

Crear proyecto

1. Crear un nuevo proyecto en Laravel utilizando el comando **composer create-project --prefer-dist laravel/laravel miProyecto**
2. Lograr que nuestro proyecto se ejecute en un servidor local utilizando el comando **php artisan serve**

Controllers y Request

1. Crear una ruta en /peliculas. Esta debe imprimir el título películas pasando previamente por un **PeliculasController**
2. Crear una ruta /pelicula/{id} donde **{id}** puede ser el id de una película. Esto pasará por **PeliculasController** en otro método. La vista debe decir “Usted esta viendo la película N” donde **N** representa el número que llega a través de la URL
3. Crear una ruta /peliculas/buscar/{texto} donde **{texto}** puede ser cualquier texto a buscar en el título de una película. Esto debe pasar por **PeliculasController** en un método distinto y en la vista imprimir “Se muestran las películas que contengan la palabra **{texto}**”.

Middleware

1. Crear un archivo Middleware llamado “CaidoPorMantenimiento”.
2. Dentro de la ejecución del middleware incluir únicamente la línea:

```
throw new HttpException(503);
```

3. Registrar dicho Middleware dentro de todas las rutas. ¿Qué sucede? Luego de experimentar, quitemos el registro de este middleware.

Blade y Controllers

1. En nuestro archivo de rutas:
 - a. Agregaremos una variable \$peliculas con el siguiente contenido:

```
$peliculas = [  
    "Toy Story",  
    "Buscando a Nemo",  
    "Avatar",  
    "Star Wars: Episodio V",
```

```
        "Up",  
        "Mary and Max"  
    ];
```

- b. Esta información debe llegar a la vista correspondiente a **/peliculas** aunque la imprimiremos en la próxima consigna.
2. Mostrar en la vista el listado de películas utilizando php en su sintaxis clásica.
3. Modificar el archivo anterior para que se vean las películas utilizando la sintaxis de blade.
4. Mostrar en la vista un texto que diga "No hay películas" en el caso de que no traiga películas.
5. Utilizar la ruta **/pelicula/{id}** para que muestre la película correcta de la variable generada. Por ejemplo, si llega el **id** 2 debe mostrar Avatar.
6. Modificar esta nueva vista para que avise si el número de película indicado no existe.
7. Agregar una ruta **/agregarPelicula** que muestre el contenido de agregarPelicula.php pasando por **PeliculasController**.
8. Modificar la vista recién creada para que contenga la sintaxis de blade.
9. Lograr que si el pedido a **/agregarPelicula** es a través de POST se redirijan los datos a **PeliculasController** con el método **agregarPelicula**. Este método debe devolver el string "Película Agregada".
10. Modificar el método **agregarPelicula** para que redirija a una nueva vista (peliculaAgregada.blade.php)
11. Utilizando el objeto **Request** y el metodo **all** imprimir lo que llegó desde el formulario en la vista.
12. Modificar todas las vistas creadas durante la clase de hoy para que la parte esencial de un archivo HTML (tags <html> <head> y <body>) estén en un archivo de layout. Esto haría que el contenido puntual de cada vista sea una **@section**