

✓ ویدئو عالی با موضوع migration در یوتیوب: <https://www.youtube.com/watch?v=neSHAWdE44c>

✓ داکيومنت سایت مرجع در مورد migration: <https://laravel.com/docs/5.1/quickstart>

✓ داکيومنت سایت مرجع برای get کردن <https://laravel.com/docs/5.7/routing>

✓ داکيومنت سایت مرجع برای post کردن <https://laravel.com/docs/5.8/eloquent>

✓ مایگريشن: مفهوم مشترک بین شبکه و نرم افزار که کاملا دو معنی جدا دارند.

○ در شبکه وقتی یک سرور را مثلا ایمیل سرور مایکروسافت که به نام اکسچینج سرور است می‌خواهید به نسخه ای بالاتر منتقل کنید یا سرور ها را جابجا کنید مایگريشن می‌گویند. چگونه دیتا ها و دیتابیس ها و کانفیگوریشن ها منتقل شود و نرم افزار هم ارتقا یابد و حتی سرور فیزیکی جابجا شود ولی همه چیز سر جایش باشد.

○ در برنامه نویسی ابتدا باید با مفهوم ORM آشنا شوید. یک مشکلی که همیشه در برنامه نویسی وجود داشت این بود که همه چیز شی گرا بود ولی دیتابیس خیر. کم کم متوجه شدند که خود دیتابیس را نمیتوان شی گرا کرد اما رفتار آن را میتوان شبیه شی گرای نمود. دیتابیس که شیگرا بشه یعنی شما هیچ کد استاتیک نمیزنید و همون کلاس برنامه نویسی همه ی کارها رو انجام میدهد. حالا وقتی در لاراوول فایل مایگريت میسازید چند کار انجام میدهید. اول اینکه میتونید بفهمید روی هر سرور کدام نسخه نصب شده و تا هر جا که جلو رفتید بتوانید اتوماتیک توسط کامپیوتر آنرا ادامه دهید و بروز رسانی کنید. با خیال راحت و بدون نگرانی. زیرا در هر مرحله از کار برنامه نویسی تست شده و مطمئن هستید که اگر این تغییرات از هر جایی به بعد اتفاق بیافتد دیتابیس و کد همسان و هماهنگ و هم نسخه حرکت میکنند

✓ درباره ی get و post و انواع روش های صدا زدن وب سرویس ها بهتر است از postman استفاده کنید و کد تولید شده را استفاده کنید.

✓ دستور زیر که برای ایجاد MVC و migration است را در پروژه لاراوولی که در جلسات پیش ساختی بزن:

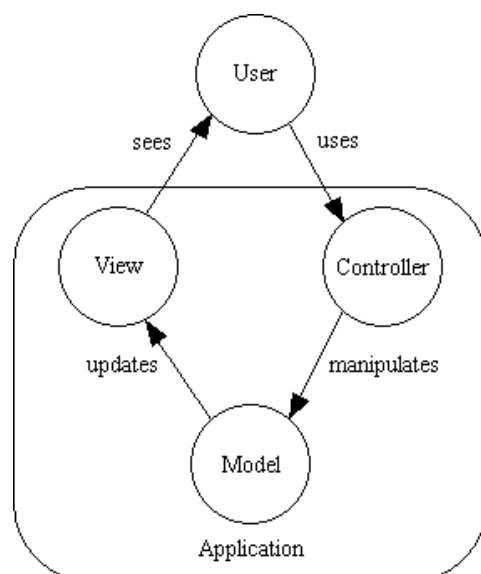
اسم مدل

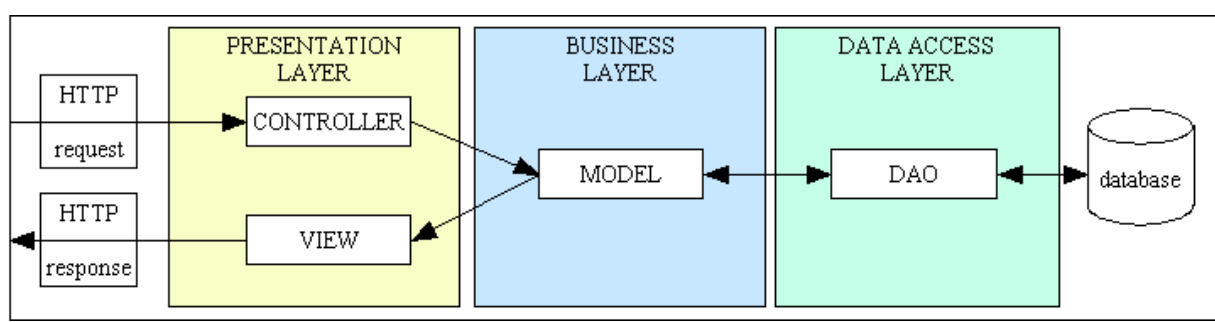
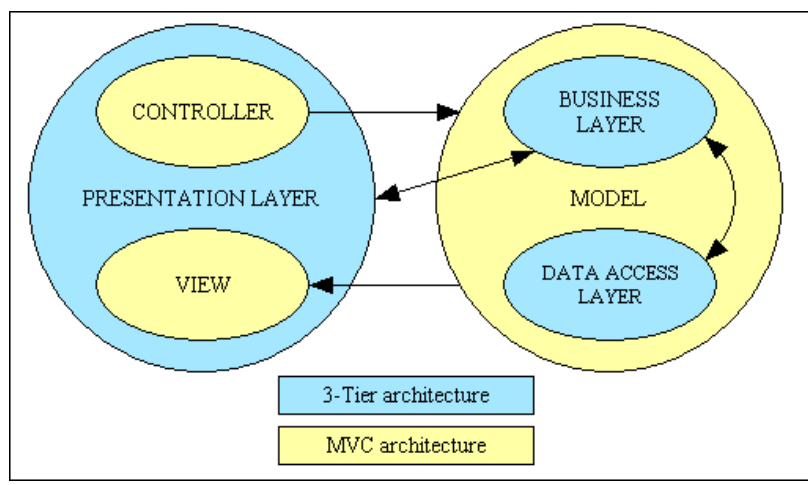
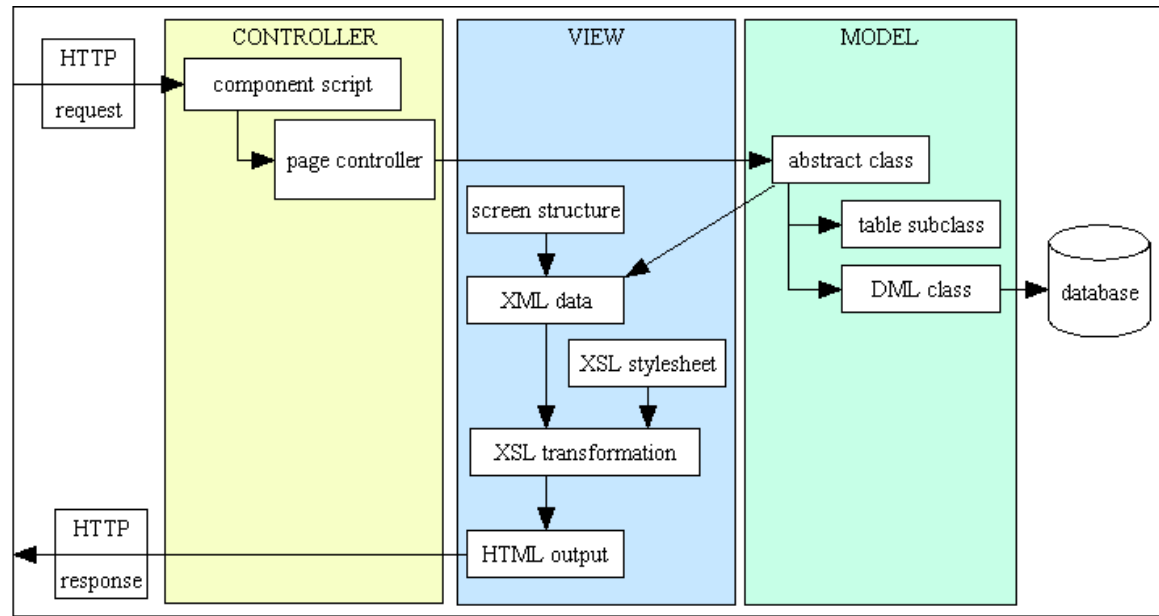
```
php artisan make:model Todo -a
```

به جای -a میشه --a هم بزنیم

✓ :MVC

- مدل، رابط کنترلر با جدول است. یعنی مدل، راهنمای کنترلر است که کنترلر بفهمه جدول من چه فیلدهایی داره.
- در زبانهای دیگه باید MVC را بسازی. ولی در لاراوول، Eloquent خودش واسطه ی اتصال به بانک و شناسایی فیلدها را برات میسازه.





- ▼ config
 - Conf.php
- ▼ controller
 - ControllerVoiture.php
 - routeur.php
- ▼ model
 - Model.php
 - ModelVoiture.php
- ▼ view
 - ▼ voiture
 - create.php
 - detail.php
 - error.php
 - list.php

```

1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8
9 DB_CONNECTION=mysql
10 DB_HOST=127.0.0.1
11 DB_PORT=3306
12 DB_DATABASE=laravel
13 DB_USERNAME=root
14 DB_PASSWORD=
15
16 BROADCAST_DRIVER=log
17 CACHE_DRIVER=file
18 QUEUE_CONNECTION=sync
19 SESSION_DRIVER=file
20 SESSION_LIFETIME=120
21
22 REDIS_HOST=127.0.0.1
23 REDIS_PASSWORD=null
24 REDIS_PORT=6379
25

```

✓ قابلیت‌های کلیدی یک برنامه نویس در تیم حرفه‌ای:

۱. فهم صورت مسئله: چی میخوایم بسازیم (ابزارها : مستند بک لاگ، جلسات اسکرام هفتگی، جلسات پوکر هفتگی، جلسات هر روز صبح)
۲. تقسیم و پیشروی
۳. مشورت
۴. مطالعه
۵. بعد از درک معماری و صورت مساله باید یاد بگیریم چی گوگل کنیم تا روش های مختلف برای پیاده سازی رو پیدا کنیم و تصمیم بگیریم بهترینش کدومه و گوگل کنیم چه اجزایی رو در گیت هاب و اینترنت در اختیار داریم و میخوایم به هم بچسونیم
۶. دیباگ و trace کردن (یعنی خط به خط کد که نوشتیم رو با یک مقدار و سناریوی مشخص تست میکنیم)
۷. Trouble shooting: یعنی مشکلات پیش آمده رو چطوری شناسایی و حل کنم
۸. تعامل و کار تیمی
۹. نظم

✓ Migration یه جور version Controller برای دیتابیس هم محسوب میشه.

✓ دستور migration : <https://laravel.com/docs/5.8/migrations>

o php artisan migrate

2014_10_12_000000_create_users_table.php - twitter_test01 - Visual Studio Code

```

1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  class CreateUsersTable extends Migration
8  {
9      /**
10     * Run the migrations.
11     *
12     * @return void
13     */
14     public function up()
15     {
16         Schema::create('users', function (Blueprint $table) {
17             $table->bigIncrements('id');
18             $table->string('name');
19             $table->string('email')->unique();
20             $table->timestamp('email_verified_at')->nullable();
21             $table->string('password');
22             $table->rememberToken();
23             $table->timestamps();
24         });
25     }

```

فیلدهای جدول ما هستن

✓ نوع داده ای bigIncrement کار کلید اصلی رو انجام میدهد. یعنی خودش یکی یکی اضافه میشه.

✓ گوگل : field type in laravel

<code>\$table->bigIncrements('id');</code>	Auto-incrementing UNSIGNED BIGINT (primary key) equivalent column.
<code>\$table->bigInteger('votes');</code>	BIGINT equivalent column.
<code>\$table->binary('data');</code>	BLOB equivalent column.
<code>\$table->boolean('confirmed');</code>	BOOLEAN equivalent column.
<code>\$table->char('name', 100);</code>	CHAR equivalent column with an optional length.
<code>\$table->date('created_at');</code>	DATE equivalent column.
<code>\$table->dateTime('created_at');</code>	DATETIME equivalent column.
<code>\$table->dateTimeTz('created_at');</code>	DATETIME (with timezone) equivalent column.
<code>\$table->decimal('amount', 8, 2);</code>	DECIMAL equivalent column with a precision (total digits) and scale (decimal digits).
<code>\$table->double('amount', 8, 2);</code>	DOUBLE equivalent column with a precision (total digits) and scale (decimal digits).
<code>\$table->enum('level', ['easy', 'hard']);</code>	ENUM equivalent column.
<code>\$table->float('amount', 8, 2);</code>	FLOAT equivalent column with a precision (total digits) and scale (decimal digits).

✓ Schema: یک شاکله از یک جدول در یک پایگاه داده. یعنی مثلاً نوع داده های جدول، نام ستونها و...

✓ الان باید بتونی هر جدولی را با هر نوع فیلدی بسازی

✓ همه فایلهای باینری هستن (عکس، فیلم، صوت و...)

✓ کد ملی که ۱۰ رقم است رو بهتره در int(10) ذخیره کنیم یا string(10)?

○ `int(10) < string(10)` چون تعداد بیت‌های `int(10)` برای نگهداری هر عدد کمتر از `string(10)` است.

○ ولی اینکه کدام بهتره بستگی به مورد کاربرد ما داره و جواب قاطعی برای آن وجود نداره.

✓ گوگل: every unicode size store: <https://stackoverflow.com/questions/5290182/how-many-bytes-does-one-unicode-character-take>

✓ میخوایم تویتر بسازیم. طبق سناریو جلسه قبلی چی نیاز داریم؟

○ یه برنامه موبایل داری که به کنترلر (وب سرویس) شما وصل میشه و اون وب سرویس وصل میشه به بانک

○ یادتون هست که وقتی وردپرس به عنوان سرور به ما خدمت میداد، خودش براتون وب سرویسها (rest api) رو ایجاد میکرد و بهتون api میداد و شما با پستمن، API های وردپرس را صدا میدادی.

○ یادآوری:

▪ وردپرس نصب کردیم

▪ با پستمن وب سرویس را صدا زدیم .. در حالت پیشفرض میتونی با وب سرویس posts ها و نوشته ها رو بخوانی.

▪ با نصب pods در وردپرس، میتونی هر entity جدیدی که دلت خواست بسازی مثلا یادت باشد quote را ساختیم و یه فیلدهایی را اضافه کردیم و رکوردهایی در اون ساختیم و خواندیم. (Set get)

○ الان با لاراول بانک را ساختیم و میخوایم یه وب سرویس ایجاد کنیم و با پستمن صداش بزنی و بگیریم این توییت را بخوان یا بذار توی بانک ت:

▪ صورت مسئله: میخوایم با پستمن یه api رو صدا بزنی. توی اون api یه تابعی داشته باشیم که بره توی بانکمون یه چیزی رو بنویسه.

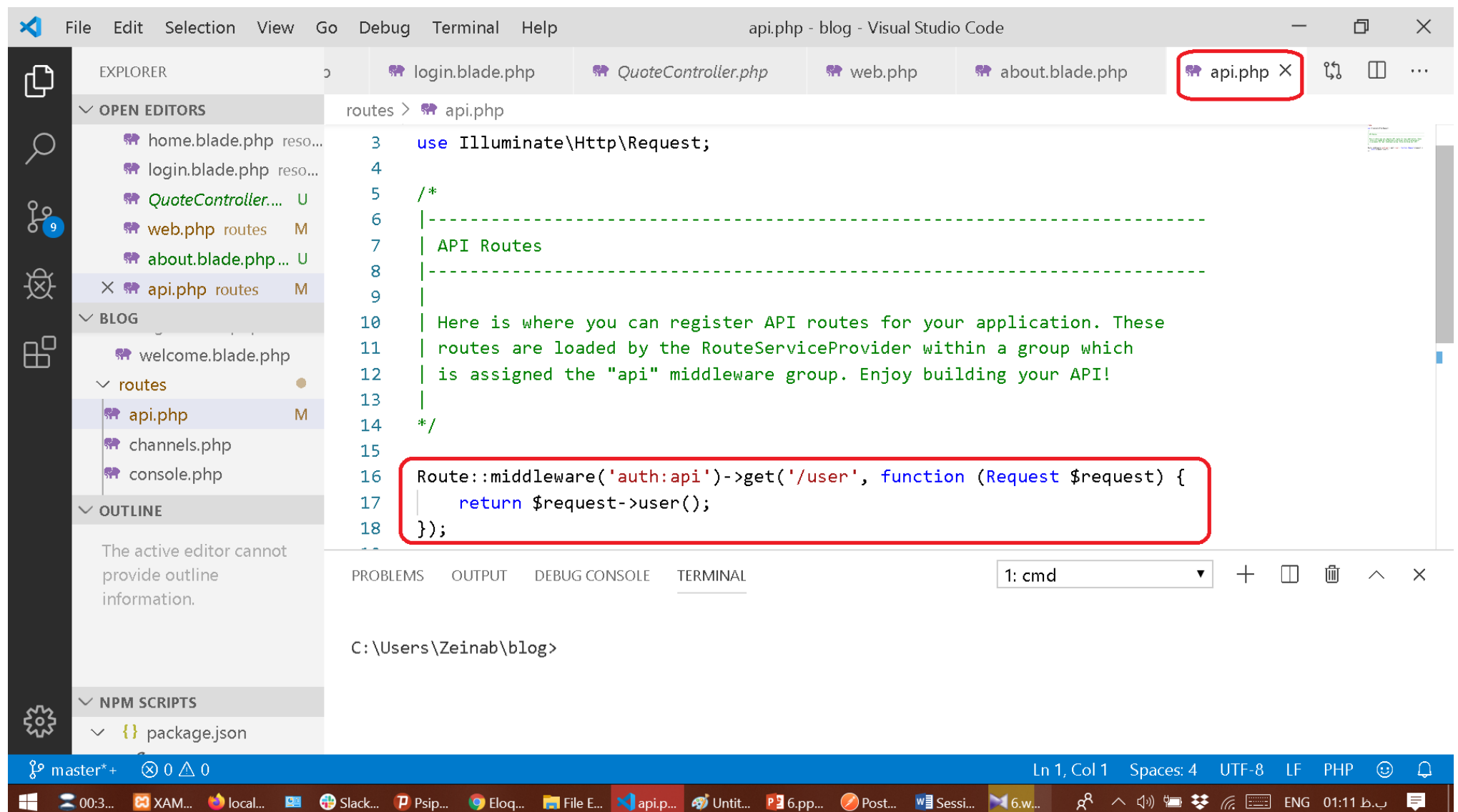
• یعنی الان بانکمون رو ایجاد کردیم و با پستمن میخوایم یه مقادیری رو برای اون api با آدرسی مانند `localhost8000/api/login` پست کنیم. ولی

این آدرس را لاراول هنوز نمیشناسه. راه حلش اینکه بریم در فایل `api.php` و در آنجا، route اش و مسیرها رو تعریف کنیم.

• گوگل: `sample route api.php laravel` (یه سمپل بین و عین اون روانجام بده)

○ نتیجه گوگل: <https://laravel.com/docs/5.7/routing>

میخوایم get کنیم



```
File Edit Selection View Go Debug Terminal Help api.php - blog - Visual Studio Code
EXPLORER
OPEN EDITORS
home.blade.php res...
login.blade.php res...
QuoteController... U
web.php routes M
about.blade.php... U
api.php routes M
BLOG
welcome.blade.php
routes
api.php M
channels.php
console.php
OUTLINE
The active editor cannot provide outline information.
NPM SCRIPTS
package.json
3 use Illuminate\Http\Request;
4
5 /*
6 |-----
7 | API Routes
8 |-----
9 |
10 | Here is where you can register API routes for your application. These
11 | routes are loaded by the RouteServiceProvider within a group which
12 | is assigned the "api" middleware group. Enjoy building your API!
13 |
14 */
15
16 Route::middleware('auth:api')->get('/user', function (Request $request) {
17     return $request->user();
18 });
19
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: cmd
C:\Users\Zeinab\blog>
```


QuoteController.php - blog - Visual Studio Code

```

5 use App\quote;
6 use Illuminate\Http\Request;
7
8 class QuoteController extends Controller
9 {
10     /**
11      * Display a listing of the resource.
12      *
13      * @return \Illuminate\Http\Response
14      */
15     public function index()
16     {
17         $quote=quote::get();
18         return $quote->toJson();
19     }
20 }

```

درون این تابع، چیزی نبود و اینا رو ما نوشتیم.

C:\Users\Zeinab\blog>

در این جلسه به اسم ها حساس نباش. یه مشکلی بود، یه پروژه دیگه رو باز کردم

- حالا دستور php artisan serve رو بزنی تا URL سایت بیادش:

api.php - twitter_test01 - Visual Studio Code

```

7 | API Routes
8 | -----
9 |
10 | Here is where you can register API routes for your application. These
11 | routes are loaded by the RouteServiceProvider within a group which
12 | is assigned the "api" middleware group. Enjoy building your API!
13 |
14 | */
15 |
16 | Route::middleware('auth:api')->get('/user', function (Request $request) {
17 |     return $request->user();
18 | });
19 |
20 |
21 |
22 |

```

Discovered Package: nunomaduro/collision
Package manifest generated successfully.

C:\Users\Zeinab\Projects\twitter_test01>php artisan serve
Laravel development server started: http://127.0.0.1:8000

- URL عکس بالا را در پستمن کپی کن و در ادامه اش بنویس: /api/ quote

○ در فایل web.php هر چیزی که صدا کیزنی از اون ریشه سایتتون شروع میشه. ولی در فایل api هر چی که صدا میزنی با /api/ شروع میشه. چون دو تا آدرس نمیتونه یه جا باشه.

The screenshot shows the Postman interface with a GET request to `http://127.0.0.1:8000/api/qoute`. The response is displayed in the 'Body' tab, showing a single item with the key '1' and the value 'qoute'. The 'Send' button is highlighted with a red box, and the response body is also highlighted with a red box.

حالا میخواهیم جای qoute در عکس بالا مقادیری که در بانکمون هست برگردونه. ولی الان جدول بانکمون خالیه. پس یه دونه post باید بنویسیم و یه دونه get. یعنی برای پست، میخواهیم وقتی به صورت عکس زیر پست کردیم، سرور مقادیر سبز در عکس زیر را بگیره و به بانک اضافه کنه:

The screenshot shows the Postman interface with a POST request to `http://127.0.0.1:8000/api/qoute?title=Zeinab`. The 'Query Params' section shows a parameter with the key 'title' and the value 'Zeinab', both highlighted with green boxes. The 'Send' button is highlighted with a red box.

✓ این، برای get بود.

حالا میخوایم post کنیم

- گوگل: insert new record into laravel by model in controller
 - نتیجه گوگل: <https://laravel.com/docs/5.8/eloquent> (این مطالب در این لینک هست)
 - Eloquent یه پیاده سازی ساده برای کار با پایگاه داده است.
 - این نتیجه گوگل میگه:
 - یه مدل بساز. که ما ساختیم (به نام route)
 - Migration اش را بساز. که ما ساختیم (برای route)
- گوگل: Laravel convert model to json

```
6+ use Illuminate\Http\Request;
7+
8+ class SmsController extends Controller
9+ {
10+     /**
11+      * Display a listing of the resource.
12+      *
13+      * @return \Illuminate\Http\Response
14+      */
15+     public function index()
16+     {
17+         $sms = sms::
18+             // ->orderBy('name', 'desc')
19+             //take(10)
20+             get();
21+
22+         return $sms->toJson();
23+     }
24+     public function post(Request $request)
25+     {
26+         $sms = new sms();
27+         $sms->title=$request["title"];
28+         $sms->title=$request->title;
29+         $sms->save();
30+         return $sms->toJson();
31+     }
32+
33+     /**
```



```
File Edit Selection View Go Debug Terminal Help
api.php - twitter - Visual Studio Code
EXTENSIONS: MARKETPLACE
category:formatters php
We cannot connect to the
Extensions Marketplace at this
time, please try again later.
routes > api.php
1 <?php
2
3 use Illuminate\Http\Request;
4
5 /*
6 |-----|
7 | API Routes
8 |-----|
9 |
10 | Here is where you can register API routes for your application. These
11 | routes are loaded by the RouteServiceProvider within a group which
12 | is assigned the "api" middleware group. Enjoy building your API!
13 |-----|
14 */
15
16 Route::middleware('auth:api')->get('/user', function (Request $request) {
17     return $request->user();
18 });
19
20
21 Route::get('/sms', 'SmsController@index');
22
23 Route::post('/sms', 'SmsController@post');
24
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
2: php
[Thu Oct 10 10:33:24 2019] 127.0.0.1:1879 [200]: /favicon.ico
[Thu Oct 10 10:33:24 2019] 127.0.0.1:1880 [200]: /favicon.ico
master 0 0 Listen for XDebug (twitter) Ln 23, Col 40 (4 selected) Spaces: 4 UTF-8 LF PHP 11:12 AM 10/10/2019
```

✓ Ctrl+Shift+F فرمتر نصب میکنه. همیشه این فرمترها رو داشته باش. کدت رو مرتب میکنه.

✓ وقتی میری به phpmyadmin، حتما vscode را از حالت debug در بیاری.

✓ پس در این جلسه:

- Api ای که در وردپرس بود و قبلا به صورت nocode با pods ایجاد میکردیم را با کدنویسی ایجاد کردیم
- با صدا زدن این api، تویبیت مون رو در بانک نوشتیم و خواندیم.
- حالا جلسه بعد میریم سمت آیونیک و کارهایی که با لارا اول انجام دادیم را با آیونیک انجام میدیم:
 - یه صفحه میسازیم: new
 - یه صفحه میسازیم: نمایش همه تویبیت ها

✓ تمرین بیشتر:

- گیت هاب استاد: پروژه های آیونیک، همشون پست دارن