



دانشگاه تهران  
پردیس دانشکده های فنی  
مهندسی نقشه برداری و اطلاعات دانشکده  
مکانی



## گزارش تمرین نهایی درس شبکه های کامپیوتری

مهیار داوطلب

استاد: دکتر زارع زردینی

## بخش MODEL

بخش مدل شامل تعریف کلاس ها است . در این بخش ۲ کلاس ساخته شد . کلاس اول car.cs است که مقادیر فیلد های دیتابیس در آن آورده می شود.

کلاس دوم CarRepository است که در این برنامه برای ارتباط با پایگاه داده SQLite به نام "car.db" استفاده می شود. این کلاس دارای متدهای مختلفی است که امکان انجام عملیات مانند درج (Insert)، به روز رسانی (Update)، حذف (Delete) و دریافت اطلاعات (Retrieve) را بر روی جدول "Cars" در پایگاه داده فراهم می کند.

در این کلاس، متدهای زیر تعریف شده اند:

**UpdateCar:** این متد برای به روز رسانی مشخصات یک ماشین در جدول "Cars" استفاده می شود. این متد با دریافت یک شیء از نوع Car، مکان و قیمت جدید را برای ماشین مشخص شده به روز می کند.

**GetDescriptiveInfo:** این متد برای دریافت اطلاعات توصیفی یک ماشین با استفاده از آن استفاده می شود. این متد یک دیکشنری با کلید-مقدارهایی مانند نام، توضیحات، تاریخ تولید، مکان و قیمت را برای ماشین مشخص شده باز می گرداند.

**InsertCar:** این متد برای درج یک ماشین جدید در جدول "Cars" استفاده می شود. با دریافت یک شیء از نوع Car، اطلاعات آن را در جدول ذخیره می کند.

**GetAllCars**: این متد برای دریافت لیستی از تمام ماشین‌ها در جدول "Cars" استفاده می‌شود. این متد لیستی از شیء‌های نوع Car را بازمی‌گرداند.

**GetCarById**: این متد برای دریافت یک ماشین بر اساس شناسه آن استفاده می‌شود. با دریافت شناسه ماشین، اطلاعات آن را از جدول "Cars" بازیابی کرده و برای شما بازمی‌گرداند.

**DeleteCar**: این متد برای حذف یک ماشین از جدول "Cars" استفاده می‌شود. با دریافت یک شیء از نوع Car، ماشین مشخص شده را از جدول حذف می‌کند.

**GetCarCountByLocation**: این متد برای دریافت تعداد خودروها بر اساس مکان آن‌ها استفاده می‌شود. این متد یک دیکشنری با کلید-مقدارهایی مانند مکان و تعداد ماشین‌ها در هر مکان را بازمی‌گرداند.

**GetAveragePriceByLocation**: این متد برای دریافت میانگین قیمت خودروها بر اساس مکان آن‌ها استفاده می‌شود. این متد یک دیکشنری با کلید-مقدارهایی که شما متدهای **GetAllCars** و **InsertCar** را به این کلاس اضافه کرده‌اید از پیاده‌سازی شما بستگی دارد و ممکن است در کد شما تعریف شده باشد. اما اگر قبلاً این متدها را تعریف نکرده‌اید، می‌توانید آن‌ها را به کلاس **CarRepository** اضافه کنید:

در این تغییرات، متدهای **GetAllCars** و **InsertCar** به کلاس **CarRepository** اضافه شده‌اند. متد **InsertCar** یک شیء از نوع Car را به عنوان ورودی می‌گیرد و اطلاعات آن را در جدول "Cars" درج می‌کند. متد **GetAllCars** تمام ماشین‌ها را از جدول بازیابی کرده و یک لیست از شیء‌های Car را بازمی‌گرداند.

## بخش Controller

در الگوی طراحی MVC (Model-View-Controller)، کنترل کننده یا "Controller" یکی از مؤلفه‌های اصلی است. کنترل کننده وظیفه مدیریت جریان کاربری، انجام عملیات واسط کاربری و هماهنگی بین مدل (Model) و نمایش (View) را بر عهده دارد.

وظایف اصلی کنترل کننده عبارتند از:

۱. دریافت واسط کاربری: کنترل کننده ورودی‌ها و اعمال کاربر را از طریق واسط کاربری (مانند فرم‌ها، دکمه‌ها و لینک‌ها) دریافت می‌کند.

۲. بروزرسانی مدل: با توجه به عملیات کاربر و اطلاعات دریافتی، کنترل کننده مدل را بروزرسانی می‌کند. این شامل خواندن و نوشتن اطلاعات در مدل و انجام عملیات مربوط به آن است.

۳. انتخاب نمایش: کنترلر تصمیم می‌گیرد کدام نمایش را به کاربر نمایش دهد. این شامل انتخاب صفحه‌ها، نمایش اطلاعات مربوطه و تغییر وضعیت نمایش است.

۴. مدیریت روابط: کنترل کننده مسئولیت مدیریت روابط بین مدل و نمایش را دارد. این شامل انتقال داده‌ها از مدل به نمایش و برعکس، ارسال پیام‌ها و رویدادها بین مدل و نمایش است.

کلاس CarController یک کنترل کننده است که از کلاس پایه 'Controller' در فضای نام System.Web.Mvc گرفته می‌شود. این کلاس برای مدیریت عملیات مربوط به خودروها در برنامه استفاده می‌شود. در زیر توضیحی در مورد Action‌های اصلی این کلاس آورده شده است:

۱. `public ActionResult Index`: این متد به عنوان عملیات `Index` کنترل کننده عمل می کند. در اینجا، تمام خودروها از `carRepository` دریافت شده و به صفحه نمایش مربوطه (`View`) ارسال می شود.

۲. `public ActionResult Create`: این متد برای نمایش صفحه ای برای ایجاد خودرو جدید استفاده می شود.

۳. `[HttpPost] public ActionResult Store(Car car)`: این متد برای ذخیره خودرو جدیدی که توسط کاربر ارسال شده استفاده می شود.

۴. `public ActionResult Edit(int id)`: این متد برای نمایش صفحه ویرایش خودرو بر اساس شناسه (`id`) استفاده می شود. ابتدا خودرو با استفاده از `carRepository` بر اساس شناسه دریافت می شود و سپس به صفحه ویرایش منتقل می شود.

۵. `[HttpPost] public ActionResult Update(Car car)`: این متد برای بروزرسانی خودروی ویرایش شده استفاده می شود. اگر اعتبارسنجی مدل معتبر باشد، خودرو با استفاده از `carRepository` به روزرسانی می شود و کاربر به عملیات `Index` هدایت می شود. در غیر این صورت، کاربر به صفحه ویرایش برگردانده می شود.

۶. `public ActionResult Delete(int id)`: این متد برای نمایش صفحه حذف خودرو بر اساس شناسه است (`id`) استفاده می شود. ابتدا خودرو با استفاده از `carRepository` بر اساس شناسه دریافت می شود و سپس به صفحه حذف خودرو منتقل می شود. اگر خودرو یافت نشود، به صفحه خطای ۴۰۴ (`HttpNotFound`) هدایت می شود.

۷. `[HttpPost] public ActionResult ConfirmDelete(int id)`: این متد برای تأیید حذف خودرو استفاده می‌شود. ابتدا خودرو با استفاده از `carRepository` بر اساس شناسه دریافت می‌شود و سپس حذف می‌شود. سپس کاربر به عملیات `Index` هدایت می‌شود.

## بخش view

در این بخش دو فولدر به نام های `Home` و `Car` ساخته شد و هر کدام شامل نماهایی می باشد. درپوشه `home` نماهای `search` - `downloaddata` و `dashboard` قرار دارد که در `dashboard` اطلاعاتی در مورد تعداد خودرو ها در شهرهای مختلف - یک چارت در مورد میانگین قیمت ها در هر شهر و یک جدول آورده شده است. از طرفی در `car folder` نماهای مربوط به `create` - `Delete` - `EDIT` و `index` قرار دارد .

## بخش Login

این بخش در پروژه طوری طراحی شده است که در صورت درست بودن رمز و پسورد از جانب کاربر `login` صورت میگیرد و در صورت غلط بودن دسترسی کاربر از سامانه قطع می شود . همچنین برای ورود به بخش `Edit` و `Dashboad` یک سطح دسترسی از طرف `admin` با پسورد ۱۲۳۴ قرار داده شده است تا نماهای مربوطه قابل ملاحظه گردد.

My Website

Home

About

Contact

DownloadData

dashboard

Search by name

Search by location

Search

Login

Username:

admin

Password:

\*\*\*\*

Login

Login successful!

My Website

Home

About

Contact

DownloadData

dashboard

Search by name

Search by location

Search

Login

Username:

mahyar

Password:

\*\*\*\*\*

Login

Invalid username or password

## بخش Search

در این قسمت کاربر میتواند از دو طریق خودرو خودروی مورد نظر خود را پیدا کند.  
و نتیجه به صورت زیر برای کاربر نمایش داده می شود

### Login

Username:

Password:

### Search Results

- changan  
Name: changan  
Location: rasht  
Price: 2000  
ProductionDate: 1393  
Price is outside the specified range.

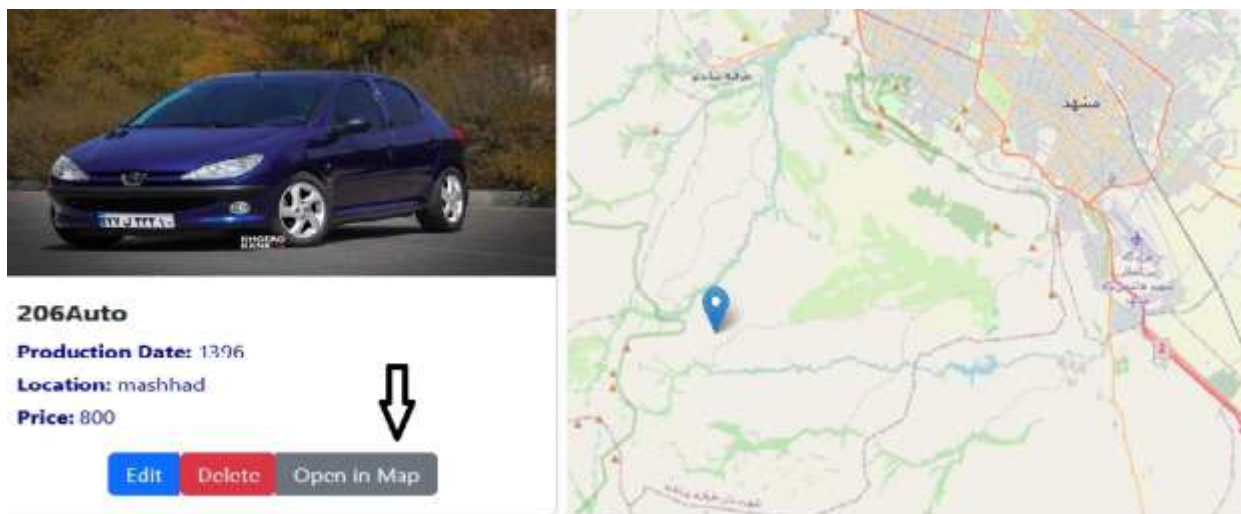
سرچ از طریق location نیز امکان پذیر است و کاربر می تواند تمام خودروهای محدوده ی شهر خود را ببیند. برای مثال برای شهر مشهد دو خودرو با مشخصات زیر پیدا شد

### Search Results

- 206Auto  
Name: 206Auto  
Location: mashhad  
Price: 800  
ProductionDate: 1396  
Price is outside the specified range.
- mazda3Auto  
Name: mazda3Auto  
Location: mashhad  
Price: 680  
ProductionDate: 1403

کاربر می تواند location خودرو را بر روی نقشه نظاره گر باشد.





## بخش EDIT

در این بخش کاربر می تواند بعضی اطلاعات را ویرایش کند. در این قسمت قبل از ویرایش یک سطح دسترسی نیاز است و بعد از تایید کاربر می تواند اقدام به ویرایش کند.



206

Production Date: 1397

Location: kerman

Price: 900



Edit Delete Open in Map

⇒ Login

Username:

admin

Password:

....

Login

### Access Denied

You do not have permission to access this page.



206

Production Date: 1397

Location: kermanshah

Price: 900

Edit Delete Open in Map

### Edit Car

Location

kermanshah

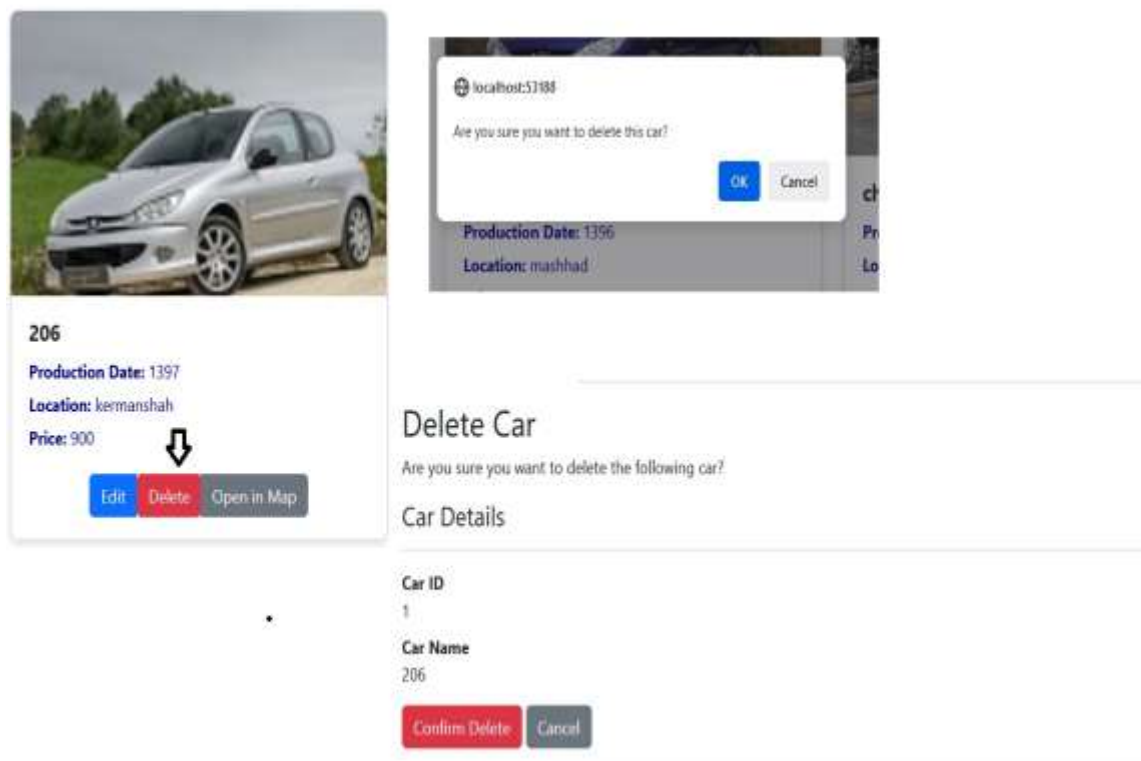
Price

900

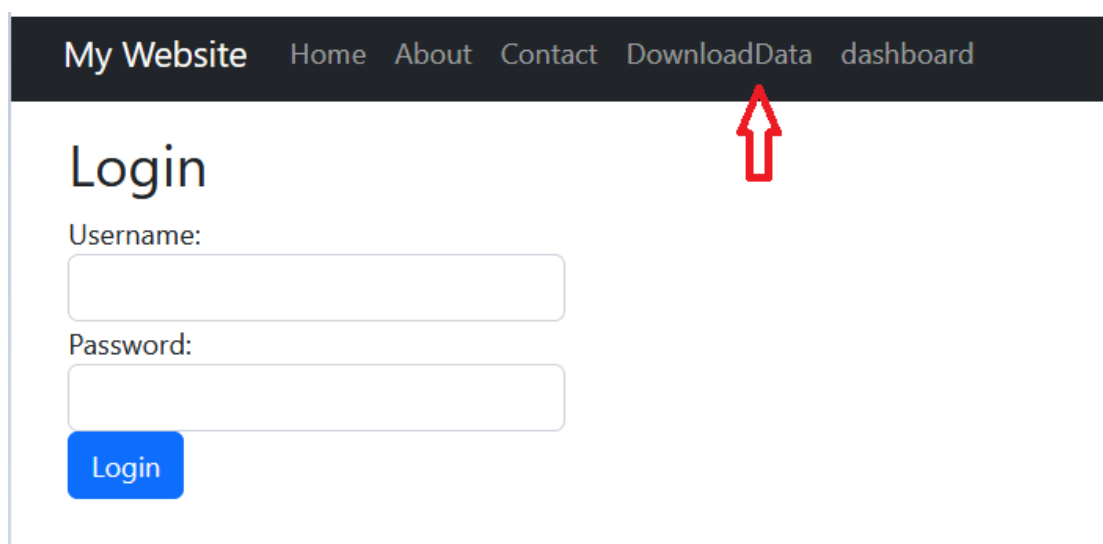
Update

## بخش Delete

در این قسمت کاربر می تواند خودروی مورد نظر را حذف کند



## بخش DownloadData



بعد از کلیک بر روی این گزینه یک اطلاعات از نوع خودروهای موجود در سایت به همراه location و قیمت در قالب یک فایل اکسل برای دانلود آماده است.

	A	B	C	D
9	Peugeot-2	kish	1800	
10	prado	sari	8000	
11	pride	lahijan	250	
12	Rio	yazd	3300	
13	zantia	semnan	400	
14	SsangYong	hormozgan	2100	
15	tigo	shiraz	900	
16	cerato	sabzevar	500	
17	tigo	tonekabon	1200	
18	prado	bandare-abb	12000	
19	landcrouzer	rasht	80000	
20	jack	tehran	1200	
21	porsche	karaj	5000	
22	samand	ghazvin	200	
23	G-class	kermanshah	10000	
24	nissan_juk	kish	2600	

## بخش dashboard

در این بخش یک سری اطلاعات در قالب چارت و نمودار و جدول به کاربر نمایش داده می شود که در قسمت زیر چند مورد از اطلاعات این بخش آورده شده است.

## Login

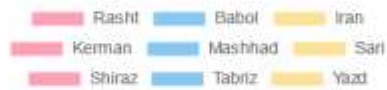
Username:

Password:

Login



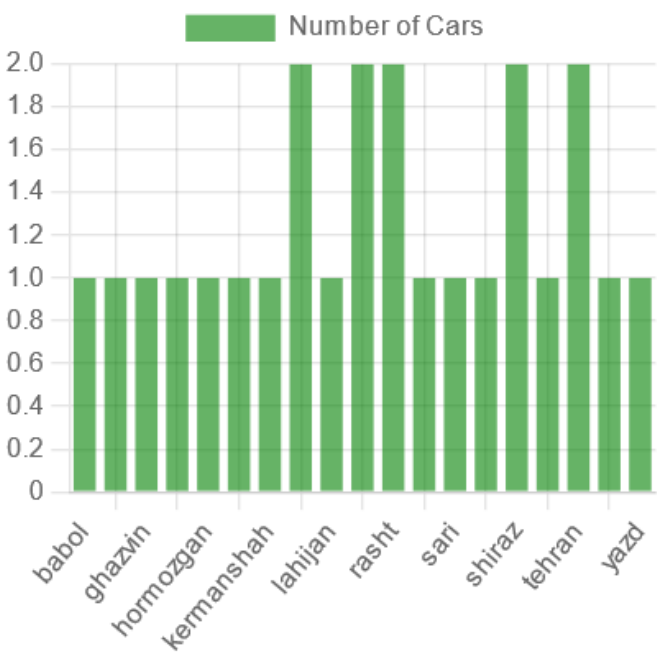
## Average Price Chart



# Average Price of the cars in cities

Location	Average Price
Rasht	2200
Babol	520
Iran	1000
Kerman	350
Mashhad	800
Sari	1015
Shiraz	1200
Tabriz	750
Yazd	450

# Car Count Chart



## بخش اضافه کردن خودرو

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

Add your Car here

### Login

Username:

Password:

Login

Name mazda3 Location shiraz Price 900 Description good Add Car