



# به نام خدا

پروژه پایانی درس پایگاه داده ها

اعضای تیم :

سید علی امامی 9912358005

محمد میرزایی 9912358039

مهدی براتی 9912358008

## محتوا

3	مقدمه :
4	فاز EER :
4	توضیحات زاویه دید نسبت به مسائل و موجودیت ها :
4	موجودیت راننده :
4	موجودیت بکسی باکس :
4	موجودیت بکسی بار :
4	موجودیت بکسی :
4	موجودیت موتورسیکلت :
5	موجودیت خودرو :
5	موجودیت کارمند :
5	موجودیت مشتری :
5	موجودیت سفر ( weak ) :
5	موجودیت سرویس بکسی :
5	موجودیت سرویس بکسی باکس :
5	موجودیت سرویس بکسی بار :
6	موجودیت تراکنش :
7	توضیحات جدول ها :
7	جدول customer :
7	جدول customer addresss :
7	جدول driver :
7	جدول judicial_confirmation :
7	جدول confirmation :
7	جدول baxi box car :
8	جدول baxi bar car :
8	جدول baxi car :
8	جدول trip :
8	جدول baxi bar service :
8	جدول baxi box service :

9	جدول : baxi service
9	جدول : destination baxi service
9	جدول : accept payment
9	جدول weak point ( نقاط ضعف ) :
9	جدول strength point ( نقاط قوت ) :
9	جدول : transactions
10	جدول : Employee
10	جدول : company deposit
11	توضیحات فایل DDL :
13	Q1 :
13	Q2 :
13	Q3 :
13	Q4 :
13	Q5 :
13	Q6 :
14	Q7 :
14	Q8 :
14	Q9 :
14	Q11 :
15	Q12 :
15	Q13 :
15	Q14 :
15	Q15 :
15	Q16 :
16	Q20 :

## مقدمه :

در این پروژه قرار است در مورد پیاده سازی پایگاه داده ی سرویس بکسی توضیحاتی را ارائه دهیم .  
این پروژه در چندین فاز زده شده که هر فاز مرتبط با یک بخش است :

- 1- طراحی EER
- 2- تبدیل EER به جدول
- 3- تبدیل جداول به SQL در برنامه My SQL
- 4- افزودن داده ها به پایگاه داده
- 5- پیاده سازی Trigger و Assertion
- 6- استفاده از test case برای تست مسئله
- 7- پیاده سازی Query های خواسته شده در فایل پروژه
- 8- بازیابی اطلاعات مورد نیاز بر اساس Query های موجود در فایل پروژه

## فاز EER :

### توضیحات زاویه دید نسبت به مسائل و موجودیت ها :

**موجودیت راننده :** در این موجودیت صفات های مرتبط با راننده در نظر گرفته شده برای هر راننده یک کد معرف تعریف شده تا چنانچه راننده جدیدی را ثبت میکند ، راننده جدید از کد این کد معرف استفاده کند . رابطه معرفی کردن به خود راننده باز میگردد که علاوه بر آن ویژگی درصد سود نیز وجود دارند تا به وسیله آن بتوانیم میزان سود راننده معرفی کننده را بدست آوریم . ویژگی زوم سایز برای حالتی است که راننده بتواند محدوده دریافت درخواست ها را گسترش دهد که بتواند درخواست های دورتر از 5 کیلومتر را نیز مشاهده کند. کد ملی صفتی است که به عنوان کلید در نظر گرفته شده است و یکتاست . موجودیت راننده به صورت مستقیم در سیستم وجود ندارد ، برای همین در یک رابطه ارث بری از نوع اجباری با موجودیت بکسی باکس یا بکسی یا بکسی بار شرکت میکند ( این حالت total disjoint در نظر گرفته شده است

**موجودیت بکسی باکس :** تمام صفات موجودیت راننده را دارد ( مورد استفاده در حمل و نقل موتوری )

**موجودیت بکسی بار :** تمام صفات موجودیت راننده را دارد ( مورد استفاده در حمل و نقل باری )

**موجودیت بکسی :** تمام صفات موجودیت راننده را دارد ( مورد استفاده برای سفر که شامل سرویس بکسی و بکسی بانوان می باشد )

**موجودیت موتورسیکلت :** این موجودیت شامل اطلاعات موتورسیکلت میباشد و کلید این جدول پلاک است و به دلیل اینکه در سیستم وسیله نقلیه بدون مالک وجود ندارد ، این موجودیت در رابطه اجباری با بکسی باکس قرار گرفته است ( موتورسیکلت به دلیل اینکه ویژگی پلاک که منحصر به فرد است را دارد امکان تبدیل به موجودیت ضعیف شدن را ندارد )

**موجودیت خودرو :** این موجودیت شامل اطلاعات خودرو میباشد به دلیل اینکه سرویس بکسی بار و بکسی نیازمند خودرو میباشد ، هردو با این موجودیت رابطه دارند و همچنین به دلیل اینکه موجودیت خودرو به صورت مستقل در سیستم وجود ندارد رابطه ی اجباری با بکسی و بکسی بار دارد . (خودرو به دلیل اینکه ویژگی پلاک که منحصر به فرد است را دارد امکان تبدیل به موجودیت ضعیف شدن را ندارد )

**موجودیت کارمند :** در این موجودیت ویژگی های مرتبط با کارمند ذکر شده است و کدپرسنلی خاصیت کلیدی دارد . ویژگی تاریخ استخدام جهت بازایی برخی نمونه از این موجودیت با صفات خاص استفاده میشود مانند کسانی که بیش از 1 سال کار میکنند و همچنین برای استفاده در log سیستم نیز بکار میرود .

**موجودیت مشتری :** در این موجودیت اطلاعات مشتری ذخیره می شود و به دلیل اینکه مشتری میتواند چندین آدرس منتخب داشته باشد ، ویژگی به صورت چندگانه در نظر گرفته شده است . سن مشتری و جنسیت آن جهت اعمال برخی از محدودیت ها از جمله محدودیت سنی بالای 15 سال و همچنین محدودیت قبول درخواست از سمت مشتری برای بکسی بانوان که جنیست باید مشخص شده باشد ( خانم باشد ) . زمان ثبت نام برای log استفاده میشود .

**موجودیت سفر ( weak ) :** موجودیت سفر یک موجودیت ضعیف است که مالک آن موجودیت مشتری است . موجودیت سفر ، مفهوم درخواست سفر را دارد . در این موجودیت هزینه کل ، هزینه ی تمام شده ی سفر است . کلید ( partial key ) این موجودیت آیدی می باشد . این موجودیت به 3 موجودیت سرویس بکسی ، سرویس بکسی باکس و سرویس بکسی بار با رابطه total disjoint متصل است و به صورت مستقل وجود ندارد .

**موجودیت سرویس بکسی :** این موجودیت تمام صفات موجودیت سفر را داراست و رفت و برگشت بودن سفر و مقصد های مختلف را پوشش می دهد .

**موجودیت سرویس بکسی باکس :** این موجودیت تمام صفات موجودیت سفر را داراست و شامل ویژگی های کالا می باشد . همچنین جابه جایی در این موجودیت ، هزینه حمل و نقل بدون درنظر گرفتن بیمه سفر است که هزینه بیمه به صورت یک صفت جداگانه است . در این موجودیت صفت کمک کاربر را داریم برای حالتی که مشتری قصد کمک در جابه جایی را دارد .

**موجودیت سرویس بکسی بار :** این موجودیت تمام صفات موجودیت سفر را داراست و شامل ویژگی های کالا می باشد.

**موجودیت تراکنش:** موجودیت های کارمند برای برداشت حقوق و برای واریز پاداش , سهم سوخت به راننده نیاز به تراکنش دارد .

موجودیت مشتری برای واریز پول به کیف پول خود نیاز به تراکنش دارد . زمانی که یک سفر توسط راننده تایید میشود برای پرداخت هزینه نیاز به تراکنش است .

راننده جهت برداشت پول موجود در کیف پول خود , نیاز به تراکنش دارد .

برای پرداخت هزینه ی سفر قبول شده , نیاز به تراکنش داریم .

با توجه به استفاده های بالا و صفات تکرار شونده تبدیل به یک موجودیت قوی شد و در آن صفات تراکنش درج شده است و کلید آن آیدی می باشد .

## توضیحات جدول ها :

**جدول customer :** در این جدول ویژگی های مشتری نوشته شده اند . phone number به عنوان کلید انتخاب شده است . از ویژگی زمان ثبت نام برای log system استفاده میکنیم .

**جدول customer addresss :** این جدول یک صفت چند مقداره و ترکیبی از جدول customer میباشد . کلید جدول ترکیبی از cphone number , location , label میباشد .

**جدول driver :** در این جدول اطلاعات راننده ذخیره شده و کلید آن national id می باشد . از سمت رابطه ی معرفی راننده , کلید خارجی identifier national id و interest rate به این جدول اضافه میشود .

**جدول judicial\_confirmation :** این جدول برای رابطه ی تایید قضایی می باشد که شامل صفات رابطه و همچنین کلید از دو سمت کارمند با عنوان employee personnel id و کلید از سمت راننده با عنوان driver national id به این جدول اضافه شده است . کلید جدول ترکیبی از این دو کلید خارجی می باشد . از ویژگی jtime برای log system استفاده میکنیم .

**جدول confirmation :** این جدول شامل صفاتی که نشان دهنده ی تایید نهایی , تاریخ تایید گواهینامه و تاریخ بررسی خودرو می باشد . کلید جدول ترکیبی از 2 کلید خارجی employee personnel id و driver national id می باشد . از ویژگی ctime برای log system استفاده میکنیم .

**جدول baxi box car :** این جدول به دلیل رابطه ی total با موجودیت موتورسیکلت آن را تبدیل به یک جدول کردیم .  
driver national id به عنوان کلید خارجی به این جدول اضافه شده همچنین ویژگی های موتورسیکلت نیز به این جدول اضافه شده است . کلید جدول پلاک موتورسیکلت می باشد .



**جدول baxi bar car :** این جدول به دلیل رابطه ی total با موجودیت خودرو آن را تبدیل به یک جدول کردیم . اطلاعات خودرو در این جدول ذکر شده و driver national id در این جدول به عنوان کلید خارجی ذکر شده . کلید این جدول پلاک می باشد ( نوع خودرو وانت باشد , ظرفیت در این قسمت به معنای کیلوگرم می باشد )

**جدول baxi car :** این جدول به دلیل رابطه ی total با موجودیت خودرو آن را تبدیل به یک جدول کردیم . اطلاعات خودرو در این جدول ذکر شده و driver national id در این جدول به عنوان کلید خارجی ذکر شده . کلید این جدول پلاک می باشد ( نوع خودرو سواری باشد , ظرفیت در این قسمت به معنای تعداد نفرات می باشد )

**جدول trip :** این جدول شامل اطلاعات سفر می باشد و به دلیل اینکه این موجودیت به صورت مستقل در سیستم وجود ندارد , مالک آن را مشتری قرار میدهیم که با رابطه ی مالکیت درخواست کردن به مشتری متصل شده است . کلید این جدول ترکیبی از id و cphone number است ( شماره ی مشتری ) از ویژگی req time برای log system استفاده میکنیم .

**توضیحات :** موجودیت سفر به صورت مستقل وجود ندارد , لذا نوع سفر باید یکی از انواع سرویس بکسی بار , سرویس بکسی باکس و سرویس بکسی باشد ( سرویس بکسی شامل بکسی بانوان نیز می باشد )

**جدول baxi bar service :** در این جدول چنانچه سفر از نوع حمل بار باشد , اطلاعات وارد می شود . در این جدول کلید خارجی از سمت مسافر که ترکیبی است , به این جدول می آید . کلید جدول ترکیبی از trip id و cphone number می باشد .

**جدول baxi box service :** این جدول شامل اطلاعات کالایی که بکسی باکس , حمل آن را بر عهده دارد . کلید این جدول ترکیبی از دو کلید خارجی از سمت سفر با نام های trip id و cphone number می باشد .

**جدول baxi service** : این جدول رفت و برگشتی بودن یک سفر را تعیین می کند . کلید آن ترکیبی از 2 کلید خارجی trip id و cphone number می باشد .

**جدول destination baxi service** : این جدول حاصل از صفت چند مقداره سرویس بکسی می باشد که چنانچه چند مقصد برای سفر داشته باشیم , میتوان چند رکورد در این جدول ثبت نمود . کلید جدول ترکیبی از موقعیت trip id , destination location , destination city و cphone number می باشد .

**جدول accept payment** : زمانی که یک سفر ثبت نهایی می شود و راننده ای آن سفر را قبول می کند , یک رکورد در این جدول ایجاد می شود که شامل صفات مربوط به قبول سفر می باشد و به دلیل اینکه برای سفر قبول شده نیاز به انجام تراکنش هست , کلید خارجی از سمت تراکنش در این جدول ذکر شده ( transaction id ) .

در این جدول کلید خارجی از سمت سفر به صورت ترکیبی trip id و cphone number می باشد . به دلیل رابطه موجود با راننده کلید راننده نیز با عنوان driver national id در این جدول ذکر شده است . با توجه به اینکه transaction id پس از ایجاد یک رکورد از قبول سفر ایجاد میشود , قابلیت کلید شدن را ندارد زیرا ارجاع به جدول transaction وجود دارد . پس به همین خاطر کلید این جدول ترکیبی از دو کلید trip id و number cphone است. از ویژگی start time برای log system استفاده میکنیم .

**جدول weak point ( نقاط ضعف )** : جدول قبول و پرداخت دارای صفت چندمقداره ی نقاط ضعف بود و این صفت مربوط به آن جدول می باشد . کلید این جدول ترکیبی از کلید خارجی driver national id و driver weak point می باشد .

**جدول strength point ( نقاط قوت )** : جدول قبول و پرداخت دارای صفت چندمقداره ی نقاط قوت بود و این صفت مربوط به آن جدول می باشد . کلید این جدول ترکیبی از کلید خارجی driver national id و driver strength point می باشد .

**جدول transactions** : این جدول شامل ویژگی های تراکنش می باشد . علاوه برآن شامل کلید خارجی cphone number از سمت مشتری جهت واریز پول به کیف پول خود قرار گرفته است . کلید خارجی employee personnel id جهت برداشت حقوق کارمند قرار گرفته است. کلید خارجی driver national id برای برداشت پول از کیف پول راننده می باشد . کلید جدول id است .

از ویژگی transaction time برای log system استفاده میکنیم .  
**جدول Employee** : این جدول شامل اطلاعات کارمند می باشد . از ویژگی employeement date برای log system استفاده میکنیم . کلید جدول personnel id می باشد .

**جدول company deposit** : در این جدول زمانی که پرداختی از سمت شرکت توسط کارمند برای راننده بابت پاداش یا سهم سوخت صورت می گیرد , رکورد ایجاد می شود .  
Driver national id کلید خارجی از سمت راننده و employee personnel id کلید خارجی از سمت کارمند می باشد .  
Transaction id کلید خارجی از سمت تراکنش می باشد .  
کلید جدول transacton id می باشد .  
به دلیل وجود کلید از سمت تراکنش در این جدول برای log میتوانیم این جدول را با رکورد موجود در تراکنش جوین کنیم و از زمان تراکنش برای log system استفاده کنیم .

## توضیحات فایل DDL :

در این فایل دستورات مورد نیاز جهت ایجاد schema و تشکیل جدول ها قرار داده شده اند و نام سر ستون ها به همراه نوع داده ای که میتوانند داشته باشند قرار گرفته شده است . از جمله استفاده از نوع داده ای int , نوع داده ای date time و نوع داده ای enum برای ستون هایی که مقادیر آن ها به صورت چندگزینه ای بود مانند ← certificate time ( نوع گواهینامه ) که میتواند به 5 دسته ی :

- Level 1
- Level 2
- Level 3
- Motorcycle
- Special

تقسیم گردد که متناظرن برابر است با

- پایه 1
- پایه 2
- پایه 3
- موتورسیکلت
- خاص

در ادامه برخی از مقادیر , مقدار default قرار گرفته اند مانند تراکنش ها یا زمان ثبت درخواست سفر و یا زوم ساینز قرار گرفته است . با توجه به نوع موجودیت و اهمیت آن ویژگی , برخی محدودیت ها روی آن قرار داده ایم مانند not null یا check .

برخی متغیرها نیازمند بارگزاری به صورت فایلی می باشد که از نوع blob درنظر گرفته شده ولی برای افزودن داده به DB برای چک کردن query ها , قابلیت افزودن تصویر ها را قرار ندادیم . به همین دلیل محدودیت not null در مقابل این صفات به صورت comment شده قرار گرفته است .

متغیر location از نوع داده ای point است که برای نگه داری موقعیت از آن استفاده شده است ولی برای تست DB , آن را به صورت یک عدد float به طور موقت درنظر گرفته ایم .

در واقع رویکرد به این صورت است که برای بدست آوردن فاصله ی میان موقعیت مبدا سفر مشتری تا موقعیت خودرو رانندگان از یکدیگر کم میشوند و چنانچه کمتر از zoom size باشد , یعنی سفر در محدوده دید راننده قرار دارد و به راننده این سفرها نمایش داده میشود .

در این سیستم فرض شده , حساب های کاربری مشتریان , کارمندان , سفرها و رانندگان حذف نمیشود تا رکوردهای موجود که از کلید خارجی بهره میبرند حذف نشوند . هرچند cascade و restrict بودن آنها نیز رعایت شده است .

**توضیح :** با توجه به اینکه در My SQL دستور Assertion وجود ندارد به همین خاطر محدودیت هایی که میبایست توسط Assertion اعمال شد را به شکل Check قرار دادیم . قابل قبول بودن مقدار وزن بار یا باکس ( در محدوده مجاز باشد ) به شکل check در دستور create table بکسی بار و بکسی باکس قید شده است . برای اینکه موجودی کیف پول راننده کمتر از 50.000- نباشد , یک check برای آن در حین ساخت جدول راننده قرار داده شده است .

منفی نبودن موجودی کیف پول کاربر نیز به شکل check در دستور create table قرار داده شده است . علاوه برآن check برای اینکه سن کاربر بیشتر از 15 سال باشد , در این دستور قرار داده شده است . حداقل سن 18 سال نیز برای راننده به صورت check پیاده سازی شده است .

## توضیحات Query ها :

**Q۱ :** برای دسترسی به اطلاعات راننده و خودرو ، 2 جدول driver و baxi car را با یکدیگر join کرده و سپس bbcname را برابر با pride و رنگ را برابر blue قرار میدهیم و اطلاعات راننده مورد نظر بدست می آید .

**Q۲ :** برای بدست آوردن اطلاعات راننده مورد نظر باید 2 جدول driver و baxi car را با یکدیگر join کنیم و sex را برابر female قرار دهیم و age باید بزرگتر از 30 سال باشد و رنگ خودرو blue و نوع آن sedan در نظر میگیریم .

**Q۳ :** برای اطلاعات خواسته شده نیاز است جداول driver و judicial\_confirmation را با هم join کنیم و باید نوع خودرو truck باشد که car\_type را برابر با ان قرار میدهیم (در صورتی که join انجام شود ،تنها اطلاعات افرادی می آید که تایید قضایی و تایید برای انها صورت گرفته است)

**Q۴ :** در این کوئری ، جدول customer و customer\_address را با یکدیگر جویین میکنیم و سپس ادرس و اطلاعات مشتریان را خواهیم داشت .

شرط wallet بزرگتر از 50000 را برای موجودی کیف پول در نظر گرفته و چنانچه شماره تلفن همراه در کوئری داخلی باشد رکورد مورد نظر در جواب حاصل میشود . در کوئری داخلی چک میشود ک مشتری از دسته مشتریانی باشد ک حداقل ۲ ادرس منتخب داشته اند.

**Q۵ :** جداول trip و customer و baxi\_service را با یکدیگر join می کنیم. با توجه به اینکه سفر رفت و برگشتی است ، round\_trip را با yes برابر قرار می دهیم و محدوده سن را نیز بین ۱۵ تا ۱۸ قرار می دهیم. اطلاعات را بر اساس req\_time دسته بندی کرده و سپس بر اساس تعداد بصورت مرتب صعودی مرتب سازی می کنیم. دستور limit 1 اولین رکورد بدست آمده را در نظر می گیرد.

**Q۶ :** جداول driver و accept\_payment را با یکدیگر join کرده و سپس زمان شروع سفرها را 2024-01-01 00:00:00 و زمان پایان سفر را نیز تاریخ 2024-01-01 00:00:00 در نظر می گیریم. سپس رکورد ها را بر اساس نام و نام خانوادگی و کدملی و موجودی کیف پول دسته بندی می کنیم. شرط تعداد سفر بیش از دو عدد را با استفاده از count در having پیاده سازی می کنیم.

**Q7 :** ابتدا جداول `customer` و `trip` و `accept_payment` و `transaction` را با یکدیگر `join` می کنیم. با توجه به اینکه زمان سفر باید بیشتر از ۲ ساعت باشد بنابراین زمان پایان را از زمان شروع کم کرده تا به این خواسته برسیم. با توجه به اینکه درخواست ها باید در یک هفته باشد، بنابراین با استفاده از `date_sub` و مقدار زمان فعلی `[ now ]` ، زمان یک هفته را بدست آورده و سپس بر اساس نام و نام خانوادگی و شماره تلفن آنها را دسته بندی می کنیم. با توجه به اینکه یخشی از خواسته سوال، جمع مبالغ هزینه سفر ها است از `sum` برای جمع مبالغ سفر استفاده می کنیم.

**Q8 :** ابتدا جداول مورد نیاز را با یکدیگر `join` می کنیم سپس شرط نوع خودرو را وانت قرار می دهیم. سپس نوع ناتوانی را نیز مشخص کرده و بعد از آن گازسوز بودن خودرو را چک می کنیم. سپس با استفاده از `all` راننده هایی که سن آنها از میانگین سنی راننده ها بیشتر است را جدا می کنیم. سپس بر اساس کدملی و نام و نام خانوادگی و سن رکورد ها را دسته بندی می کنیم. در `having` شرط تعداد سفر های راننده را چک می کنیم که بیشتر از تعداد سفر های رانندگان با حداقل ظرفیت ۴ مسافر باشد. ( این کوئری به صورت تو در تو است که بر اساس کد ملی رکورد ها را دسته بندی می کنیم).

**Q9 :** ابتدا جداول مورد نیاز را با یکدیگر `join` می کنیم. (برای بدست آوردن پلاک به جدول `baxibar_car` نیازمندیم). سپس صفت نوع کالا را از جدول `baxibar_service` برابر با شکستی قرار می دهیم. ارزش کالا را نیز حداکثر برابر با ۵۰ میلیون قرار می دهیم. برپای بدست آوردن اطلاعات راننده نیازمند جدول `driver` هستیم.

**Q11 :** ابتدا جداول مورد نیاز را با یکدیگر `join` می کنیم. با استفاده از `in` یک کوئری دیگر برای بدست آوردن مجموعه شماره همراه هایی که طول سفر آنها حداقل یک ساعت بوده باشد می نویسیم. ( بصورتی که تفاوت زمان پایان با زمان شروع سفر حداقل ۱ ساعت باشد) و بر اساس شماره همراه آنها را دسته بندی می کنیم. با شرط موجود در `having` تعداد حداکثر را برابر با 2 قرار می دهیم. سپس در کوئری بیرونی ، رکورد ها را بر اساس نام و نام خانوادگی و شماره همراه دسته بندی می کنیم. در `having` شرط مجموع `cost` ها را بیشتر از ۱۰۰ هزار قرار می دهیم. ( در صورتی که مجموع `cost` ها بیشتر از ۱۰۰ هزار باشد به این معناست که کیف پول بیشتر از ۱۰۰ هزار شارژ شده است).

**Q۱۲:** برای بدست آوردن خواسته مسئله، نیاز است که تعداد کل گزارشات پیگیری شده را با استفاده از `count` بدست آورده و بر تعداد کل گزارشات تقسیم کنیم. در این حالت می توانیم از `case when` استفاده کنیم که شرط پیگیری شدن را در جلوی `count` قرار دهیم. تعداد کل نیز با استفاده از `count` بدست می آوریم.

**Q۱۳:** در این کوئری شرط مدرک را با توجه به `enum` نسبت داده شده قرار می دهیم. شرط بعدی برای حقوق کارمندان است که آن را بیشتر از ۵ میلیون قرار دادیم. سپس تاریخ استخدام را با تاریخ بدست آمده که حاصل تفریق زمان حال از زمان یک سال قبل است مقایسه می کنیم و در پایان با استفاده از `not in` چک می کنیم که مدیر دپارتمان در جواب های کوئری ظاهر نشود.

**Q۱۴:** ابتدا جداول مورد نیاز را با یکدیگر `join` می کنیم. شرط نوع پرداخت را با `cash` برابر قرار می دهیم. سپس شرط سال تولید خودرو را قبل از ۲۰۰۹ ست می کنیم که خودرو های فرسوده را شامل شود. سپس بر اساس نام و نام خانوادگی و نام خودرو رکورد ها را مرتب می کنیم. سپس بر اساس تعداد سفرها آنها را مرتب می کنیم و درپایان ۳ رکورد اول را به عنوان پاسخ در نظر می گیریم. در قسمت `select` برای بدست آوردن مجموع مبالغ از `sum` استفاده می کنیم.

**Q۱۵:** ابتدا جداول مورد نیاز را `join` میکنیم سپس موقعیت را `sharrey` قرار میدهیم. در `select` با استفاده از تابع `count` و تقسیم و ضرب، جواب مورد نیاز را ایجاد میکنیم. برای اینکه بتوانیم به سوال پاسخ دهیم باید 2 نمونه از موجودیت `trip` ایجاد کنیم. 1 موجودیت برای `trip` ها موفق و 1 موجودیت برای تمام `trip` ها. موجودیت `trip` که با موجودیت `accept payment` جوین شده است، سفرهای موفق است. روش محاسبه به این صورت است که تعداد سفرهای موفق را از تعداد کل سفرها کم کرده و سپس مقدار بدست آمده را در عدد 100 ضرب میکنیم و در نهایت برکل تعداد سفرها تقسیم میکنیم که نتیجه موردنظر بدست آید.

**Q۱۶:** ابتدا جداول مورد نیاز را با یکدیگر جوین میکنیم. سپس با استفاده از `timediff` چک میکنیم که تفاوت بین زمان شروع و پایان از زمان تخمینی کمتر باشد. سپس با استفاده از `timestampdiff` تفاوت بین زمان سفر و زمان تخمینی را برحسب ثانیه بدست آورده و براساس آن رکوردهای جدول را مرتب میکنیم. در پایان 10 `limit` را برای جداکردن 10 رکورد جدا میکنیم.



**Q20:** ابتدا جداول مورد نیاز را با یکدیگر join کرده سپس چک میکنیم که نوع خودرو سواری باشد. پس از آن با استفاده از `date_sub` چک میکنیم که از تاریخ تایید گواهینامه و تایید نامه قضایی بیش از 2 ماه گذشته باشد. در مرحله بعد، مجموعه ی کدملی هایی که مسافر آنها مرد بوده و حداکثر 3 سفر داشته اند را با یک query تودرتو پیاده سازی میکنیم. سپس مجموعه ی کدملی هایی که در سفرهای بالاتر از 10 کیلومتر در تاریخ موردنظر امتیاز بالاتر از 3 گرفته اند را با یک query تودرتو دیگر پیاده سازی میکنیم.

تصویر از اجراهای کوئری ها :

```
2
3
4 -- Q1
5 select Driver.*
6 from Driver, Baxi_Car
7 where national_id = driver_national_id and bcname = 'pride' and color = 'blue';
8
9
10 -- Q2
11 select Driver.*
12 from Driver join Baxi_Car on national_id = driver_national_id
13 where sex = 'female' and age > 30 and color = 'blue' and car_type = 'sedan';
14
15
16 -- Q3
17 select fname, lname, judicial_confirmation_letter_date, judicial_confirmation_number, certificate_confirmation_date, certificate_number
18 from Driver, BaxiBar_Car, Confirmation, Judicial_Confirmation
```

fname	lname	ident_code	sex	age	national_id	sheba_number	distance	wallet	score	phone_number	national_card_scan	certificate_scan	identifier_national_id	prof_image	disability_type	certificate_type	certificate_date	act_permission	local
karim	sarabi	456789	male	35	65987412	IR2327232143	9.8	20000	3.5	09903002121	none	none	none	none	deaf	level 1	1398-02-03	1	7

```
8
9
10 -- Q2
11 select Driver.*
12 from Driver join Baxi_Car on national_id = driver_national_id
13 where sex = 'female' and age > 30 and color = 'blue' and car_type = 'sedan';
14
15
16 -- Q3
17 select fname, lname, judicial_confirmation_letter_date, judicial_confirmation_number, certificate_confirmation_date, certificate_number
18 from Driver, BaxiBar_Car, Confirmation, Judicial_Confirmation
```

fname	lname	ident_code	sex	age	national_id	sheba_number	distance	wallet	score	phone_number	national_card_scan	certificate_scan	identifier_national_id	prof_image	disability_type	certificate_type	certificate_date	act_permission	local
samine	ahmadi	123457	female	37	049066407	IR2324597126	7.1	2000000	3.4	09181001111	none	none	055055403	none	no	level 3	1399-06-25	1	1

```
14
15
16 -- Q3
17 select fname, lname, judicial_confirmation_letter_date, judicial_confirmation_number, certificate_confirmation_date, certificate_number
18 from Driver, BaxiBar_Car, Confirmation, Judicial_Confirmation
19 where Driver.national_id = BaxiBar_Car.driver_national_id and Driver.national_id = Confirmation.driver_national_id
20 and Driver.national_id = Judicial_Confirmation.driver_national_id and car_type = 'truck';
21
22
23 -- Q4
24 select distinct Customer.*
25 from Customer, Customer_Address
26 where wallet >= 50000 and phone_number in (select cphone_number
27 from Customer, Customer_Address
28 where Customer.phone_number = Customer_Address.cphone_number
29 group by cphone_number
30 having count(*) >= 2);
```

fname	lname	judicial_confirmation_letter_date	judicial_confirmation_number	certificate_confirmation_date	certificate_number
amin	ahmadi	2024-03-08	JC123456	2022-03-08	CN123456

```

21
22 -- Q4
23 • select distinct Customer.*
24   from Customer, Customer_Address
25  where wallet >= 50000 and phone_number in (select cphone_number
26                                             from Customer, Customer_Address
27                                             where Customer.phone_number = Customer_Address.cphone_number
28                                             group by cphone_number
29                                             having count(*) >= 2);
30
31
32 -- Q5
33 • select date(req_time)
34   from Trip, Customer, Baxi_Service
35  where Customer.phone_number = Trip.cphone_number and Trip.id = Baxi_Service.trip_id

```

Result Grid							
Filter Rows:							
Export:   Wrap Cell Content:							
fname	lname	sex	age	phone_number	reg_time	email	wallet
▶ kazem	kazemi	male	16	09184561221	2024-03-11 07:48:05	kazem@cust.com	70000

```

31
32 -- Q5
33 • select date(req_time)
34   from Trip, Customer, Baxi_Service
35  where Customer.phone_number = Trip.cphone_number and Trip.id = Baxi_Service.trip_id
36         and Trip.cphone_number = Baxi_Service.cphone_number and round_trip = 'yes'
37         and age <= 18 and age >= 15
38  group by date(req_time)
39  order by count(*) asc
40  limit 1;
41

```

Result Grid							
Filter Rows:							
Export:   Wrap Cell Content:   Fetch rows:							
date(req_time)							
▶ 2024-01-03							

```

42
43 -- Q6
44 • select fname, lname, national_id, wallet
45 from Driver join Accept_Payment on Driver.national_id = Accept_Payment.driver_national_id
46 where start_trip_time >= '2024-01-01 00:00:00' and start_trip_time < '2024-01-02 00:00:00'
47 group by fname, lname, national_id, wallet
48 having count(*) >= 2;
49
50
51 -- Q7
52 • select fname, lname, sum(Transactions.cost)
53 from Customer, Trip, Accept_Payment, Transactions

```

Result Grid				
	fname	lname	national_id	wallet
▶	karim	sarabi	65987412	20000

```

50
51 -- Q7
52 • select fname, lname, sum(Transactions.cost)
53 from Customer, Trip, Accept_Payment, Transactions
54 where Customer.phone_number = Trip.cphone_number and Trip.id = Accept_Payment.trip_id and Trip.cphone_number = Accept_Payment.cphone_number
55 and Transactions.id = Accept_Payment.transaction_id and timediff(end_trip_time, start_trip_time) > '02:00:00'
56 and start_trip_time >= date_sub(now(), interval 1 week)
57 group by phone_number, fname, lname;
58
59
60 -- Q8
61 • select fname, lname, age
62 from Driver, BaxiBar_Car, Accept_Payment
63 where Driver.national_id = BaxiBar_Car.driver_national_id and car_type = 'truck' and disability_type = 'deafness'
64 and fuel_type = 'gas' and Driver.national_id = Accept_Payment.driver_national_id and age > all(select avg(age) as age
65 from Driver)

```

Result Grid			
	fname	lname	sum(Transactions.cost)
▶	salim	salimi	32000