

٩

سبعين  
٣٠ ماي  
أدى المنه

`update()`  $\rightarrow$   $\text{put}$   $\text{Tasks}/\text{id}$   $\rightarrow$   $\text{patch}$   $\text{Tasks}/\text{id}$   $\rightarrow$   $\text{put}$   $\text{Tasks}/\text{id}$

`destroy()`  $\rightarrow$   $\text{delete}$   $\text{Tasks}/\text{id}$

**fork**  $\text{URL}$  controller method

`get tasks index()`  $\rightarrow$   $\text{index}$   $\text{TasksController}$

`get taskstore create()`  $\rightarrow$   $\text{create}$   $\text{TasksController}$

`post tasks store()`  $\rightarrow$   $\text{store}$   $\text{TasksController}$

`get tasks/task show()`  $\rightarrow$   $\text{show}$   $\text{TasksController}$

`get tasks/task/edit edit()`  $\rightarrow$   $\text{edit}$   $\text{TasksController}$

`put patch tasks/task update()`  $\rightarrow$   $\text{update}$   $\text{TasksController}$

`delete tasks/task destroy()`  $\rightarrow$   $\text{destroy}$   $\text{TasksController}$

Route::match(['get', 'post'], '/', function() {})

In post & get view  $\rightarrow$   $\text{TasksController}$   $\rightarrow$   $\text{match}$

$\cdot$  post & get exist in view

Route::any('/', function() {})

2023  $\rightarrow$   $\text{any}$   $\text{verb}$   $\rightarrow$  any

laravel

١٨.٥.٢٠٢٣

دوشنبه

.env  $\rightarrow$  ٢٩ ماي



generate

`php artisan key:generate`  $\rightarrow$   $\text{app-key}$   $\rightarrow$   $\text{env}$   $\rightarrow$   $\text{key}$

$\rightarrow$   $\text{key}$   $\rightarrow$   $\text{secret}$   $\rightarrow$   $\text{private}$   $\rightarrow$   $\text{public}$

`db-database=`  $\rightarrow$   $\text{laravel}$

`db-username=`  $\rightarrow$   $\text{root}$

`db-password=`  $\rightarrow$   $\text{password}$   $\rightarrow$   $\text{laravel}$

: controller  $\rightarrow$   $\text{TasksController}$

`index()`  $\rightarrow$   $\text{index}$   $\text{TasksController}$

`edit()`  $\rightarrow$   $\text{edit}$   $\text{TasksController}$

`create()`  $\rightarrow$   $\text{create}$   $\text{TasksController}$

`store()`  $\rightarrow$   $\text{store}$   $\text{TasksController}$

`destroy()`  $\rightarrow$   $\text{destroy}$   $\text{TasksController}$

`update()`  $\rightarrow$   $\text{update}$   $\text{TasksController}$

`show()`  $\rightarrow$   $\text{show}$   $\text{TasksController}$

`edit()`  $\rightarrow$   $\text{edit}$   $\text{TasksController}$

`create()`  $\rightarrow$   $\text{create}$   $\text{TasksController}$

`store()`  $\rightarrow$   $\text{store}$   $\text{TasksController}$

`destroy()`  $\rightarrow$   $\text{destroy}$   $\text{TasksController}$

`update()`  $\rightarrow$   $\text{update}$   $\text{TasksController}$

`index()`  $\rightarrow$   $\text{index}$   $\text{TasksController}$

١١

پنجشنبه

١ June

الدویت

For auth provider & group  $\leftarrow$  middleware  
 For middleware provider  $\rightarrow$  provider function (group)  
 Route::middleware('auth')  $\rightarrow$  ...  
 Route::middleware('auth')  $\rightarrow$  ...  
 sub domain  $\leftarrow$  sub domain  
 Route::domain('test.myapp.com')  $\rightarrow$  group(function(){  
 Route::get('/', function(){  
 return 'hi';});});  
 Route::domain('account.myapp.com')  $\rightarrow$  group(function(){  
 Route::get('/user/{id}', function(\$id){  
 return \$id;});});  
 For fullback routes on jw tokens E.g. no route: 222  
 Route::fullback(function(){  
 return 'hi';});  
 want to Route by jw tokens  $\leftarrow$  fullback routes

١٤٠٢

2023

١٤٤٤

چهارشنبه

٣ May

الدویت

١٠

Route::get('/{id}', function(\$id){  
 return \$id;});

URL to go to health check (URL endpoint)

healthcheck:8000/12.

12.  $\leftarrow$  id, \$idRoute::get('/{id?}', function(\$id=2){  
 return \$id;});

URL to go to account page (use just ID)

for the account page (use just ID)

for others where no route is specified

Route::get('/{id}', function(\$id){  
 return \$id;});

← ID is not yet defined

{id}  $\rightarrow$  where ('id' {<--> ?});

only for account routes

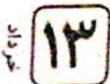
put account route to URL path

sub controller login "login" which has two routes  $\leftarrow$  middlewareRoute::middleware('auth')  $\rightarrow$  group(function(){  
 Route::get('/dashboard', function(){  
 return view('dashboard');});});Route::get('/account', function(){  
 return view('account');});});

٢٠٢٣

٢٠٢٣

٢٠٢٣



سند

3 جلد

مکالمہ

```
Route::get('/', 'HomeController@index');
Route::get('/name/create', 'HomeController@create');
Route::post('/name', 'HomeController@store');
Route::get('/name/show/{id}', 'HomeController@show');
Route::get('/edit/{id}', 'HomeController@edit');
Route::put('/update/{id}', 'HomeController@update');
Route::delete('/name/destroy/{id}', 'HomeController@destroy');
```

Route resource('name', 'HomeController'); ← (پڑھنے کا لئے جو  
spatie/laravel/resource (بے دلیل) name id  
php artisan route:list ← تیرگاہی کریں)

php artisan make: controller
 (controller)
 (singleaction ہے)

Route::get('user/{id}', 'UserController@index'); ← Route Class (2)

Public function \_\_invoke(Request \$request) ← controller class (1)

return "this is a user";

overide (جذب کرنے کے لئے) (2) & (3) کا عمل  
register controller class کو routes: redirect ہے

Route::get('redirect-with-helper', function() { ← (3) (1)

return redirect()->to();});

لیکن جو

1907

2023

1999

for example you can use helper like redirect::to('home')



سند

2 جلد

مکالمہ

view also is passed to the first

with get statement passing view and logic

①
 

```
Route::get('/', function() {
    return view('welcome') → with('var', 'test');
});
```

②
 

```
return view('welcome', ['var' => 'test']);
```

③
 

```
return view('welcome', compact('var'));
```

④
 

```
php artisan make: controller HomeController --resource
    . (controller class)
```

env var = env variables (جذب کرنے کے لئے) (2) کا عمل

App-BEBUG = true
 (true کی وجہ سے serve ہے جذب کرنے کا پاریز)

1907
 2023
 1999

Route :: get('redirect-with-helper-shortcut', function() {  
 return redirect('');  
});

Route :: get('redirect-with-facade', function() {  
 return Redirect::to('');  
});

Route :: redirect('redirect-with-route', '');

Route :: get('redirect', function() {  
 return redirect()->route('numer-route');  
});

return redirect()->route('numer-route', [جواب => 'ستار']);

Route :: get('redirect-with-action', function() {  
 return redirect()->back();  
});

return redirect()->action('');

Route :: get('redirect-with-action', function() {  
 return redirect()->action('@join');  
});

```
php artisan vendor:publish --tag=laravel-errors
```

بررسی این دو مدل برای ایجاد اثر پیش‌بازار و اندیشه‌اندیشه

return [a!, b!, c!];

return response('string', 'status')  
→ headers('content-type', 'text/plain')  
→ cookie('name', 'value', # minutes);

۱۱) آنچه در این حالت می‌گذرد و نیز تواند سعادتمند باشد، را می‌توان در اینجا برای  
آنکه تصور کنیم که تأثیر دارد، بجزئی صورت عکل صاف نماییم.

app → http → middleware → EncryptCookies.php

مُسْتَبْدَلٌ بِالْمُنْسَخَاتِ الْمُؤْكَلَاتِ، مُحْفَظٌ إِلَّا مَنْ يَرِيدُ

return response(1) is 15

1 For  
2023 'age' => 2.,  
1FFF 'status' => 'enable' ]);

new #request ~~pin message~~ / Request ;)

Public Functions store / Request ~~Request~~

```
} dd( #request->all());
```

```
use Illuminate\Http\Request;
```

دستهٔ هرچیز طریق درخواست و پیشنهاد می‌نماید درین request وجود دارد.

return redirect() → away('https://www.google.com');

public function index()

abort (2-2);

دستور (abortion) در اینجا دسترسی دارد فقط اینبار  
دستور (abortion) نهفته و داریم.

abort (e.g., you can not see this page) <= E.T.  $\rightarrow$  it

اصحاف از دستور (abort) و فرمان استفاده می کنم برای حذف چیزی که نباید باشد.

شیوه ارائه ملک و جنگ است برادر ۳ (۱۹۷۴) برو

Route :: post ('something - you - can  
abort (E.g., 'error', 'shelters');

`abort-unless (Auth::user () → is admin), §.4.†;`

if not simpl

لینک سایت مدیریت صنیع admin از اینجا میتوانید دسترسی داشته باشید.

about-if(!Auth::user()->is\_admin(), \$r);

حرداد د س ج آ چ ش ی د س ج ب چ ش ی د س ج **burned** ب چ ش ی د س ج







٢٥  
١٥ June  
ذى القعده ٢٤

view composer is about :2 for what in the service provider

```
public function boot(): void
{
    view() -> composer('view', function ($view) {
        $view->with('count', 5);
    });
}
```

View composer jstioner : 3  
program boot into sys service provider

PHP

test composer.php

namespace App\Http\Composers;

class Testcomposer {

    public function compose(View \$view) {

        \$view->with('want', 5);

    }

}

views>composer(\*'s' function)

views>composer(['test', 'test'], ...)

views>composer(['a', '\*'s' function...])

Component interface

```
    @slot ('color')  
    danger <--> danger  
  @endslot  
  
  @slot ('text')  
  text <--> text  
  @endslot  
Some text  
@end component
```

<button type="button" class="btn btn-ff-color"></button><p>11#stat44</p></div>

sortent  dangor

۱۰. ارسال پیغام درخواست برای ایجاد کاربر (user) با نام این رکورد و از دسترسی محدود نباشد

app / provider / appService provider

دربند boot (جنر bootcamp) درجه بندی از قدر استفاده از میکروسافت سیم و جرام پروتکل سیستم را درین نسخه نشان میدارد.

public function boot():  
    \$count = 5;

عکس صفتی های پستاره است  
حکم از مسدود شدن اینجا می باشد  
وین نه اینجا بلکه در وین اینجا  
وینچه import class import class  
نیزه view support facets view

php artisan make:provider TestServiceProvider  
config/app.php

اپلیکیشن سرویس پرور : application service provider :: class.

APP\providers\Test Service provider :: class,



```

    public function user_id()
    {
        return $this->hasOne(App\user::class); // one to one
    }

    public function invoices()
    {
        return $this->hasMany(App\invoices::class); // one to many
    }

    public function products()
    {
        return $this->belongsToMany(App\products::class); // many to many
    }
}

```

عمر سعید: نعم، این کد برای ایجاد یک رابطه بین کلاس `user` و `invoices` است. در اینجا `user` کلاس اصلی است و `invoices` کلاس مرتبط است. این رابطه یک رابطه یک-به-یک (one-to-one) است.

عمر سعید: این کد برای ایجاد یک رابطه بین کلاس `user` و `products` است. در اینجا `user` کلاس اصلی است و `products` کلاس مرتبط است. این رابطه یک رابطه یک-به-یک (one-to-one) است.

عمر سعید: این کد برای ایجاد یک رابطه بین کلاس `user` و `products` است. در اینجا `user` کلاس اصلی است و `products` کلاس مرتبط است. این رابطه یک رابطه یک-به-یک (one-to-one) است.

id	suppliers	product_id
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
11	11	11
12	12	12
13	13	13
14	14	14
15	15	15
16	16	16
17	17	17
18	18	18
19	19	19
20	20	20
21	21	21
22	22	22
23	23	23
24	24	24
25	25	25
26	26	26
27	27	27
28	28	28
29	29	29
30	30	30
31	31	31
32	32	32
33	33	33
34	34	34
35	35	35
36	36	36
37	37	37
38	38	38
39	39	39
40	40	40
41	41	41
42	42	42
43	43	43
44	44	44
45	45	45
46	46	46
47	47	47
48	48	48
49	49	49
50	50	50
51	51	51
52	52	52
53	53	53
54	54	54
55	55	55
56	56	56
57	57	57
58	58	58
59	59	59
60	60	60
61	61	61
62	62	62
63	63	63
64	64	64
65	65	65
66	66	66
67	67	67
68	68	68
69	69	69
70	70	70
71	71	71
72	72	72
73	73	73
74	74	74
75	75	75
76	76	76
77	77	77
78	78	78
79	79	79
80	80	80
81	81	81
82	82	82
83	83	83
84	84	84
85	85	85
86	86	86
87	87	87
88	88	88
89	89	89
90	90	90
91	91	91
92	92	92
93	93	93
94	94	94
95	95	95
96	96	96
97	97	97
98	98	98
99	99	99
100	100	100
101	101	101
102	102	102
103	103	103
104	104	104
105	105	105
106	106	106
107	107	107
108	108	108
109	109	109
110	110	110
111	111	111
112	112	112
113	113	113
114	114	114
115	115	115
116	116	116
117	117	117
118	118	118
119	119	119
120	120	120
121	121	121
122	122	122
123	123	123
124	124	124
125	125	125
126	126	126
127	127	127
128	128	128
129	129	129
130	130	130
131	131	131
132	132	132
133	133	133
134	134	134
135	135	135
136	136	136
137	137	137
138	138	138
139	139	139
140	140	140
141	141	141
142	142	142
143	143	143
144	144	144
145	145	145
146	146	146
147	147	147
148	148	148
149	149	149
150	150	150
151	151	151
152	152	152
153	153	153
154	154	154
155	155	155
156	156	156
157	157	157
158	158	158
159	159	159
160	160	160
161	161	161
162	162	162
163	163	163
164	164	164
165	165	165
166	166	166
167	167	167
168	168	168
169	169	169
170	170	170
171	171	171
172	172	172
173	173	173
174	174	174
175	175	175
176	176	176
177	177	177
178	178	178
179	179	179
180	180	180
181	181	181
182	182	182
183	183	183
184	184	184
185	185	185
186	186	186
187	187	187
188	188	188
189	189	189
190	190	190
191	191	191
192	192	192
193	193	193
194	194	194
195	195	195
196	196	196
197	197	197
198	198	198
199	199	199
200	200	200
201	201	201
202	202	202
203	203	203
204	204	204
205	205	205
206	206	206
207	207	207
208	208	208
209	209	209
210	210	210
211	211	211
212	212	212
213	213	213
214	214	214
215	215	215
216	216	216
217	217	217
218	218	218
219	219	219
220	220	220
221	221	221
222	222	222
223	223	223
224	224	224
225	225	225
226	226	226
227	227	227
228	228	228
229	229	229
230	230	230
231	231	231
232	232	232
233	233	233
234	234	234
235	235	235
236	236	236
237	237	237
238	238	238
239	239	239
240	240	240
241	241	241
242	242	242
243	243	243
244	244	244
245	245	245
246	246	246
247	247	247
248	248	248
249	249	249
250	250	250
251	251	251
252	252	252
253	253	253
254	254	254
255	255	255
256	256	256
257	257	257
258	258	258
259	259	259
260	260	260
261	261	261
262	262	262
263	263	263
264	264	264
265	265	265
266	266	266
267	267	267
268	268	268
269	269	269
270	270	270
271	271	271
272	272	272
273	273	273
274	274	274
275	275	275
276	276	276
277	277	277
278	278	278
279	279	279
280	280	280
281	281	281
282	282	282
283	283	283
284	284	284
285	285	285
286	286	286
287	287	287
288	288	288
289	289	289
290	290	290
291	291	291
292	292	292
293	293	293
294	294	294
295	295	295
296	296	296
297	297	297
298	298	298
299	299	299
300	300	300
301	301	301
302	302	302
303	303	303
304	304	304
305	305	305
306	306	306
307	307	307
308	308	308
309	309	309
310	310	310
311	311	311
312	312	312
313	313	313
314	314	314
315	315	315
316	316	316
317	317	317
318	318	318
319	319	319
320	320	320
321	321	321
322	322	322
323	323	323
324	324	324
325	325	325
326	326	326
327	327	327
328	328	328
329	329	329
330	330	330
331	331	331
332	332	332
333	333	333
334	334	334
335	335	335
336	336	336
337	337	337
338	338	338
339	339	339
340	340	340
341	341	341
342	342	342
343	343	343
344	344	344
345	345	345
346	346	346
347	347	347
348	348	348
349	349	349
350	350	350
351	351	351
352	352	352
353	353	353
354	354	354
355	355	355
356	356	356
357	357	357
358	358	358
359	359	359
360	360	360
361	361	361
362	362	362
363	363	363
364	364	364
365	365	365
366	366	366
367	367	367
368	368	368
369	369	369
370	370	370
371	371	371
372	372	372
373	373	373
374	374	374
375	375	375
376	376	376
377	377	377
378	378	378
379	379	379
380	380	380
381	381	381
382	382	382
383	383	383
384	384	384
385	385	385
386	386	386
387	387	387
388	388	388



نهضه  
29 June  
اذى الحجه

down, up هجاتح migration

php artisan migrate

php artisan migrate:rollback

dropIfExist('suppliers')

رائع up بجزء تفريغ داده های موجود در این مقاله.

\$table->double('name', 2)

\$table->float('name', 2)

ویژه سطح جزوی است و درینجا مستعار نموده شد

status ← enum('status')

\$table->enum('status', [0, 1]);  
↓  
[enable, disable]

زیرا در این زمانها این داده های سطح جزوی است و درینجا مستعار نموده شد.

string, text ← string, text

\$table->string('name', 255);

\$table->text('name');

زیرا در این زمانها این داده های سطح جزوی است

\$table->timestamps();

نکته: سری داده های زیر را درست نماییم

2023

[laravel.com/docs/6.x/migrations](https://laravel.com/docs/6.x/migrations)

آغاز داده های اولیه در سال 2023

نهضه

20 June

اذى الحجه



Supplier class

public function productHistory()

return \$this->hasManyThrough(App\history::class, App\product::class);

این داده های سطح جزوی است و درینجا مستعار نموده شد. درینجا داده های سطح جزوی است.

این داده های سطح جزوی است و درینجا مستعار نموده شد.

این داده های سطح جزوی است و درینجا مستعار نموده شد.

این داده های سطح جزوی است و درینجا مستعار نموده شد.

public function productHistory()

return \$this->hasManyThrough(App\history::class, App\product::class);

این داده های سطح جزوی است و درینجا مستعار نموده شد.

② stack: برای دسترسی به داده های سطح جزوی دارای دو طبقه است و کل کاملاً با هم مرتبط هستند.

→ @stack('name') / @push('name') ② @stack('name')

→ @endpush

→ @prepend('name') ② @stack('name')

→ @endprepend

→ @in('@stack', '@push')

→ @in('@stack', '@prepend')

→ @in('@stack', '@endpush')

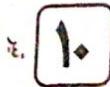
→ @in('@stack', '@endprepend')

برای این داده های سطح جزوی است.

برای این داده های سطح جزوی است.



CamScanner



شنبه

July  
١٢ ذي الحجه

## ٦٥: migrations

دستور: create\_table('table\_name', function (\$table) {  
});

دستور after: من سنن رابطه رسول خبری تواریخ.

دستور: قبل الماقبل دستور آن باقى بقىم دستور که اسماه مکالم.

php artisan make:migration create\_name\_table --table=users

دستور: create\_table('table\_name', function (\$table) {  
});

#table-&gt;text('image')-&gt;after('title')

عنوان title را از سنن

دستور: \$table-&gt;integer('id')-&gt;autoincrement();

#table-&gt;integer('user\_id')-&gt;charset('utf8');

دستور: \$table-&gt;string('title')-&gt;default('');

عنوان default مفهوم سیف و پنهان گوی

#table-&gt;string('title')-&gt;default('');

دستور: \$table-&gt;integer('user\_id')-&gt;unsigned();

لرین دستورها برای دستور از نظر خودها همان دستورات هستند.

#table-&gt;integer('user\_id')-&gt;unsigned();

٦٦: migrations

دستور: create\_table('table\_name', function (\$table) {  
});

لرین دستورها برای دستور از نظر خودها همان دستورات هستند.

#table-&gt;integer('user\_id')-&gt;unsigned();

دور صفت و معدن

جاء

30 June

١١ ذي الحجه



## ٦٧: migrations

php artisan make:migration create\_x\_table --create=x

دستور: create\_table('table\_name', function (\$table) {  
});

status

php artisan m:igration:reset

دستور: migrate:reset

php artisan make:migration  
create\_and\_table --table=x

: phoyadmin, batch

دستور: migrate:batch

٢٠٢٣

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤٤٤

١٤









پیشنهاد

9 July

٢ ذي الحجه

`$users = DB::table('users')->whereBetween('id', [1, 1])->get();`  
• يُرجى إدخال id

`$users = DB::table('users')->whereNotBetween('id', [1, 1])->get();`  
• يُرجى إدخال id

`$users = DB::table('users')->whereLte('id', [1, 1])->get();`  
• يُرجى إدخال id

`$users = DB::table('users')->whereNotLte('id', [1, 1])->get();`  
• يُرجى إدخال id

`$users = DB::table('users')->whereGte('id', [1, 1])->get();`  
• يُرجى إدخال id

`$users = DB::table('users')->whereNotGte('id', [1, 1])->get();`  
• يُرجى إدخال id

`$users = DB::table('users')->whereNull('id')->get();`  
• يُرجى إدخال id

`$users = DB::table('users')->whereNotNull('id')->get();`  
• يُرجى إدخال id

`$users = DB::table('users')->whereDate('update-at', '2023-01-01')->get();`  
• يُرجى إدخال date

`$users = DB::table('users')->whereColumn('1', '2')->get();`  
• يُرجى إدخال id

`$users = DB::table('users')->whereColumn('id', 'name')->get();`  
• يُرجى إدخال id و name

درب حضرت امام موسی کاظم علیه السلام (۱۲۸ھ) - رور ادبیات کودکان و نوجوانان

• where culture is based on the same principles

١٣

8 July

卷二十一



public function index()

```
#users = DB::table('users')->get();
```

`#users = DB::table('users')->select('email')->get();`

where('created\_at', 'now'))  
and now timestamp is fine  
like this  
we can use where now timestamp where  
etc

```
#user=DB::table('users')->where('email','mmond@jhu.edu');  
#           ->where('name','mond')->get();
```

```
$user = DB::table('users')->where([
```

{'email': 'mash@ yahoo.com'},

('name', 'mail') → get()

#USER = DB::table('users')->where('name', 'a')  
                  ->orWhere('name', 'b')->get()

" with name = b & name = a

and where there is no such association between the two variables.

East or west or where <sup>ITP</sup> 2023

`users = DB::table('users')` → `latest()` → `first()`

هان نهاد و میگذرد از میان میوه های خوش رنگ و خوش بو

وَسِرْجَانٌ فِي الْمُدْرَسَةِ وَالْمُدْرَسَةِ الْمُكَانِيَةِ: in Random order

`users = DB.table('users')`  $\rightarrow$  `randomOrder()`  $\rightarrow$  `first()`

دستورات معمولی و معمولیات انتگرال  $\int_a^b f(x) dx$

`users = DB::table('users')->take(5)->get();`

#users = DB::table('users')  
          ->skip(1) //跳過第一筆資料  
          ->take(5) //取前五筆資料  
          ->get(); //執行查詢

`SELECTS = DB::table('posts')->groupBy('user-id')` • criar o lai group by

In most species the ventral and dorsal groupings are always in

'odbc' => database:php => 'mysql' => strict => false  
• Create local user & grant all privs

~~users = DB :: table('posts') → groupBy('user\_id') → having('user\_id', '>', 5)~~

2023 DB := raw('count(\*) as total')

FFFF  $\rightarrow$  معاشر و معاشر

; ~~six~~ where

`Busers=DB::table('users')->where('id',1)->where(function($query)){`

query → where ('name', 'all') → or where ('email', 'x@yahanam'); y) → jets  
(or)  
(email=x@yahanam) (and) id=1 (or

→ select \* from users where id=1 and (name='ali' or email='n@yahahucum').

`$ users = DB::table('posts')->join('users', 'users.id', 'post.user_id')->get();`

جنسیت user-id پر مطابق، posts وجد نمایند. این جملہ users میں شامل نہیں۔ خاص طور پر متن، دیکھا جائے وہی کام وکلہ کوئی نہیں۔

it's just a user interface for a post that's been posted

\* First = DB::table('users')->whereNull('updated\_at'); زیستگاه

#users - DB :: table('users') → where Null ('create\_at') → union (#first) → get();

`users = DB.table('users')  
users.orderBy('date', 'desc').get();`

collection use ordering in 'asc' 1 F.O.Y  
2023

۱۴۴۴ مکالمه اخلاقی شور و دادگستری



٢٣

الجمعة  
14 July  
٢٥ ذي الحجه

مسرّر increment : دستور يزيد قيمة متغير بـ ١ واحد افقياً  
مسرّر decrement : دستور ينقص قيمة متغير بـ ١ واحد افقياً

DB::table('users')->where('id', 1)->increment('age');  
decrement('age', 5);

مسار id=1 في الجدول users محدد بـ 1، لزيادة عمره  
أي مسار id=1 في الجدول users يحدد مسار null باس، وهو خطأ لأن id مسار وابد  
أن يكون معرفاً

مسرّر increment و المسار id=1 في الجدول users دوم حذفه، لعدم انتهاكه بـ 1  
مسرّر decrement لمسار id=1 في الجدول users يدون بـ 1  
و مصدر id=1 في الجدول users يدون بـ 5 لزيادة عمره  
نحو دعوه

مسرّر delete

DB::table('users')->where('id', 1)->delete();

مسرّر transaction

مسرّر transaction يطبق على جميع المجموعات التي تدخل في  
نفس المجموعة، فتسقط كل دعوه في المجموعة  
إذا أخطأ في إحدى الدعوات

DB::transaction(function(){})

DB::table('user')->where('id', 2)->update(['name' => 'John']);

IF ٢٠٢٣  
2023  
١٤٤٤

٢٦

دوشنبه  
17 July  
۲۸ ذی الحجه

## ساختار این پروژه

protected \$primaryKey = 'user\_id';  
 ساختار کلی این پروژه این است که در مجموعه کاربران (User) می‌تواند اینکه این کاربران دارای یک ID مخصوص باشند و این ID را می‌توانند افزایش داد. این ID معمولاً از نوع integer است و ممکن است این ID را می‌توانند افزایش داد.  
 public \$incrementing = false;  
 protected \$keyType = 'string';  
 public \$timestamps = false;  
 last\_update: این کالری می‌تواند مقدار last\_update را ذخیره کند. این کالری معمولاً از نوع timestamp است.  
 const updateAt = 'last\_update';  
 protected \$fillable = [  
 'status' = false;  
 ...  
 ];  
 public function index()  
 \$user = User::all();  
 dd(\$user);  
 . integers collection لیست all می‌باشد!

۱۴۰۴

2023

۱۴۰۴

۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	۱۵	۱۶	۱۷	۱۸	۱۹	۲۰	۲۱	۲۲	۲۳	۲۴	۲۵	۲۶	۲۷	۲۸	۲۹	۳۰	۳۱			
۱۷	۱۸	۱۹	۲۰	۲۱	۲۲	۲۳	۲۴	۲۵	۲۶	۲۷	۲۸	۲۹	۳۰	۳۱	۳۲	۳۳	۳۴	۳۵	۳۶	۳۷	۳۸	۳۹	۴۰	۴۱	۴۲	۴۳	۴۴	۴۵	۴۶	۴۷	۴۸	۴۹	۵۰

سالروز ناسیم نهاد شورای شهرستان

٢٧

یکشنبه  
16 July  
۲۷ ذی الحجه

٢٨

دوشنبه

## ساختار این پروژه

① web.php →  
 Route::get('show-user/{name}',  
 'HomeController@showUser');  
 dd(\$user);

② app/providers/RouteServiceProvider.php  
 public function boot()  
 parent::boot();

Route::bind('name', function(\$value){  
 if (\$value == 'Route')  
 return User::where('name', \$value)->first() ?? abort(404);  
 if (\$value == 'user')  
 return User::where('name', \$value);  
});

show-user / hosseini

Hosseini

model →

myPost: این کالری می‌تواند مقدار myPost را ذخیره کند. این کالری معمولاً از نوع post است و ممکن است این کالری را می‌توانند افزایش داد.  
 myPosts: این کالری می‌تواند مقدار myPosts را ذخیره کند. این کالری معمولاً از نوع post.php است و ممکن است این کالری را می‌توانند افزایش داد.  
 my\_posts: این کالری می‌تواند مقدار my\_posts را ذخیره کند. این کالری معمولاً از نوع post است و ممکن است این کالری را می‌توانند افزایش داد.

protected \$table = 'my\_posts'.

Hosseini

۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	۱۵	۱۶	۱۷	۱۸	۱۹	۲۰	۲۱	۲۲	۲۳	۲۴	۲۵	۲۶	۲۷	۲۸	۲۹	۳۰	۳۱			
۱۷	۱۸	۱۹	۲۰	۲۱	۲۲	۲۳	۲۴	۲۵	۲۶	۲۷	۲۸	۲۹	۳۰	۳۱	۳۲	۳۳	۳۴	۳۵	۳۶	۳۷	۳۸	۳۹	۴۰	۴۱	۴۲	۴۳	۴۴	۴۵	۴۶	۴۷	۴۸	۴۹	۵۰

سالروز ناسیم نهاد شورای شهرستان

CS

CamScanner



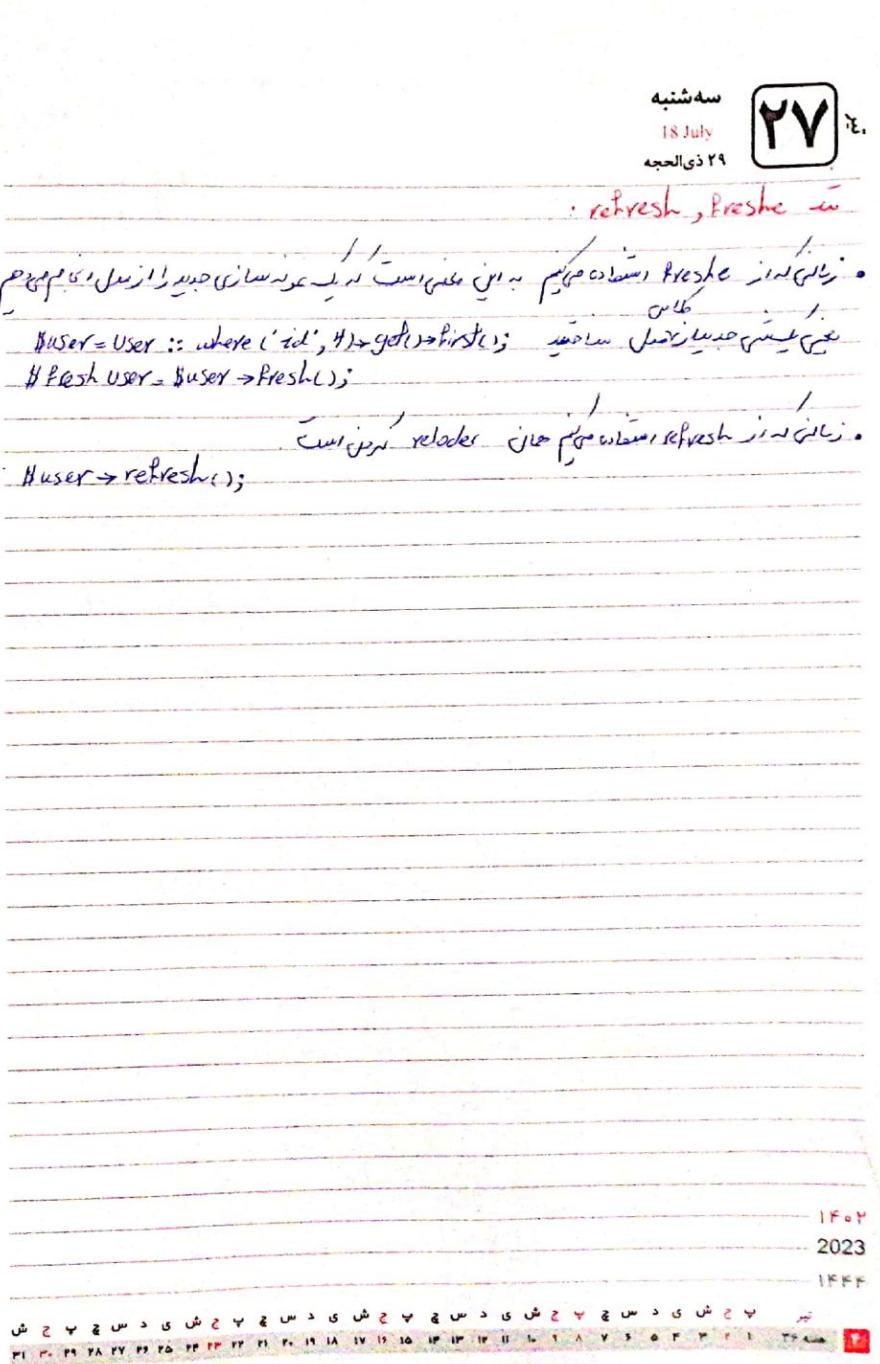
چهارشنبہ

19 July

۱۰۷

1FOP  
2023  
1FFAQ

نامه سال ۱۴۰۵ هجری قمری - روز شنبه و ادبیات آسیا - روزگارگاه حشم کاشانی





جامعة

21 July

part

لقد قررت مراجعة الدرس الذي درسناه من قبل لذا سأكتب

```
json
'key1': 'value1'
'key2': 'value2'
'key3':
  'nestedKey1': 'nestValue1'
  'nestedKey2': 'nestValue2';
  'key4': [
    'element1', 'element2', 'elements'
  ]
  'key5': {
    'id': 1,
    'name': 'John Doe',
    'age': 32,
    'isStudent': false,
    'courses': ['a', 'b', 'c'],
    'address': {
      'city': 'New York',
      'zipCode': '100-100'
    }
  }
}
```

```
json
'menu': 'main',
'age': 32,
'isStudent': false,
'courses': ['a', 'b', 'c'],
'address': {
  'city': 'New York',
  'zipCode': '100-100'
}
```

لذلك سأكتب

```
var obj = {
  menu: 'main',
  age: 32,
  isStudent: false,
  courses: ['a', 'b', 'c'],
  address: {
    city: 'New York',
    zipCode: '100-100'
  }
};
```

لذلك سأكتب

```
public function up() {
  // table->json('users');
  // json
  class yourModel extends Model {
    protected $fillable = ['users'];
  }
}
```

: migration (لما ذكرت سابقاً)

: Class

لذلك سأكتب

14:47

2023

iffa

لذلك سأكتب

پنجشہر

20 July

part



: php constant

لقد قررت مراجعة الدرس الذي درسناه من قبل لذا سأكتب

لذلك سأكتب

لذلك سأكتب

لذلك سأكتب

لذلك سأكتب

```
class MyClass {
  const x = 'hi';
}

echo MyClass::x;
```

: global variable

```
namespace myNameSpace;
const x = 'hi';

echo myNameSpace\z;
```

: global variable

: controller, model, const statement.

```
class myController extends Controller {
  public function index() {
    $model = new Model();
    dd($model);
    return view('index');
  }
}
```

: controller

: model

: const statement

١

پیکنیک  
23 July  
دسمبر

دستورات  
لایه های

۲ (لایه) در این لایه هر چیزی که برای این سطح مورد نیاز باشد در این لایه قرار گیرد مثلاً address، user

User class (name, coordinates) فرمی که این دو address و coordinates را در یکجا address و coordinates

public function up() {  
 Schema::create('address', function(Blueprint \$table){

\$table->increments('id');

\$table->string('address');

\$table->string('city');

\$table->string('zip\_code');

\$table->bigIntegerUnsigned('user\_id')->index();

\$table->timestamps();

\$table->foreign('user\_id')->references('id')->on('users');

->onDelete('cascade')->onUpdate('cascade');});

IFP

2023

IFPO

دستورات لایه دوم

٣١

شنبه  
22 July  
دسمبر

Route model

/user model id از خود است  
/user/  
جستجوی user با home

Route::get('/edit/{id}', [HomeController::class, 'edit']);  
Route::get('/edit/{User}', [HomeController::class, 'edit']);  
public function edit(\$user) {  
 return view('home', compact('user'));  
}  
public function edit (\$id) {  
 \$user = User::where('id', '\$id');  
 return view('home', compact('user'));

home controller

IFP

2023

IFPO

دستورات لایه سوم



سہ شنبہ  
25 July  
۷ محرم

public function comments() {  
 return #this->has many (comment::class);  
 }  
 public function product() {  
 return #this->belongs to (product ::class);  
 }  
 \$product = product::has('comments')->get;  
 \$comment = \$product->associate(\$comment);

comment class  
 has many comments  
 belongs to product  
 \$comment = \$product->associate(\$comment);

comment class  
 has many comments  
 belongs to product  
 \$comment = \$product->associate(\$comment);

comment class  
 has many comments  
 belongs to product  
 \$comment = \$product->associate(\$comment);

comment class  
 has many comments  
 belongs to product  
 \$comment = \$product->associate(\$comment);

• can associate & can dissociate



دوشنبه  
24 July  
۶ محرم

public function address() {
 return #this->has one (address::class);
 }

belongs to #this User class

public function user() {
 return #this->belongs to (user::class);
 }

belongs to #this User class

برهانیت و مکالمہ و نظریت

public function upc() {  
 pris migration comment as 5  
 #table → big increments('id');  
 \$table → text ('comment');  
 \$table → biginteger unsigned(); index;  
 #table → Foreign('product\_id') → references('id') → on('product\_id')  
 → ondelete('cascade') → onupdate('cascade');  
 →



پنجشنبه

27 July

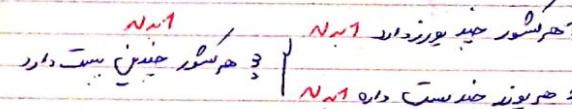
احمد

table 'car' doesn't exist.

```
class car extends model {
    protected $table = 'cars';
```

countries → user → post

N = 1



public function up()

← country (users)

schema::create('countries', function(Blueprint \$table) {

\$table→bigIncrements('id');

\$table→string('name');

});

public function up()

← user (posts)

schema::create('users', function(Blueprint \$table) {

\$table→bigIncrements('id');

\$table→unsignedBigInteger('country\_id');

\$table→foreign('country\_id')→references('id')→on('countries')→...;

public function up()

← post (users)

schema::create('posts', function(Blueprint \$table) {

\$table→bigIncrements('id');

});

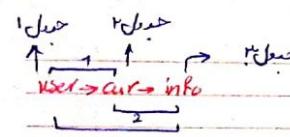
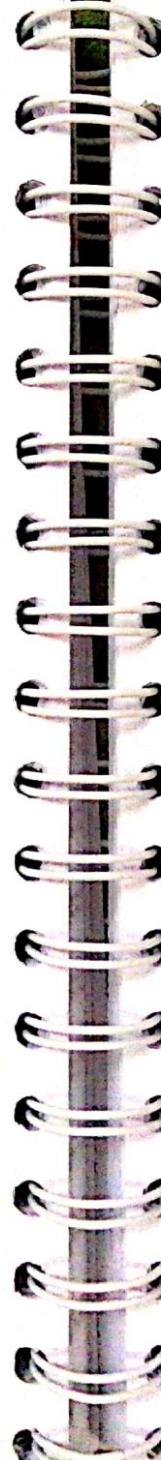
\$table→foreign('user\_id')→references('id')→on('users')→...;

نحوی حسین (عطف) - سالرو علیکم افضل آفرین مرحوم (۱۳۷۴)

نحوی حسین (عطف) - سالرو علیکم افضل آفرین مرحوم (۱۳۷۴)

نحوی حسین (عطف) - سالرو علیکم افضل آفرین مرحوم (۱۳۷۴)

نحوی حسین (عطف) - سالرو علیکم افضل آفرین مرحوم (۱۳۷۴)



جواب  
جواب  
جواب  
جواب

public function up()

← car (info)

← user (info)

← car (user)

← user (car)



چهارشنبه

26 July

احمد

1 ~ 7 ~ 1 ~ 5  
1 ~ 7 ~ 1 ~ 5  
1 ~ 7 ~ 1 ~ 5

جواب  
جواب  
جواب

جواب  
جواب  
جواب

public function up()

← car (migration)

← schema::create('cars', function(Blueprint \$table) {

\$table→bigIncrements('id');

\$table→String('name');

});

\$table→unsignedBigIncrements('user\_id');

\$table→foreign('user\_id')→references('id')→on('user')→onDelete('cascade');

→ onUpdate('cascade');

public function up()

← (information) (User) (migration)

schema::create('information', function(Blueprint \$table) {

\$table→bigIncrements('id');

\$table→String('color');

});

\$table→unsignedBigIncrements('car\_id');

\$table→foreign('car\_id')→references('id')→on('cars')→...

→ ...

public function up()

← (User) (information) (car) (migration)

public function up()

← (User) (information) (car) (migration)

return \$this→hasOneThrough('information::class', 'car::class');

});

#user = user::find(1);

\$user→info

});



session : ان اعمال راجحة تتحقق في دروسه ما يليه من اعمال  
من اعماله مهارات ارجح اعماله مهارات مهارات مهارات مهارات  
يذكرها مهارات دروسه دروسه دروسه دروسه

در سیاست ایرانی مادر.

```
Session::put('key', 'value');  
Session::push('user', 'users', 'developers');
```

Prosternon pusillum pet. auf Wurzel geschnitten

اللهم اذْهَبْ لِي مَا  
فِي سَاعَةِ الْجُنُوبِ وَامْكِنْ لِي مَا  
فِي سَاعَةِ الْجُنُوبِ

value = Session::get('key');

Postponed Budget and Sessional Paper

Session #1 forget('key');

comes into play in session in number 5 message.

## Session :: flush();

لیکن این پس از اینکه میگذرد و در نهایت میگذرد.

it (session :: bas('key'))

2023 Februar 11 Freitag

卷之三

卷之三

11

چهارشنبه

2 August

ساعت ۱۲

## • query builder •

از پیش از اینکه دستورات SQL را بخواهیم بنویسیم، میتوانیم از دستورات increment و decrement استفاده کنیم.

DB::table('users')->where('id', 1)->increment('decrement', 'age', 5)

این دستورات برای اینکه مقداری از مقدار فعلی یک ستون را افزایش یا کاهش داده و آن را ذخیره کنند.

استفاده از دستور truncate : truncate

DB::table('users')->truncate();

این دستورات برای حذف تمامی محتویات یک مدل هستند. **transaction**

مزایت بعد از اجرای دستور حالت خطا در میان فرآیند را ندارد.

DB::transaction(function() {

    // کد

}); این دستورات محدودیت میزداید.

php artisan make: model post

: model name

php artisan make: model --help

: route model

in local host : e. /showUser/2

① route = get ('/showUser/{id}', 'homeController@showUser');

② route = get ('/show-user/{user}', 'homeController@showUser');

③ public function showUser (\$id) {

    ② public function showUser

        \$user = User::find(\$id);

        User

        \$user = User::find(\$id);

        User

        \$user = User::find(\$id);

        User

        User









Query Scope

شیوه ایجاد شده در این مسیر را می توان در مسیرهای دیگری نیز ایجاد کرد.

*act = -1/T ln d; -P<0; D<0;*

لَكَمْبِنْتُ بَعْدَ مُهَاجَرَةِ الْأَنْجَوْنِيَّ

142

卷之三

yesterdays

```

    graph TD
      A[public function query(states, status)] --> B[scrap(states)]
      B --> C["return query('states', 1);"]
  
```

```
return query->where('status', 1);  
} }  
public function index() {  
    $users = User::where('status', 1)  
        ->get();  
    return view('users.index', [
```

scope  
scope status scope.php

17

namespace App\Scope;  
use Illuminate\Database\Eloquent\Scope;  
class staticScope implements Scope  
{  
 public function applyBuilder(Builder \$builder)  
 {  
 \$builder->where('is\_deleted', 0);  
 }  
}

```
blicker->where('status', 1); {
```

مکالمہ حضرت امام حنفی علیہ السلام کے درجات میں ایک ممتاز مدرس تھا۔

برای حذف یک سند از دستور soft delete استفاده کنید.

لَكَمْبِنْتُ بَعْدَهُمْ وَيَقْرَأُونَ مِنْ كُلِّ  
كِتَابٍ وَلَا يَعْلَمُونَ

142

مکر راست پیشوند  
لے کر اپنا بھروسہ نہ کریں گے اس کے ساتھ مل کر جو مدد ہے پس پڑیں گے

*User - user* file name number class user extends *method*

protected #guarded [redacted]

2 protected \$ Efficacy = [ number incorrect ]

front and the strong egg-sigma.

مکتبہ نظریہ دین اسلام

public function store(Request \$request) {  
 \$category = new Category;  
 \$category->name = \$request->name;  
 \$category->save();  
 return redirect('/categories');  
}

卷之三







18

## • Signantes

polysyndetic : je veux bien

close to one's goals  
and a strong  
positive image

passenger & goods (b) <

large shrubs → pierce, damage, plant, wear (from)

note factors w/ p. - positive & negative

Show a variety images showing blueprint stability

*Streptomyces* sp.

stable - strong (weak);

stable →  $\text{AgNO}_3$  (magnesium salt) →  $\text{Mg}^{2+}$  +  $\text{NO}_3^-$

$\text{Shoe} \rightarrow \text{frag. (angustifrons)} \rightarrow \text{subtropical}$

\$table->timestamps();});

			possible values ↪
id	are	longable-id	longable- <u>id</u> factory
1	test	1	AppPost null
2	test2	1	AppUser null

class Image extends mold {  
 public function Imageable()

return #this->empty(); } //check for null image class

更多資訊請上網查詢：[www.taiwantravel.com.tw](http://www.taiwantravel.com.tw)

۱۷ August

17 AUGUST

۲۶

17

~~form~~ action="if radio('post', destroy); & post->add('method', 'post');  
@script  
@method('Delete');  
button type="submit" class="...">> input[type="button"]> [form]  
  
public function destroy(post \$post)  
\$post->delete();  
return redirect('post\_index')->with('s', 'record deleted successfully');

لهم اذكّر انك رب العالمين اسألك عافيتك

۲۹

پیشنهاد  
20 August  
۳ صفر

one to many relationship  
posts comments  
video comments belongs to post (one to many relationship)

هر یک پست دارای بیش از یک نظر می باشد

هر یک نظر مربوط به یک پست است

post reviews belongs to post (one to many relationship)

public function up() {

schema::create ('comments', function (Blueprint \$table) {

\$table->bigIncrements ('id');

\$table->text ('body');

\$table->integer ('commentable\_id');

\$table->string ('commentable\_type');

\$table->timestamps();

id	body	commentable_id	commentable_type
1	null	1	App\post
2	null	1	App\video

public function commentable() {

return \$this->morphTo();

I F O P

2023

I F F O

مددگار آنلاین برای مادرانه شدن (۱۲۲۲ دش)

۳۰

شنبه  
19 August  
۴ صفر

ویرایش post در پست

class Post extends Model {

public function image() {

return \$this->morphOne ('App\image', 'imageable');

one to many relationship

ویرایش هر یک پست برای نمایش user

imageable-type imageable-id

#post = Post::find (1);

dd (\$post->image);

← post (۱)

#user = User::find (1);

← user (۱)

dd (\$user->image);

← user (۱)

id

#image = Image::find (1);

dd (\$image->imageable);

.user & post

مددگار آنلاین برای مادرانه شدن (۱۲۲۲ دش)



سے شنبہ  
22 August  
۵ صفر

many to many		
tag-id	taggable-id	taggable-type
1	20	App\blog
1	21	"
2	20	"
1	1	App\lore

```
public function tags():  
    : blog  
  
    return $this->morphToMany('App\Tag', 'taggable');  
}  
2023  
  
IFPS  
    : podes podre blog  
      
    
```



دوشنبه  
21 August  
۴ صفر

```
public function comments() {  
    postUrl = 'http://';  
    return $this->morphMany('App\Comment', 'commentable');  
}  
//first class yes video (url) , 5
```

```
#post = post :: find(1); post or's
do(&post); del(&post->comments);  

#video = video :: find(1); video or's
do(&video); del(&video->comments);  

$post = post :: find(1);  

$post->comment = new Comment;  

$post->comment->body = 'bah bah';  

$post->comment->save();  

$post->comment->id
```



پنجشنبہ  
24 August  
۷ صفر

صفر

[View Details](#)

10

در نسخه سیستمی است برای test و مجدد داد

Compound : unit test / integration test

php artisan make:test firstTest --unit / feature

class firstTest extends TestCase {  
 public function only logged in users can visit this page() {  
 \$this->assertTrue(true);  
 }  
}

۲۰۱۷-۰۶-۰۸ ۱۰:۳۴:۵۹  
پیغام از سمت دستور خط  
برای دستور `phpunit` در پوشه `tests` از فرمان زیر استفاده شود:

```
./vendor/bin/phpunit --filter "App\\Http\\Controllers\\Auth\\LoginControllerTest"
```

public function blogs()

```
return $this->morphByMany('APP\\blob', 'taggable');
```

public function does() {

```
return #this→morphBy.morph('App1.dore','taggable');{
```

```
# blog = blog :: find(2.);  
# dd( blog->tags);
```

```
#dore = dore :: find(1);  
#dd(dore->tags);
```

`#tag = tag :: finds(1);  
#dd (#tag -> dores );  
#del (#tag -> blogs);`



۷

دوشنبه

23 August

11

```
@if ($errors->any())
    <div class='alert alert-danger'>
        @foreach ($errors->all() as $error)
            <li> {{ $error }} </li>
        @endforeach
    </div>
@elseif
    @include('layouts.partials.error')

```

150

2023

1FFA

A horizontal scroll with Persian text, likely a portion of the Quran, written in a traditional calligraphic style.

A vertical photograph of a spiral-bound notebook. The notebook has a dark grey or black textured cover. A metal spiral binding runs vertically down the center. The spiral rings are visible along the edge. The background is plain white.

شیرور ۸ ۲۷ شی دس ۲۷ بچ شی دس ۲۷ بچ شی دس ۸ ۲۷ همه ۳۲۱



چهارشنبہ  
30 August  
۱۴ صفر

~~Request-Call();~~

١٠٣٦ میں اپنے بھائی کے ساتھ اپنے بھائی کے ساتھ اپنے بھائی کے ساتھ

`request->input('input form');` request class has input job jobs in order to  
get the value of input field. This code will indicate what input field  
`request->input('input form', 'input id');` passing param  
`request->input form;` `request->title;` .  
`request->only('the form');` . we can use this function to get the value of  
input form . we can use this function to get the value of input form .  
only

`request->except('nojs');` only works in older browsers  
`request->except('token');` works in older browsers  
`request->has('nojs');` works in newer browsers  
`request->filled('nojs');` works in newer browsers

1 <input type="text" name="employees[0][firstName]>  
2 <input type="text" name="employees[0][lastName]>  
3 <input type="text" name="employees[1][firstName]>  
4 <input type="text" name="employees[1][lastName]>

فرم بحثت وبری به دست ملکان و شهادت ملکوان شهداں رجایی و سهرم ۱۳۹۷ میں  
Request → input('employees.10.firstname') → ۳۲ بار یکسان



سہ شنبہ  
29 August  
۱۲ صفر

## : HTTP Request

• Request is also used to ask for something.

```
use Illuminate\Http\Request;
```

```
public function inter(Request $request) {
```

```
dd( #request->path());
```

\$request → path

ستل: میتوان سرطان را در سرمهای بود ملان طراحی نماید

#request → isp . ذا (آدمی) میں سے کوئی بھی واحد اور مخصوص نہیں۔

**مثال:** میتوان شرط نهادت اول رسیر ملان بور (درست هدر) هر مرعوط آن میتواند

`prequest.url();` . موسیقی‌خواهی درست

`$request->fullUrl(); // usage: $querystring = $request->fullUrl();`

#request to method(); (postponed) requests, inizi

`if request.method('post/get');`

method in j2ee is ~~single~~





دوشنبه  
4 September  
۱۶ صفر

۱۱۴

ساعت ۱۹  
۵ September

tests → unit → first test.php

public function test\_input\_missing\_a\_title\_is\_rejected() {

\$response = \$this->post('/rate', 'post.store'));

→ signed in HTTP browser

'title' => 'test title';

\$response -> assertRedirect();  
"assertSessionErrors();"

app\http\request

• Civil rules , authorize exception

(true view). now validations given in middleware authorized (rules)

public function authorize() {

return true;

public function rules() {

return ['  
title' => 'required|min:5';

store

↑ public function storePostRequest(Request \$request) {  
(use App\http\Request\StorePost; middleware, yes private)

5  
2

\$validated = \$request->validated();

1140  
1141  
2023  
1142  
1143  
1144  
1145  
1146  
1147  
1148  
1149  
1150  
1151  
1152  
1153  
1154  
1155  
1156  
1157  
1158  
1159  
1160  
1161  
1162  
1163  
1164  
1165  
1166  
1167  
1168  
1169  
1170  
1171  
1172  
1173  
1174  
1175  
1176  
1177  
1178  
1179  
1180  
1181  
1182  
1183  
1184  
1185  
1186  
1187  
1188  
1189  
1190  
1191  
1192  
1193  
1194  
1195  
1196  
1197  
1198  
1199  
1200  
1201  
1202  
1203  
1204  
1205  
1206  
1207  
1208  
1209  
1210  
1211  
1212  
1213  
1214  
1215  
1216  
1217  
1218  
1219  
1220  
1221  
1222  
1223  
1224  
1225  
1226  
1227  
1228  
1229  
1230  
1231  
1232  
1233  
1234  
1235  
1236  
1237  
1238  
1239  
1240  
1241  
1242  
1243  
1244  
1245  
1246  
1247  
1248  
1249  
1250  
1251  
1252  
1253  
1254  
1255  
1256  
1257  
1258  
1259  
1260  
1261  
1262  
1263  
1264  
1265  
1266  
1267  
1268  
1269  
1270  
1271  
1272  
1273  
1274  
1275  
1276  
1277  
1278  
1279  
1280  
1281  
1282  
1283  
1284  
1285  
1286  
1287  
1288  
1289  
1290  
1291  
1292  
1293  
1294  
1295  
1296  
1297  
1298  
1299  
1300  
1301  
1302  
1303  
1304  
1305  
1306  
1307  
1308  
1309  
1310  
1311  
1312  
1313  
1314  
1315  
1316  
1317  
1318  
1319  
1320  
1321  
1322  
1323  
1324  
1325  
1326  
1327  
1328  
1329  
1330  
1331  
1332  
1333  
1334  
1335  
1336  
1337  
1338  
1339  
1340  
1341  
1342  
1343  
1344  
1345  
1346  
1347  
1348  
1349  
1350  
1351  
1352  
1353  
1354  
1355  
1356  
1357  
1358  
1359  
1360  
1361  
1362  
1363  
1364  
1365  
1366  
1367  
1368  
1369  
1370  
1371  
1372  
1373  
1374  
1375  
1376  
1377  
1378  
1379  
1380  
1381  
1382  
1383  
1384  
1385  
1386  
1387  
1388  
1389  
1390  
1391  
1392  
1393  
1394  
1395  
1396  
1397  
1398  
1399  
1400  
1401  
1402  
1403  
1404  
1405  
1406  
1407  
1408  
1409  
1410  
1411  
1412  
1413  
1414  
1415  
1416  
1417  
1418  
1419  
1420  
1421  
1422  
1423  
1424  
1425  
1426  
1427  
1428  
1429  
1430  
1431  
1432  
1433  
1434  
1435  
1436  
1437  
1438  
1439  
1440  
1441  
1442  
1443  
1444  
1445  
1446  
1447  
1448  
1449  
1450  
1451  
1452  
1453  
1454  
1455  
1456  
1457  
1458  
1459  
1460  
1461  
1462  
1463  
1464  
1465  
1466  
1467  
1468  
1469  
1470  
1471  
1472  
1473  
1474  
1475  
1476  
1477  
1478  
1479  
1480  
1481  
1482  
1483  
1484  
1485  
1486  
1487  
1488  
1489  
1490  
1491  
1492  
1493  
1494  
1495  
1496  
1497  
1498  
1499  
1500  
1501  
1502  
1503  
1504  
1505  
1506  
1507  
1508  
1509  
1510  
1511  
1512  
1513  
1514  
1515  
1516  
1517  
1518  
1519  
1520  
1521  
1522  
1523  
1524  
1525  
1526  
1527  
1528  
1529  
1530  
1531  
1532  
1533  
1534  
1535  
1536  
1537  
1538  
1539  
1540  
1541  
1542  
1543  
1544  
1545  
1546  
1547  
1548  
1549  
1550  
1551  
1552  
1553  
1554  
1555  
1556  
1557  
1558  
1559  
1560  
1561  
1562  
1563  
1564  
1565  
1566  
1567  
1568  
1569  
1570  
1571  
1572  
1573  
1574  
1575  
1576  
1577  
1578  
1579  
1580  
1581  
1582  
1583  
1584  
1585  
1586  
1587  
1588  
1589  
1590  
1591  
1592  
1593  
1594  
1595  
1596  
1597  
1598  
1599  
1600  
1601  
1602  
1603  
1604  
1605  
1606  
1607  
1608  
1609  
1610  
1611  
1612  
1613  
1614  
1615  
1616  
1617  
1618  
1619  
1620  
1621  
1622  
1623  
1624  
1625  
1626  
1627  
1628  
1629  
1630  
1631  
1632  
1633  
1634  
1635  
1636  
1637  
1638  
1639  
1640  
1641  
1642  
1643  
1644  
1645  
1646  
1647  
1648  
1649  
1650  
1651  
1652  
1653  
1654  
1655  
1656  
1657  
1658  
1659  
1660  
1661  
1662  
1663  
1664  
1665  
1666  
1667  
1668  
1669  
1670  
1671  
1672  
1673  
1674  
1675  
1676  
1677  
1678  
1679  
1680  
1681  
1682  
1683  
1684  
1685  
1686  
1687  
1688  
1689  
1690  
1691  
1692  
1693  
1694  
1695  
1696  
1697  
1698  
1699  
1700  
1701  
1702  
1703  
1704  
1705  
1706  
1707  
1708  
1709  
17010  
17011  
17012  
17013  
17014  
17015  
17016  
17017  
17018  
17019  
17020  
17021  
17022  
17023  
17024  
17025  
17026  
17027  
17028  
17029  
17030  
17031  
17032  
17033  
17034  
17035  
17036  
17037  
17038  
17039  
17040  
17041  
17042  
17043  
17044  
17045  
17046  
17047  
17048  
17049  
17050  
17051  
17052  
17053  
17054  
17055  
17056  
17057  
17058  
17059  
17060  
17061  
17062  
17063  
17064  
17065  
17066  
17067  
17068  
17069  
17070  
17071  
17072  
17073  
17074  
17075  
17076  
17077  
17078  
17079  
17080  
17081  
17082  
17083  
17084  
17085  
17086  
17087  
17088  
17089  
17090  
17091  
17092  
17093  
17094  
17095  
17096  
17097  
17098  
17099  
170100  
170101  
170102  
170103  
170104  
170105  
170106  
170107  
170108  
170109  
170110  
170111  
170112  
170113  
170114  
170115  
170116  
170117  
170118  
170119  
170120  
170121  
170122  
170123  
170124  
170125  
170126  
170127  
170128  
170129  
170130  
170131  
170132  
170133  
170134  
170135  
170136  
170137  
170138  
170139  
170140  
170141  
170142  
170143  
170144  
170145  
170146  
170147  
170148  
170149  
170150  
170151  
170152  
170153  
170154  
170155  
170156  
170157  
170158  
170159  
170160  
170161  
170162  
170163  
170164  
170165  
170166  
170167  
170168  
170169  
170170  
170171  
170172  
170173  
170174  
170175  
170176  
170177  
170178  
170179  
170180  
170181  
170182  
170183  
170184  
170185  
170186  
170187  
170188  
170189  
170190  
170191  
170192  
170193  
170194  
170195  
170196  
170197  
170198  
170199  
170200  
170201  
170202  
170203  
170204  
170205  
170206  
170207  
170208  
170209  
170210  
170211  
170212  
170213  
170214  
170215  
170216  
170217  
170218  
170219  
170220  
170221  
170222  
170223  
170224  
170225  
170226  
170227  
170228  
170229  
170230  
170231  
170232  
170233  
170234  
170235  
170236  
170237  
170238  
170239  
170240  
170241  
170242  
170243  
170244  
170245  
170246  
170247  
170248  
170249  
170250  
170251  
170252  
170253  
170254  
170255  
170256  
170257  
170258  
170259  
170260  
170261  
170262  
170263  
170264  
170265  
170266  
170267  
170268  
170269  
170270  
170271  
170272  
170273  
170274  
170275  
170276  
170277  
170278  
170279  
170280  
170281  
170282  
170283  
170284  
170285  
170286  
170287  
170288  
170289  
170290  
170291  
170292  
170293  
170294  
170295  
170296  
170297  
170298  
170299  
170300  
170301  
170302  
170303  
170304  
170305  
170306  
170307  
170308  
170309  
170310  
170311  
170312  
170313  
170314  
170315  
170316  
170317  
170318  
170319  
170320  
170321  
170322  
170323  
170324  
170325  
170326  
170327  
170328  
170329  
170330  
170331  
170332  
170333  
170334  
170335  
170336  
170337  
170338  
170339  
170340  
170341  
170342  
170343  
170344  
170345  
170346  
170347  
170348  
170349  
170350  
170351  
170352  
170353  
170354  
170355  
170356  
170357  
170358  
170359  
170360  
170361  
170362  
170363  
170364  
170365  
170366  
170367  
170368  
170369  
170370  
170371  
170372  
170373  
170374  
170375  
170376  
170377  
170378  
170379  
170380  
170381  
170382  
170383  
170384  
170385  
170386  
170387  
170388  
170389  
170390  
170391  
170392  
170393  
170394  
170395  
170396  
170397  
170398  
170399  
170400  
170401  
170402  
170403  
170404  
170405  
170406  
170407  
170408  
170409  
170410  
170411  
170412  
170413  
170414  
170415  
170416  
170417  
170418  
170419  
170420  
170421  
170422  
170423  
170424  
170425  
170426  
170427  
170428  
170429  
170430  
170431  
170432  
170433  
170434  
170435  
170436  
170437  
170438  
170439  
170440  
170441  
170442  
170443  
170444  
170445  
170446  
170447  
170448  
170449  
170450  
170451  
170452  
170453  
170454  
170455  
170456  
170457  
170458  
170459  
170460  
170461  
170462  
170463  
170464  
170465  
170466  
170467  
170468  
170469  
170470  
170471  
170472  
170473  
170474  
170475  
170476  
170477  
170478  
170479  
170480  
170481  
170482  
170483  
170484  
170485  
170486  
170487  
170488  
170489  
170490  
170491  
170492  
170493  
170494  
170495  
170496  
170497  
170498  
170499  
170500  
170501  
170502  
170503  
170504  
170505  
170506  
170507  
170508  
170509  
170510  
170511  
170512  
170513  
170514  
170515  
170516  
170517  
170518  
170519  
170520  
170521  
170522  
170523  
170524  
170525  
170526  
170527  
170528  
170529  
170530  
170531  
170532  
170533  
170534  
170535  
170536  
170537  
170538  
170539  
170540  
170541  
170542  
170543  
170544  
170545  
170546  
170547  
170548  
170549  
170550  
170551  
170552  
170553  
170554  
170555  
170556  
170557  
170558  
170559  
170560  
170561  
170562  
170563  
170564  
170565  
170566  
170567  
170568  
170569  
170570  
170571  
170572  
170573  
170574  
170575  
170576  
170577  
170578  
170579  
170580  
170581  
170582  
170583  
170584  
170585  
170586  
170587  
170588  
170589  
170590  
170591  
170592  
170593  
170594  
170595  
170596  
170597  
170598  
170599  
170600  
170601  
170602  
170603  
170604  
170605  
170606  
170607  
170608  
170609  
170610  
170611  
170612  
170613  
170614  
170615  
170616  
170617  
170618  
170619  
170620  
170621  
170622  
170623  
170624  
170625  
170626  
170627  
170628  
170629  
170630  
170631  
170632  
170633  
170634  
170635  
170636  
170637  
170638  
170639  
170640  
170641  
170642  
170643  
170644  
170645  
170646  
170647  
170648  
170649  
170650  
170651  
170652  
170653  
170654  
170655  
170656  
170657  
170658  
170659  
170660  
170661  
170662  
170663  
170664  
170665  
170666  
170667  
170668  
170669  
170670  
170671  
170672  
170673  
170674  
170675  
170676  
170677  
170678  
170679  
170680  
170681  
170682  
170683  
170684  
170685  
170686  
170687  
170688  
170689  
170690  
170691  
170692  
170693  
170694  
170695  
170696  
170697  
170698  
170699  
170700  
170701  
170702  
170703  
170704  
170705  
170706  
170707  
170708  
170709  
170710  
170711  
170712  
170713  
170714  
170715  
170716  
170717  
170718  
170719  
170720  
170721  
170722  
170723  
170724  
170725  
170726  
170727  
170728  
170729  
170730  
170731  
170732  
170733  
170734  
170735  
170736  
170737  
170738  
170739  
170740  
170741  
170742  
170743  
170744  
170745  
170746  
170747  
170748  
170749  
170750  
170751  
170752  
170753  
170754  
170755  
170756  
170757  
170758  
170759  
170760  
170761  
170762  
170763  
170764  
170765  
170766  
170767  
170768  
170769  
170770  
170771  
170772  
170773  
170774  
170775  
170776  
170777  
170778  
170779  
170780  
170781  
170782  
170783  
170784  
170785  
170786  
170787  
170788  
170789  
170790  
170791  
170792  
170793  
170794  
170795  
170796  
170797  
170798  
170799  
170800  
170801  
170802  
170803  
170804  
170805  
170806  
170807  
170808  
170809  
170810  
170811  
170812  
170813  
170814  
170815  
170816  
170817  
170818  
170819  
170820  
170821  
170822  
170823  
170824  
170825  
170826  
170827  
170828  
170829  
170830  
170831  
170832  
170833  
170834  
170835  
170836  
170837  
170838  
170839  
170840  
170841  
170842  
170843  
170844  
170845  
170846  
170847  
170848  
170849  
170850  
170851  
170852  
170853  
170854  
170855  
170856  
170857  
170858  
170859  
170860  
170861  
170862  
170863  
170864  
170865  
170866  
170867  
170868  
170869  
170870  
170871  
170872  
170873  
170874  
170875  
170876  
170877  
170878  
170879  
170880  
170881<br





شنبه  
۹ September  
۲۲ صفر

نحوی و محتوا در این فصل  
protected \$middlewareGroups = [  
 'web' => [ \App\Http\middleware\checkAge::class ],  
 'web.php' => [ \App\Http\middleware\checkAge::class ]  
 : to run middleware just one  
 Route::get('/', 'HomeController@index')->middleware('web');  
 نحوی و محتوا : to middleware just one  
 Route middleware ← → فرآیند میانبری که در چند گزینه  
 'age' => \App\Http\middleware\checkAge::class ]  
 Route::get('test', 'home controller@test')->middleware('age');  
 public function test() {  
 dd(test);  
 public function handle(\$request, Closure \$next) {  
 dd(\$age);  
 return \$next(\$request); }  
 نحوی و محتوا , Route on middleware  
 . سفارشی دادن Route , Age  
 ۱۴۰۴  
 ۲۰۲۳  
 ۱۴۰۴

۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	۱۵	۱۶	۱۷	۱۸	۱۹	۲۰	۲۱	۲۲	۲۳	۲۴	۲۵	۲۶	۲۷	۲۸	۲۹	۳۰	۳۱
۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	۱۵	۱۶	۱۷	۱۸	۱۹	۲۰	۲۱	۲۲	۲۳	۲۴	۲۵	۲۶	۲۷	۲۸	۲۹	۳۰	۳۱



جمعه  
۸ September  
۲۲ صفر

b: guest enter  
 @guest  
 @endguest  
 نحوی http که درست هاست : middleware  
 . اینجا ب لاین input (Validation) : middleware validation که  
 . اینجا ب لاین http (Middleware)  
 app->Http->middleware  
 . green (to middleware) ← check  
 php artisan make: middleware CheckAge  
 controller , Route on middleware  
 نحوی و محتوا  
 . نوشته در middleware jobs در چند گزینه  
 public function handle(\$request, Closure \$next) {  
 // do something  
 return \$next(\$request); }  
 نوشته در middleware  
 . register up to web middleware micro  
 app->Http->kernel.php  
 # middleware jobs  
 protected \$middleware = [  
 \App\Http\middleware\middleware::class ];  
 نحوی و محتوا در خدمت کارکردن  
 ۱۴۰۴  
 ۲۰۲۳  
 ۱۴۰۴

۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	۱۵	۱۶	۱۷	۱۸	۱۹	۲۰	۲۱	۲۲	۲۳	۲۴	۲۵	۲۶	۲۷	۲۸	۲۹	۳۰	۳۱
۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	۱۵	۱۶	۱۷	۱۸	۱۹	۲۰	۲۱	۲۲	۲۳	۲۴	۲۵	۲۶	۲۷	۲۸	۲۹	۳۰	۳۱





چهارشنبه

13 September  
صفر ۲۷

میانبر : middleware  
میانبر که میان دو مسیر وارد و خروجی قرار دارد و این میانبر استفاده شده است.

```
Route::get('/', 'homeController@index')->middleware('auth:api');
```

guard پروردگار  
این دو سیستم برای web و api میباشد.

```
// config/auth.php  
'guards' =>
```

کاربر : act

```
'admin' => [  
    'driver' => 'session',  
    'provider' => 'admin', ],  
  
'writer' => [  
    'driver' => 'session',  
    'provider' => 'writers', ],
```

شما در صورتی که در این اعلان دار  
و چند نوشتیده  
آن توپوس ها دسترسی به این اعلان خواهند  
داشت و این میتواند معمولی نباشد و درست بارز است  
که اینها کوئی است که نیت شما اینجا نباشد.

```
'providers' =>
```

```
'admin' => [  
    'driver' => 'eloquent',  
    'model' => App\ admin::class],
```

```
'writer' => [  
    'driver' => 'eloquent'  
    'model' => App\ writer::class],
```

user , admin و writer

۱۴۰۴

2023

۱۴۰۴

۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	
۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴



سه شنبه

12 September  
صفر ۲۶

guard : guard  
بروز اخیر حیثیت را بخواهد استفاده میکند.

که بصورت دینامیک ساخته میشود

api نویکار

config> auth.php

"guards" =>

```
'web' => [  
    'driver' => 'session',  
    'provider' => 'users', ],
```

نهایت دسترسی خارج از session است

```
'api' => [  
    'driver' => 'token',  
    'provider' => 'users',  
    'hash' => false, ],
```

"providers" =>

users =>

```
'driver' => 'eloquent',  
'model' => App\user::class,
```

ردیف دویست و سی و هشت

```
'user' => [  
    'driver' => 'database',  
    'table' => 'users', ],
```

ردیف هشتاد و هشت

"defaults" =>

```
"guard" => "web",  
"passwords" => "users", ]
```

ردیف هشتاد و هشت

۱۴۰۴

2023

۱۴۰۴

۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴
۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴	۱۴۰۴