1. Apa itu Laravel?

Laravel adalah framework aplikasi web berbasis PHP dengan sumber terbuka, menggunakan konsep Model-View-Controller. Laravel juga dapat mempercepat pengerjaan aplikais website karena sudah disediakan berbagai fitur dan library.

1. Apa perbedaan antara Laravel dan framework PHP lainnya

Laravel, sebuah framework PHP terkenal, menonjol dengan sintaks yang ekspresif, fitur-fitur bawaan seperti otentikasi dan manajemen sesi, serta penggunaan Eloquent ORM untuk interaksi dengan basis data.

1. Apa kegunaan dari migration dalam Laravel?

Migration adalah sebuah fitur untuk mengelola struktur basis data secara efisien. Dengan migration ini dapat mendefinisikan skema basis data dalam kode PHP, mengelola versi skema, dan melakukan perubahan struktur tanpa perlu melakukan intervensi manual di database.

1. Bagaimana cara membuat route di Laravel?

Di Laravel, route dibuat melalui file web.php. Dengan metode seperti Route::get(), Route::post(), dll., menentukan jenis HTTP request dan tindakan yang dijalankan.

1. Apa yang dimaksud dengan Eloquent ORM dalam konteks Laravel?

Eloquent ORM adalah fitur dalam Laravel adalah fitur untuk berinteraksi dengan basis data menggunakan objek-objek PHP. Ini memungkinkan operasi CRUD pada tabel basis data menggunakan model PHP yang terkait dengan struktur data dalam tabel tersebut.

1. Apa itu middleware dalam Laravel dan bagaimana cara kerjanya?

Middleware dalam Laravel adalah lapisan perantara untuk memfilter permintaan HTTP. Saat permintaan diterima, middleware dipanggil untuk memprosesnya dengan logika seperti autentikasi atau validasi. Jika memenuhi kriteria, proses dilanjutkan; jika tidak, respons sesuai dikirimkan.

1. Apa yang dimaksud dengan Blade dalam Laravel?

Blade adalah tempalting engine dalam Laravel yang memungkinkan penulisan template HTML dengan sintaks bersih dan minimalis.

1. Apa yang dimaksud dengan CSRF protection dalam Laravel dan mengapa penting?

CSRF protection dalam Laravel adalah fitur yang mencegah serangan penipuan permintaan lintas situs. Ini dilakukan dengan menggunakan token CSRF yang disertakan dalam setiap permintaan POST. Pentingnya CSRF protection adalah melindungi aplikasi dari serangan yang dapat memaksa pengguna untuk melakukan tindakan tidak disengaja, menjaga keamanan data, mematuhi standar keamanan, dan mengurangi risiko kerentanan terhadap serangan penipuan dan manipulasi data.

1. Bagaimana cara melakukan validasi input data pada form menggunakan Laravel?

Langkah pertama adalah menentukan aturan validasi, seperti jenis data yang diharapkan dan panjang minimum atau maksimum. Validasi kemudian dilakukan di dalam controller menggunakan fitur bawaan Laravel, seperti method validate(). Jika validasi gagal, pengguna akan diarahkan kembali ke form dengan pesan kesalahan yang sesuai. Jika berhasil, tindakan selanjutnya, seperti penyimpanan data, dapat dilakukan.

1. Apa itu "Eloquent Relationships" dalam Laravel dan bagaimana cara menggunakannya?

"Eloquent Relationships" dalam Laravel adalah fitur yang memungkinkan pengembang mendefinisikan hubungan antara model dalam basis data. Dengan menggunakan metode seperti hasOne(), hasMany(), belongsTo(), dan belongsToMany().

1. Apa perbedaan antara "soft delete" dan "hard delete" dalam Laravel?

Dalam Laravel, "soft delete" melibatkan penandaan data yang dihapus dan mempertahankan mereka dalam database, sementara "hard delete" menghapus data secara permanen tanpa jejak. Soft delete memungkinkan pemulihan data yang dihapus, sedangkan hard delete tidak. Soft delete menggunakan fitur Eloquent "Soft Deletes", sementara hard delete adalah penghapusan langsung dari database.

1. Bagaimana cara meng-handle error dan exception dalam Laravel?

Dalam Laravel, error dan pengecualian ditangani melalui kelas App\Exceptions\Handler, yang mengatur cara pengecualian ditampilkan. Pengecualian juga bisa ditangkap langsung dengan blok try-catch. Umpan balik yang jelas diberikan kepada pengguna dengan menampilkan pesan error atau halaman khusus.

1. Bagaimana cara membuat dan menggunakan migration untuk mengubah struktur database dalam Laravel?

Dalam Laravel, untuk mengubah struktur database, kita gunakan migration. Pertama, buat migration baru dengan php artisan make:migration. Buka file migration tersebut dan tentukan perubahan yang diinginkan. Jalankan migrasi dengan php artisan migrate untuk menerapkan perubahan. Opsionalnya, kita bisa membatalkan migrasi dengan php artisan migrate:rollback. Dalam file migration, kita bisa tambahkan query tambahan sesuai kebutuhan.