

Praktikum 8 - Pemrograman Web

Program Studi: Teknik Informatika

Lakukan praktikum dibawah ini, dan buat screenshot untuk pembuktian mengerjakan setiap poin dengan mengisi tabel dibawah, kemudian tunjukan hasil akhir dari men-share repository github yang telah dibuat.

A. Menyiapkan Database

1. Aktifkan Apache, dan database mysql anda pada xampp control panel,
2. Buatlah sebuah database dengan nama **webtoko**
3. Buatlah sebuah tabel dengan nama **tblproduk**
4. Dalam tabel produk terdiri dari beberapa field seperti id (primary, auto increment), nama, deskripsi, dan harga.

Table name: Add column(s)

Structure

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index
<input type="text" value="id"/> <small>Pick from Central Columns</small>	INT	11	None			<input type="checkbox"/>	PRIMARY <input checked="" type="checkbox"/>
<input type="text" value="nama"/> <small>Pick from Central Columns</small>	VARCHAR	100	None			<input type="checkbox"/>	---
<input type="text" value="deskripsi"/> <small>Pick from Central Columns</small>	TEXT		None			<input type="checkbox"/>	---
<input type="text" value="harga"/> <small>Pick from Central Columns</small>	INT	7	None			<input type="checkbox"/>	---

5. Isi data minimal 20 item seperti pada gambar dibawah ini (data tidak harus sama seperti pada gambar)

	id	nama	deskripsi	harga
<input type="checkbox"/>	1	Pepsodent	Pepsodent adalah merek pasta gigi yang tertua di I...	10000
<input type="checkbox"/>	2	Lux	Lux adalah merek sabun kecantikan yang dikelola ol...	12000
<input type="checkbox"/>	3	Mama Lemon	Sabun cuci piring merupakan bahan pencuci yang dig...	16000
<input type="checkbox"/>	4	Hayat (Minyak Goreng)	Minyak masakan (atau lebih dikenal dengan istilah ...	18000

B. Membuat Model Laravel

1. Persiapkan project baru di laravel, dan lakukan perintah berikut.
2. Pada file `.env` ubah setting database seperti setting database yang kalian miliki. Masukkan nama database **webtoko** dan **username/password** sesuai dengan yang kalian miliki, untuk default (user: root, password: null)

```
.env
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:Ac1hPOpiAXVE2eBoVp5Csp3FQN79VxwYU5LHTpp5avQ=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=dbjualmobil
15 DB_USERNAME=root
16 DB_PASSWORD=
17
```

3. Buatlah model untuk tabel produk dengan menggunakan php artisan **php artisan make:model Produk** (Penamaan model sesuai konvensi penamaan yaitu Camel Case)
4. Buka file Produk.php pada app/Models/Produk.php
Tambahkan protected \$table = 'tblproduk' ← sesuai dengan nama tabel di database

```
app
├── Console
├── Exceptions
├── Http
├── Models
│   ├── Produk.php
│   └── User.php
├── Providers
├── bootstrap
├── config
├── database
└── factories

1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Produk extends Model
9 {
10     use HasFactory;
11     protected $table = 'tblproduk';
12 }
13
```

5. Setelah proses ini artinya class Model Produk sudah siap digunakan mewakili tblproduk yang ada di mysql.

C. Penggunaan Model pada Controller

1. Buatlah sebuah controller untuk menyajikan kontrol ListProduk dengan php artisan
Php artisan make:controller ListProdukController
(Penamaan Controller Camel Case dan ditambahkan kata Controller)
2. Buka file ListProdukController pada app/Http/Controllers/ListProdukController.php
3. Pada controller tambahkan nama model
use App\Models\Produk

```
app > Http > Controllers > ListProdukController.php
Click here to ask Blackbox to help you code faster
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use App\Models\Produk;
7
8  class ListProdukController extends Controller
9  {
```

4. Buat sebuah method dengan nama show() untuk menarik data dari database ke controller

```
app > Http > Controllers > ListProdukController.php
5  use Illuminate\Http\Request;
6  use App\Models\Produk;
7
8  class ListProdukController extends Controller
9  {
10     public function show()
11     {
12         $data = Produk::get();
13         foreach ($data as $produk) {
14             $nama[] = $produk->nama;
15             $desc[] = $produk->deskripsi;
16             $harga[] = $produk->harga;
17         }
18         return view('list_produk', compact('nama', 'desc', 'harga'));
19     }
20 }
21 }
```

Pada gambar diatas, kita mengambil **seluruh data** di model Produk, dan model ini berasosiasi langsung dengan tblproduk di database.

Kemudian melakukan looping dan memasukan ke variable \$nama, \$desk, dan \$harga

Perhatikan nama, desc, dan harga. Desc adalah variable di laravel, sedangkan deskripsi adalah nama field di database.

list_produk adalah view yang nanti akan kita buat.

5. Buatlah Routing menuju Controller ListProdukController pada path /listproduk dengan menambahkan pada file routes/web.php
Tambahkan **use App\Http\Controllers>ListProdukController;**
Dan tambahkan **Routes::get** (seperti pada gambar dibawah ini)

```
routes > web.php
... | Click here to ask Blackbox to help you code faster
1  <?php
2
3  use Illuminate\Support\Facades\Route;
4  use App\Http\Controllers>ListProdukController;
5
6
7  Route::get('/listproduk', [ListProdukController::class, 'show'] );
8
```

Memanggil method “Show” pada class ListProdukController

C. Menampilkan View

1. Buatlah view dengan nama list_produk dengan menggunakan perintah **php artisan make:view list_produk**
(Ingat untuk view menggunakan snake case, lowercase)
2. Pada file resources/views/list_produk.blade.php buatlah kode program seperti gambar berikut

```
resources > views > list_produk.blade.php
Click here to ask Blackbox to help you code faster
1
2
3  <div class="ml-10 mt-20">
4  @foreach ($nama as $index => $item)
5      Nama Produk {{$index + 1}}: {{ $item }} <br>
6      Deskripsi Produk {{$index + 1}}: {{ $desc[$index] }} <br>
7      Harga Produk {{$index + 1}}: {{ $harga[$index] }} <br>
8  @endforeach
9  </div>
10
```

3. Jalankan Laravel, php artisan serve
4. Buka browser dan kunjungi localhost:8000/listproduk

D. Menampilkan View (Extra)

1. Kamu juga bisa memodifikasi view agar terlihat rapih dengan menggunakan tabel seperti pada gambar dibawah ini.

```
resources > views > 🐞 list_produk.blade.php
💡 Click here to ask Blackbox to help you code faster
1  <div class="ml-10 mt-20">
2      <table>
3          <thead>
4              <tr>
5                  <th>No</th>
6                  <th>Nama Produk</th>
7                  <th>Deskripsi Produk</th>
8                  <th>Harga Produk</th>
9              </tr>
10         </thead>
11         <tbody>
12             @foreach ($nama as $index => $item)
13                 <tr>
14                     <td>{{ $index + 1 }}</td>
15                     <td>{{ $item }}</td>
16                     <td>{{ $desc[$index] }}</td>
17                     <td>{{ $harga[$index] }}</td>
18                 </tr>
19             @endforeach
20         </tbody>
21     </table>
22 </div>
23
```

2. Pelajari penyajian view diatas, dan sajikan data produk anda dengan menggunakan template tailwind dengan tampilan yang lebih menarik dengan menggunakan template.
3. SCREENSHOT hasil tabel produk dengan template, akan dijadikan penilaian akhir di praktikum ke-8 ini.

Hasil Pengerjaan

No.	Instruksi	Screenshot	Kendala/Saran
A.	Extends - Layout		
1.			
2.			
...			
B.	Components → Include		
1.			
2.			
...			