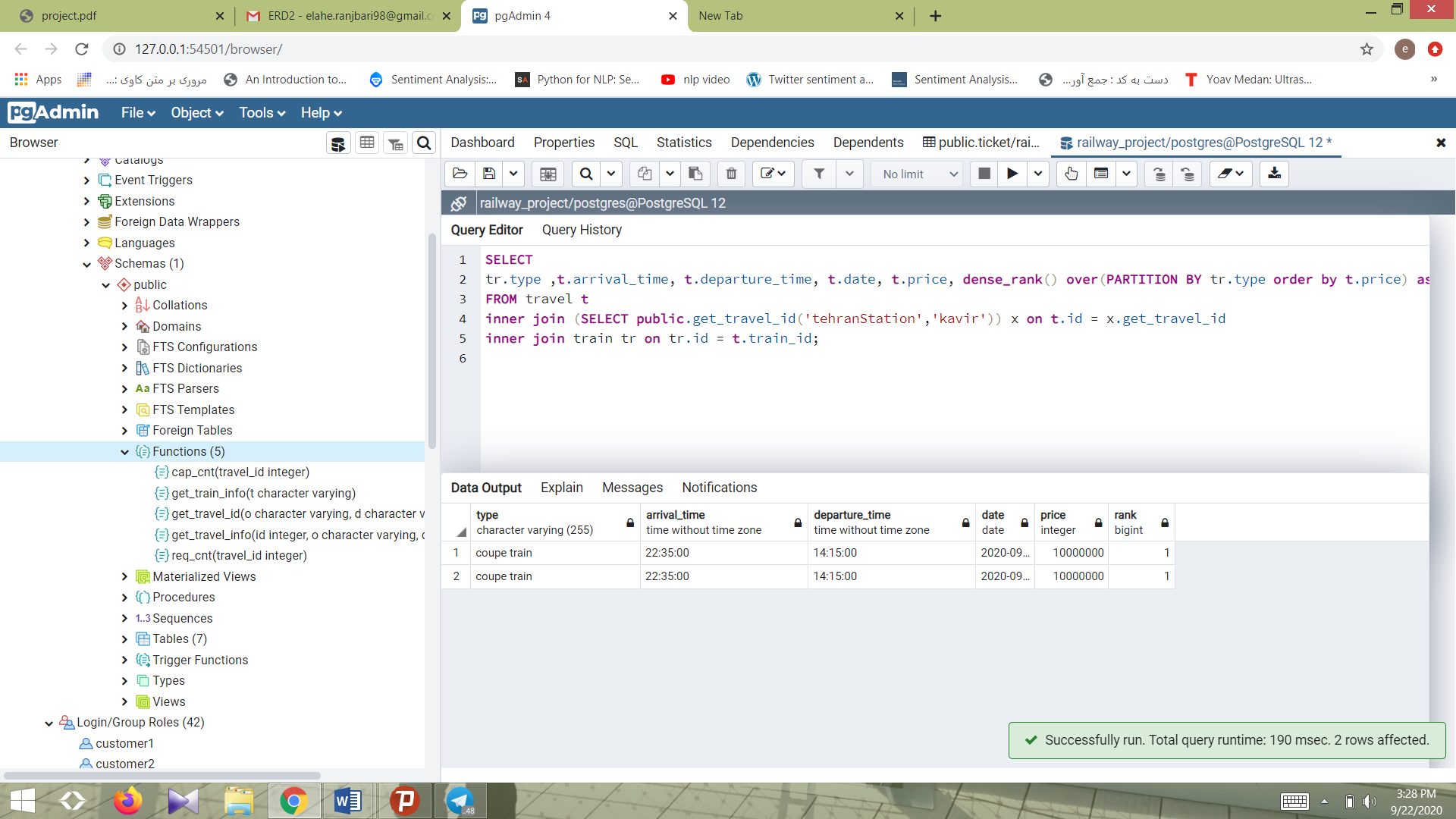
SELECT

tr.type ,t.arrival\_time, t.departure\_time, t.date, t.price, dense\_rank() over(PARTITION BY tr.type order by t.price) as rank

FROM travel t

inner join (SELECT public.get\_travel\_id('tehranStation','kavir')) x on t.id = x.get\_travel\_id

inner join train tr on tr.id = t.train\_id;



1. گزارش خرید بلیت بر اساس روز ، ماه و سال در خصوص سفری که خدمه آن از شرکت خاصی باشد.

SELECT

extract(year from t.date) as year,

extract(month from t.date) as month,

extract(day from t.date) as day,

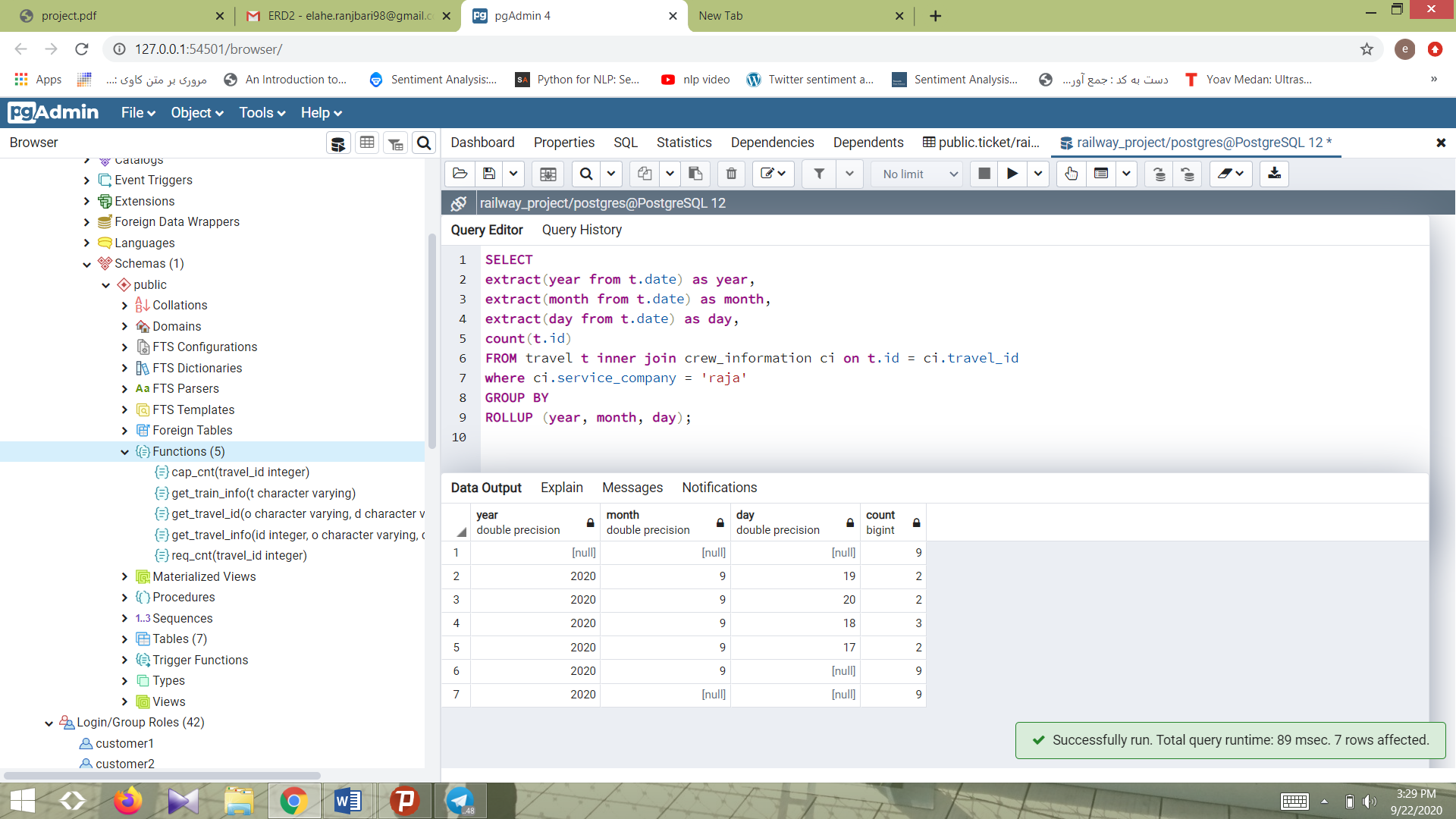
count(t.id)

FROM travel t inner join crew\_information ci on t.id = ci.travel\_id

where ci.service\_company = 'raja'

GROUP BY

ROLLUP (year, month, day);



1. گزارش میانگین بلیت های فروخته نشده براساس نوع، کلاس و سرعت قطار.

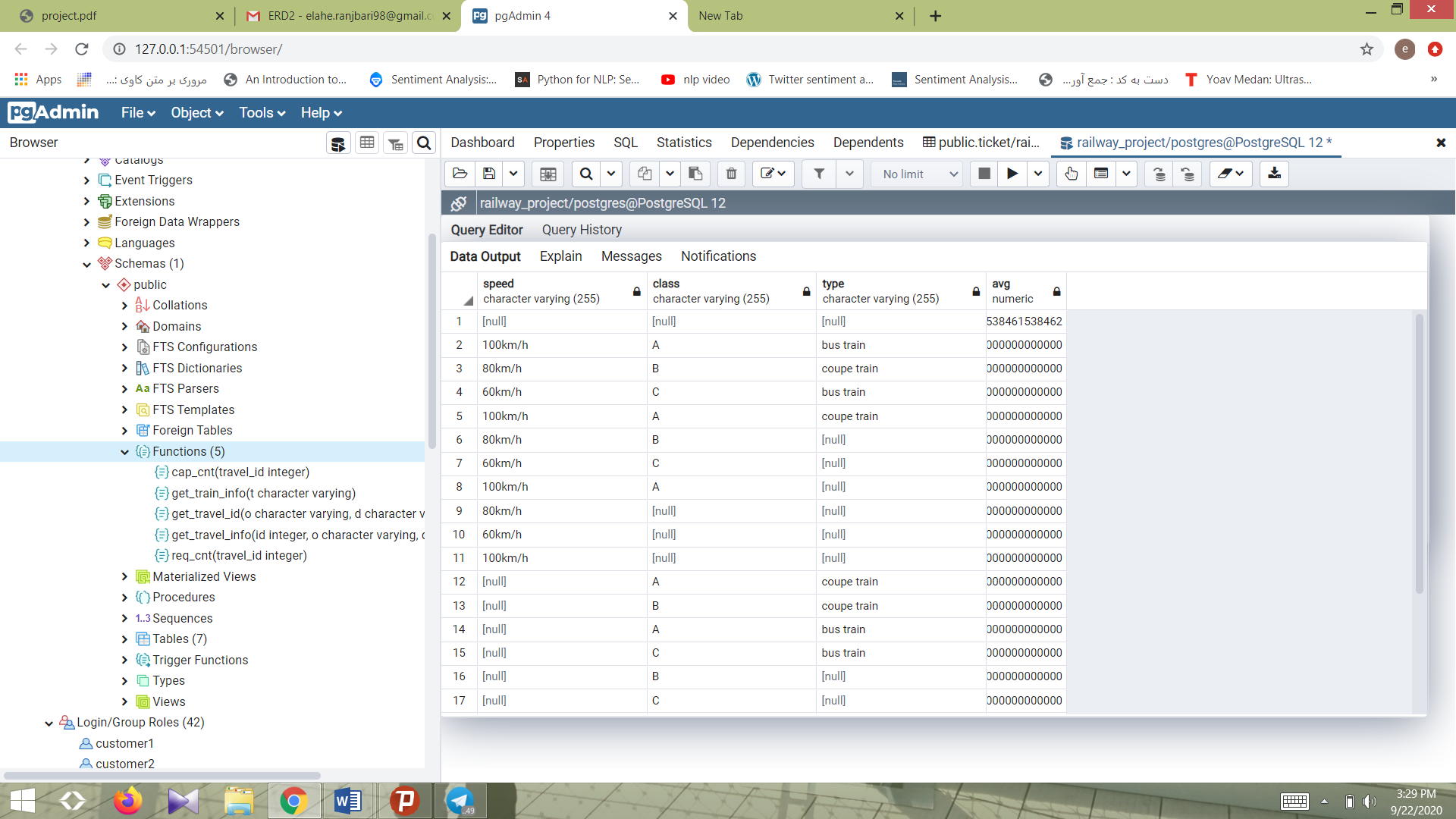
SELECT

tr.speed, tr.class, tr.type, avg(t.capacity)

FROM travel t inner join train tr on t.train\_id = tr.id

GROUP BY

cube(tr.speed, tr.class, tr.type);



1. تعداد قطارهای کوپه ای که هر روز در مسافت های به فاصله زمانی میانگین کمتر از 3 ساعت را با بیش از 10 ظرفیت خالی حرکت می کنند.

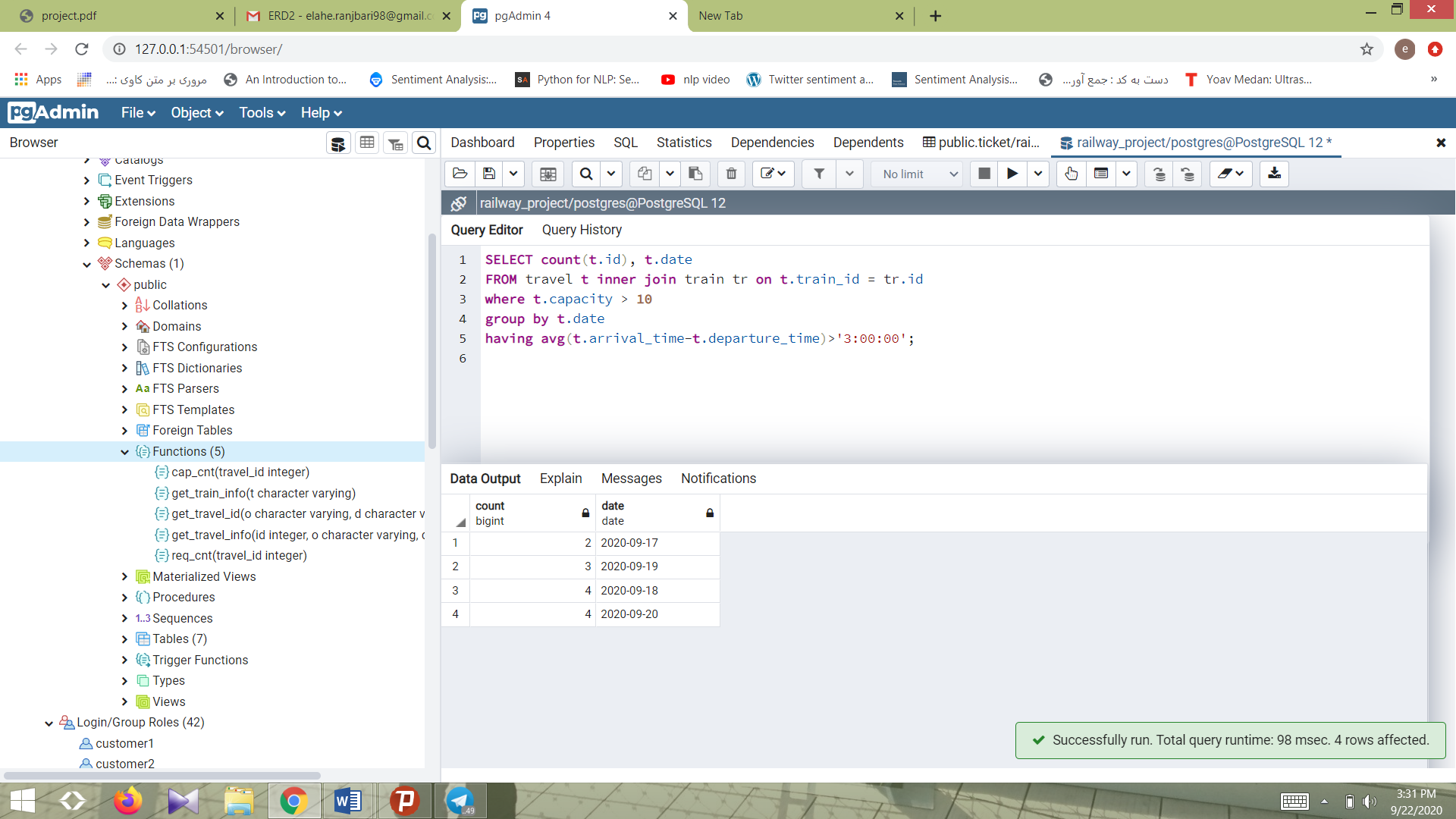
SELECT count(t.id), t.date

FROM travel t inner join train tr on t.train\_id = tr.id

where t.capacity > 10

group by t.date

having avg(t.arrival\_time-t.departure\_time)>'3:00:00';



1. سه مقصدی که کمترین درخواست بلیت را از مقصد خاصی داشته اند.

SELECT t.id, t.request, ds.dst as destination

FROM (select travel\_station.travel\_id as tid

from travel\_station inner join station on travel\_station.station\_id = station.id

where travel\_station.status = 'begining' and station.name = 'tehranStation') bg

inner join (select travel\_station.travel\_id as tid, station.name as dst

from travel\_station inner join station on travel\_station.station\_id = station.id

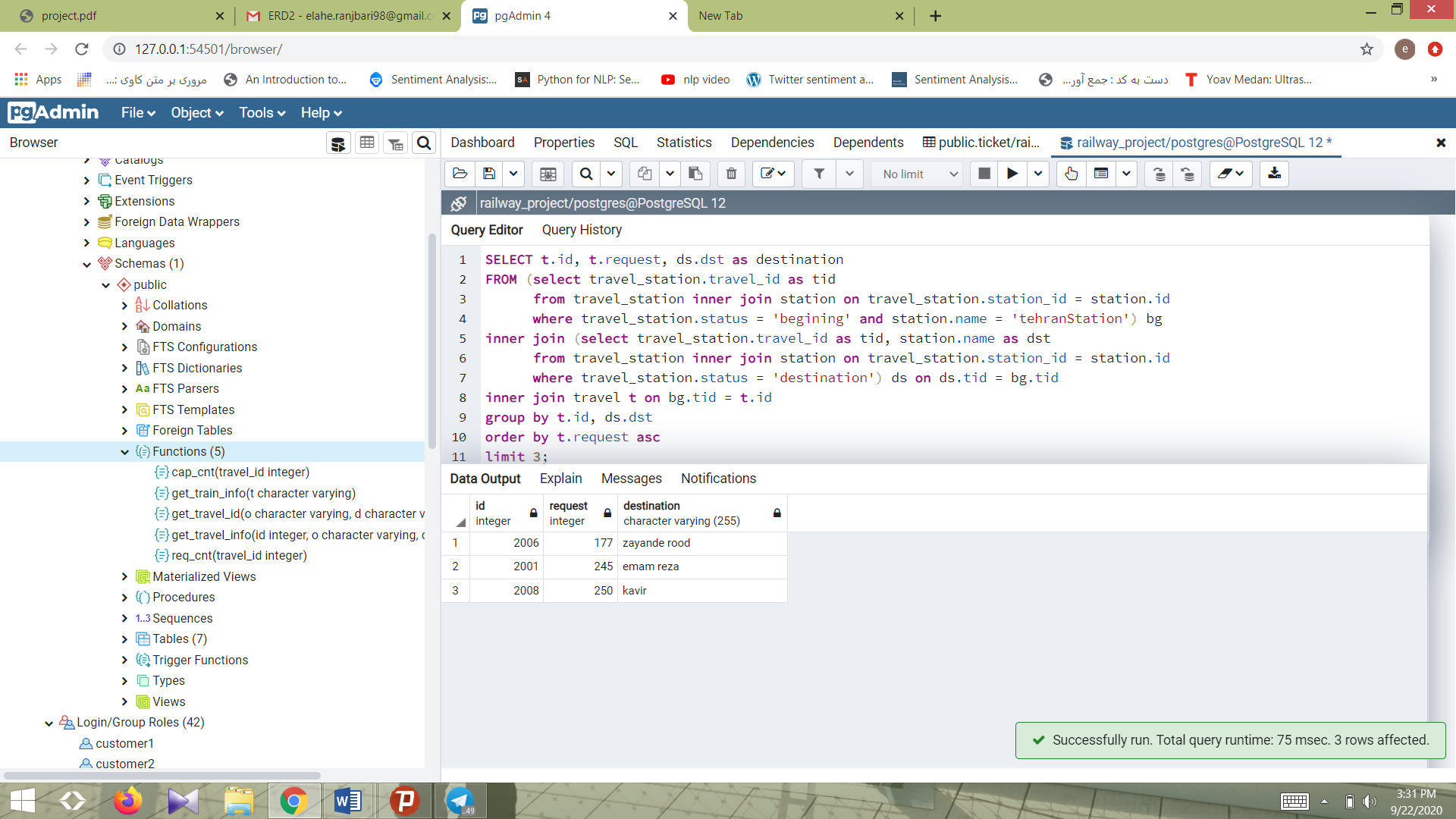
where travel\_station.status = 'destination') ds on ds.tid = bg.tid

inner join travel t on bg.tid = t.id

group by t.id, ds.dst

order by t.request asc

limit 3;



1. گزارش میانگین سنی، ظرفیت خالی و شناسه هر سفر بین دو مقصد مشخص

SELECT t.id, t.capacity, avg(ti.age)

FROM travel t

inner join ticket ti on t.id = ti.travel\_id

inner join (SELECT public.get\_travel\_id('tehranStation','kavir')) x on t.id = x.get\_travel\_id

where t.capacity > 10

group by t.id;

