Proyecto de laravel con debian

Link de repositorio remoto

Episodio 5

Explicacion de rutas
Aplicar negrita a la vista de welcome
Funciones que se pueden poner dentro del return de las rutas
Codigo

Episodio 6

Cambiar welcome page
Agregar CSS
Crear alertas con JS
Codigo

Episodio 7

Agregar mas CSS
Primeros blog Posts
Codigo

Episodio 8

Agregar las rutas y archivos para obtener el contenido de los post de manera dinamica
Codigo

Episodio 9

Atrapar los errores posibles que puede tener una ruta, aprender de expresiones regulares
Codigo

Episodio 10

Utilizar cache para economizar recursos de peticiones al servidor
Codigo

Episodio 11

Utilizar clases para mejorar el codigo, de manera que encontrar los posts sea mas facil y no ensucie tanto el codigo de rutas
Utilizar el FileSystem para obtener de manera dinamica el numero y el contenido de los posts existentes
Introduccion a la metadata
Codigo

Episodio 12

Utilizar librerias externas para manejar metadata (Spazie YamlFrontMatter)
Utilizar collections para iterar y mapear de mejor manera
Codigo

Episodio 13

Sortear por fecha para acomodar los post
Cachear el resultado de los post para no hacer futuras peticiones al servidor de informacion constante
Codigo

Episodio 14

Introduccion a Blade
Diferentes sintaxis para mejorar el rendimiento del codigo
Codigo

Episodio 15

Introduccion e implementacion de layout files para facilitar el reciclaje de codigo en los views
Codigo

Episodio 16

Implementar un findOrFail para atrapar el error cuando una ruta de post no existe
Codigo

Episodio 17

Introduccion a la conexion a base de datos
Se cambian parametros en el archivo .env para poder conectar a nuestra instancia de MySql
Codigo

Episodio 18

Introduccion a las migraciones, estructura de la base de datos y rollbacks
Tip: Laravel bloquea las migraciones para limpiar la base de datos si el ambiente es de produccion `php artisan migrate:fresh`
Codigo

Episodio 19

Introduccion a los modelos, creacion de usuarios e interaccion con tinker
Comandos como:

```
$user-> new User;
$user -> save();
$user -> Find::all();
```

Codigo

Episodio 20

Creacion de modelos y migraciones utilizando PHP artisan

```
php artisan make:migration create_posts_table
```

Crear modelo

```
php artisan make:model Post
```

Codigo

Episodio 21

Consultas a la base de datos para actualizar el primer post

```
$post = App\Models\Post::first();

$post->body = = new body;
$post->save();
```

Codigo

Episodio 22

Crear post con el siguiente codigo:

```
Post::create(['title'=>'My second post','excerpt'=>'Lorem Ipsum',
'body'=>'Lorem ipsum Lorem ipsum Lorem Ipsum']);
```

Agregar el $fillable en el modelo de Post para que el codigo anterior corra de manera correcta
Codigo

Episodio 23

Mejorar la busqueda del post mediante el cambio de enviar un `Post $post` en lugar de un `$id` en el request del Endpoint

```
Route::get('/posts/{post}', function (Post $post) {
```

```
    // $post = Post::findOrFail($id);

    return view('post',['post' => $post]);

});
```

Codigo

Episodio 24

Crear migracion y modelo para agregar categorias a los blog posts

```
php artisan make:model Category -m
```

Se crea una relacion entre cada post y categoria mediante una llave foranea llamada `category_id` en el modelo de *post* Codigo

Episodio 25

Crear la relacion Post - Categoria pero desde el modelo de *Category* mediante este codigo:

```
public function posts(){

        return $this->hasMany(Post::class);
    }
```

Para poder consultar cada categoria y traer todos los posts que pertenecen
Codigo

Episodio 26

Solucionar el llamado 'N+1 problem'
Instalar la libreria 'Clockwork'

```
composer require itsgoingd/clockwork
```

Tambien se puede agregar la extension del navegador de manera opcional
Codigo

Episodio 27

Agregar creador de cada post mediante una relacion similar a cada post con su categoria
Crear un seeder para poblar la base de datos de manera automatica

```
php artisan db:seed
```

Codigo

Episodio 28

Utilizar factories para crear contenido en la base de datos de manera aleatoria y de forma rapida

```
php artisan make:factory PostFactory
php artisan make:factory CategoryFactory
```

Codigo

Episodio 29

Ver todos los posts relacionados al autor similar al proceso realizado con las categorias
Codigo

Episodio 30

Solucionar el N+1 previamente detallado pero en las categorias de cada post mediante el siguiente codigo en Post

```
protected $with = ['category','author'];
```

Codigo