

1. Agregar en el detalle de la película un listado de sus actores. Para esto necesitaremos como mínimo crear en el modelo de **Pelicula** un método **actores** que utilice **belongsToMany**. A continuación se recuerda el uso de la misma:

```
return $this->belongsToMany('App\NombreModeloFinal', 'tabla_intermedia',
'fk_intermedia_hacia_esta_clase', 'fk_intermedia_clase_destino');
```

Pagination

1. Crear un comando donde se podran visualizar 5 actores por pagina y se “pagine” automaticamente el resto ([ayuda](#)).

Colecciones

1. Modificar el detalle de una película para que:
 - a. Muestre solo los actores con rating mayor que 7.
 - b. Muestre solo los actores que hayan actuado en al menos 5 películas.
2. Usando métodos de colecciones, modificar el listado de películas para que:
 - a. Se muestren ordenadas por título.
 - b. Se muestren ordenadas por duración.
 - c. Se muestren ordenadas al azar.
 - d. Se muestren únicamente las películas que duren más de 90 minutos.
 - e. Se muestren únicamente las películas que duren más de 90 minutos o tengan rating mayor a 5.
 - f. Además, se muestre un valor de “premios por minuto” (generado a través del método **map**)
3. A modo de prueba, modificar el listado de películas para que la variable que se envía por parámetro a la vista deje de ser una colección y sea únicamente un array con los nombre de los géneros y el id del mismo como clave. Si no existe, crearlo. Utilizar para esto el método **pluck**.

STORAGE

1. Crear un formulario en el podamos subir una imagen.
2. Crear el método storage que pueda subir la imagen con nombre aleatorio.
 - a. Este método debe retornar una vista con la imagen subida.

3. Crear otro método que pueda subir una imagen y guardarla con un nombre de imagen, este nombre puede ser un hash o el mismo nombre del archivo.
 - a. Este método también debe retornar una vista donde se muestre la imagen subida.

----- Fin de la ejercitación -----