

بسم الله الرحمن الرحيم

پروژه درس پایگاه داده فاز دوم : سیستم تاکسیرانی

گروه شماره 9

محمد افضل زاده

محمد شریفی

محمد امین افشارین شانديز

امير ارشيا درودي

## نصب دیتابیس:

نصب MySQL بر روی Ubuntu به سادگی اجرای چند دستور ساده در Terminal است که با توجه به منبع داده شده (<https://www.mysqltutorial.org/install-mysql-ubuntu/>) به صورت زیر است :

1- اجرای دو دستور زیر برای دانلود و نصب repository package های مورد نیاز:

```
wget -c https://repo.mysql.com//mysql-apt-config\_0.8.13-1\_all.deb
```

```
sudo dpkg -i mysql-apt-config_0.8.13-1_all.deb
```

2- دانلود MySQL

```
sudo apt-get install mysql-server
```

3- نصب امن MySQL

```
sudo mysql_secure_installation .a
```

4- اجرای MySQL و بعد از زدن دستور زیر پسورد را وارد میکنیم

```
sudo mysql -u root -p .a
```

5- نصب Phpmyadmin برای صحت در کار با استفاده از سایت

<https://www.digitalocean.com/community/tutorials/how-to-install-and-secure-phpmyadmin-on-ubuntu-20-04>

6- در انتها اجرای دستور `mysqldump -u root -p db_databasePrgP2G9 > ../dump.sql` برای خروجی گرفتن از دیتابیس

## نحوه ساخت دیتابیس :

ابتدا با استفاده از دستور زیر دیتابیس را میسازیم و این دستور به گونه ای است که اگر از قبل دیتابیزی با همین نام موجود باشد آنرا حذف و دیتابیس جدید را میسازد و و برای پشتیبانی دیتابیس از فارسی از utf8 استفاده می کنیم :

```
DROP DATABASE IF EXISTS `db_databasePrgP2G9`;  
CREATE DATABASE `db_databasePrgP2G9` DEFAULT CHARACTER SET utf8mb4 COLLATE  
utf8mb4_general_ci;
```

با توجه به نمودار ER و این که اینکه ارث بری از نوع کامل و Overlab هست یک از مواردی که میتوان این ارث بری را با استفاده از آن پیاده سازی کرد پس سه جدول Person و Passenger و Driver را به صورت زیر با صفات ساده شان می سازیم .

جدول افراد :

```
CREATE TABLE `db_databasePrgP2G9`.`person` ( `ID` INT NOT NULL AUTO_INCREMENT  
, `Phone` CHAR(11) NOT NULL , `FullName` VARCHAR(70) NOT NULL , `Gender`  
ENUM('Male','Female') NULL , `Pass` VARCHAR(70) NOT NULL , PRIMARY KEY  
(`ID`), UNIQUE `PhonePerson` (`Phone`(11))) ENGINE = InnoDB;
```

در جدول افراد هر فرد یک ID یکتا دارد که خود پایگاه داده به صورت خودکار برای هر فردی که در سیستم ثبت نام می کند می سازد. همچنین صفت شماره تلفن یک کلید کاندید هست چون نمیتواند یک شماره تلفن مربوط به دو نفر باشد .

جدول مسافر ها :

```
CREATE TABLE `db_databasePrgP2G9`.`passenger` ( `ID` INT NOT NULL , PRIMARY  
KEY (`ID`)) ENGINE = InnoDB;
```

از آنجایی که هر مسافر به جز اطلاعات مربوط به افراد اطلاعات دیگری را ذخیره نمی کند پس با داشتن ID فرد که کلید اصلی این جدول است و داشتن رابطه با جدول افراد می توان به مشخصات مسافر را داشت که این رابطه را با دستور زیر ایجاد می کنیم :

```
ALTER TABLE `db_databasePrgP2G9`.`passenger` ADD CONSTRAINT  
`Passenger_To_Person` FOREIGN KEY (`ID`) REFERENCES `person` (`ID`) ON DELETE  
CASCADE ON UPDATE RESTRICT;
```

البته واضح هست که اگر فردی از جدول Person حذف شود باید از جدول Passenger هم حذف شود. و با دستور (ON UPDATE RESTRICT) از به روزرسانی ID مسافر ها در جدول افراد جلوگیری میکند .

از دستور زیر برای ایجاد رابطه یک به چند معرف های مسافر ها استفاده میکنیم (هر مسافر حداکثر یک معرف دارد و هر مسافر می تواند معرف هر چند نفر باشد). همچنین دقت میکنیم که با حذف معرف ، معرفی شده ها حذف نشوند و معرفشان NULL شود.

```
ALTER TABLE `db_databasePrgP2G9`.`passenger` ADD `IdendifierID` INT NULL
AFTER `ID`; -- for relation identify
ALTER TABLE `db_databasePrgP2G9`.`passenger` ADD CONSTRAINT
`Identify_passenger` FOREIGN KEY (`IdendifierID`) REFERENCES
`passenger`(`ID`) ON DELETE SET NULL ON UPDATE RESTRICT;
```

جدول راننده ها :

```
CREATE TABLE `db_databasePrgP2G9`.`driver` ( `ID` INT NOT NULL ,
`NationalNumber` CHAR(10) NOT NULL , `FatherName` VARCHAR(50) NOT NULL ,
`Addres` VARCHAR(535) NOT NULL , `Birthday` DATE NOT NULL , `img`
VARCHAR(535) NOT NULL , PRIMARY KEY (`ID`), UNIQUE
`NationalNumber_driver_idx` (`NationalNumber`(10))) ENGINE = InnoDB;

ALTER TABLE `db_databasePrgP2G9`.`driver` ADD CONSTRAINT `ID_Person` FOREIGN
KEY (`ID`) REFERENCES `person`(`ID`) ON DELETE CASCADE ON UPDATE RESTRICT;
-- FOREIGN KEY Driver to Person
```

مشابه جدول مسافر ها این جدول یک کلید اصلی ID دارد که به جدول افراد وصل می شود (کلید خارجی هم هست). این جدول اطلاعات دیگر راننده را هم دارد با نوع مورد نیاز و از آنجایی که افراد هر یک ، یک کد ملی یکتا دارند پس این صفت یک کلید کاندید هست.

رابطه بین راننده و ماشینش و گواهینامه اش یک رابطه یک به یک و اجباری هست پس میتوان با دستور زیر صفات ماشین و گواهینامه راننده را به جدول راننده ها افزود همچنین صفت پلاک به طور کامل و شماره گواهینامه یک کلید کاندید هست.

```
-- ADD Car
ALTER TABLE `db_databasePrgP2G9`.`driver` ADD `Car_PNum1` TINYINT NOT NULL
AFTER `img`, ADD `Car_PChr` CHAR(1) NOT NULL AFTER `Car_PNum1`, ADD
`Car_PNum2` SMALLINT NOT NULL AFTER `Car_PChr`, ADD `Car_CNum` SMALLINT NOT
NULL AFTER `Car_PNum2`, ADD `Car_EndInsurance` DATE NOT NULL AFTER
`Car_CNum`, ADD `Car_Model` VARCHAR(70) NOT NULL AFTER `Car_EndInsurance`,
ADD `Car_Color` VARCHAR(70) NOT NULL AFTER `Car_Model`;
ALTER TABLE `db_databasePrgP2G9`.`driver` ADD UNIQUE(`Car_PNum1`, `Car_PChr`,
`Car_PNum2`, `Car_CNum`);
-- end Of Add Certificate

-- ADD Certificate
ALTER TABLE `db_databasePrgP2G9`.`driver` ADD `Certificate_Number` INT NOT
NULL AFTER `Car_Color`, ADD `Certificate_IssueDate` DATE NOT NULL AFTER
`Certificate_Number`, ADD `Certificate_ValidityDuration` DATE NOT NULL AFTER
`Certificate_IssueDate`, ADD UNIQUE `Certi_Num_idx` (`Certificate_Number`);
-- End Certificate
```

راننده ها هم مانند مسافر ها معرف دارند که رابطه 1-N هست و مثل مسافر ها باید دقت کنیم که با حذف معرف ، معرفی شده ها حذف نشوند و معرفشان NULL شود.

```
ALTER TABLE `db_databasePrgP2G9`.`driver` ADD `IdendifierID` INT NULL AFTER
`img`;
```

```
ALTER TABLE `db_databasePrgP2G9`.`driver` ADD CONSTRAINT `IdentifyDriver`
FOREIGN KEY (`IdendifierID`) REFERENCES `driver`(`ID`) ON DELETE SET NULL ON
UPDATE RESTRICT;
```

جدول Discount و SavedLocation :

این دو موجودیت های ضعیف هستند پس کلید اصلی آنها کلید اصلی جدول مسافر ها همراه با کلید جزئی این موجودیت ها است.  
همچنین باید یک ارتباط بین این جدول ها با جدول مسافر ها باشد .

```
-- CREATE TABLE discount
CREATE TABLE `db_databasePrgP2G9`.`discount` ( `ID` INT NOT NULL , `Code` INT
NOT NULL , `Percent` DECIMAL(4,2) NOT NULL , `Max` BIGINT NULL DEFAULT NULL ,
PRIMARY KEY (ID , Code)) ENGINE = InnoDB;

ALTER TABLE `db_databasePrgP2G9`.`discount` ADD CONSTRAINT
`Passenger_Discount_FK` FOREIGN KEY (`ID`) REFERENCES `passenger`(`ID`) ON
DELETE CASCADE ON UPDATE RESTRICT; -- foreign Key To Passenger

-- end Of Discount

-- CREATE TABLE SavedLocation
CREATE TABLE `db_databasePrgP2G9`.`savedLoc` ( `ID` INT NOT NULL , `LocId`
INT NOT NULL , `NameOfLoc` VARCHAR(70) NOT NULL , `Loc` POINT NOT NULL ,
`Address` VARCHAR(170) NULL DEFAULT NULL , PRIMARY KEY (`ID` , `LocId`))
ENGINE = InnoDB;

ALTER TABLE `db_databasePrgP2G9`.`savedloc` ADD CONSTRAINT
`Passenger_ImpLoc_FK` FOREIGN KEY (`ID`) REFERENCES `passenger`(`ID`) ON
DELETE CASCADE ON UPDATE RESTRICT; -- foreign Key To Passenger

-- end Of SavedLocation
```

و چون باید صفت Code جدول Discount و صفت LocId جدول SavedLoc باید مرجع کلید های خارجی جدول Trip باشند پس با دستور زیر index گذاری می کنیم :

```
ALTER TABLE `db_databasePrgP2G9`.`discount` ADD INDEX(`Code`);

ALTER TABLE `db_databasePrgP2G9`.`savedloc` ADD INDEX(`LocId`);
```

جدول Trip :

این جدول عملاً رابطه بین مسافر ها و راننده ها برای رفتن به سفر هست . که با توجه به ER رسم شده با اجرای دستورات زیر ساخته می شوند . همچنین چون روابط 1 – N هستند کلید های خارجی را در این سمت قرار می دهیم.

```
CREATE TABLE `db_databasePrgP2G9`.`trip` ( `TripId` BIGINT NOT NULL
AUTO_INCREMENT , `PassId` INT NOT NULL , `DriverId` INT NOT NULL , `Amount`
BIGINT NOT NULL , `PaymentMethod` ENUM('نقدی','اعتباری') NOT NULL ,
`DiscountCode` INT NULL , `StartTime` DATETIME NOT NULL DEFAULT
```

```
CURRENT_TIMESTAMP , `EndTime` DATETIME NULL DEFAULT NULL , `StartLoc` POINT  
NULL DEFAULT NULL , `EndLoc` POINT NULL DEFAULT NULL , `StartSavedLoc` INT  
NULL DEFAULT NULL , `EndSavedLoc` INT NULL DEFAULT NULL , `DriverRate`  
TINYINT NULL DEFAULT NULL , `PassRate` TINYINT NULL DEFAULT NULL , PRIMARY  
KEY (`TripId`)) ENGINE = InnoDB;
```

```
ALTER TABLE `db_databasePrgP2G9`.`trip` ADD CONSTRAINT `Pass_To_Trip_FK`  
FOREIGN KEY (`PassId`) REFERENCES `passenger`(`ID`) ON DELETE RESTRICT ON  
UPDATE RESTRICT;
```

```
ALTER TABLE `db_databasePrgP2G9`.`trip` ADD CONSTRAINT `Driver_To_Trip_FK`  
FOREIGN KEY (`DriverId`) REFERENCES `driver`(`ID`) ON DELETE RESTRICT ON  
UPDATE RESTRICT;
```

```
ALTER TABLE `db_databasePrgP2G9`.`trip` ADD CONSTRAINT `StartSavedLoc_Trip`  
FOREIGN KEY (`StartSavedLoc`) REFERENCES `savedloc`(`LocId`) ON DELETE  
RESTRICT ON UPDATE RESTRICT;
```

```
ALTER TABLE `db_databasePrgP2G9`.`trip` ADD CONSTRAINT `EndSavedLoc_Trip`  
FOREIGN KEY (`EndSavedLoc`) REFERENCES `savedloc`(`LocId`) ON DELETE RESTRICT  
ON UPDATE RESTRICT;
```

```
ALTER TABLE `db_databasePrgP2G9`.`trip` ADD CONSTRAINT `Discount_Trip`  
FOREIGN KEY (`DiscountCode`) REFERENCES `discount`(`Code`) ON DELETE RESTRICT  
ON UPDATE RESTRICT;
```