

## Ejercicios 3

En esta sección de ejercicios vamos a continuar con la aplicación iVideoclub para enlazar todas las páginas que habíamos creado y además añadir una nueva página con la vista detalle de una película.

### Ejercicio 3.1 - Enlazar las páginas

Empezaremos por enlazar todas las páginas:

- Modifica la página principal "home" para que al pulsar sobre "Acceso a iVideoclub" se abra la página "videoclub" y para que al pulsar sobre "Consultar información" se abra la página "autor".
- Modifica la página "autor" para que al pulsar sobre "Currículum vitae" se abra la página "curriculum".
- Modifica el toolbar de las páginas videoclub, autor y curriculum para añadir el botón que permite volver a la página anterior.

Recuerda que para abrir una nueva página lo podemos realizar tanto con la propiedad "routerLink" desde la propia plantilla, como añadiendo un evento "click" para que ejecute una función definida en la clase .ts que realice dicha acción. Para este ejercicio se recomienda utilizar en todos los casos la primera opción, es decir, enlazar las páginas directamente desde las vistas.

### Ejercicio 3.2 - Ciclo de vida de las páginas

Modifica las páginas "home", "autor", "curriculum" y "videoclub" para que muestren por consola todos los eventos de su ciclo de vida. Por ejemplo, para el evento "ionViewDidEnter" de la página "videoclub" tendríamos que añadir el siguiente código al fichero "videoclub.ts":

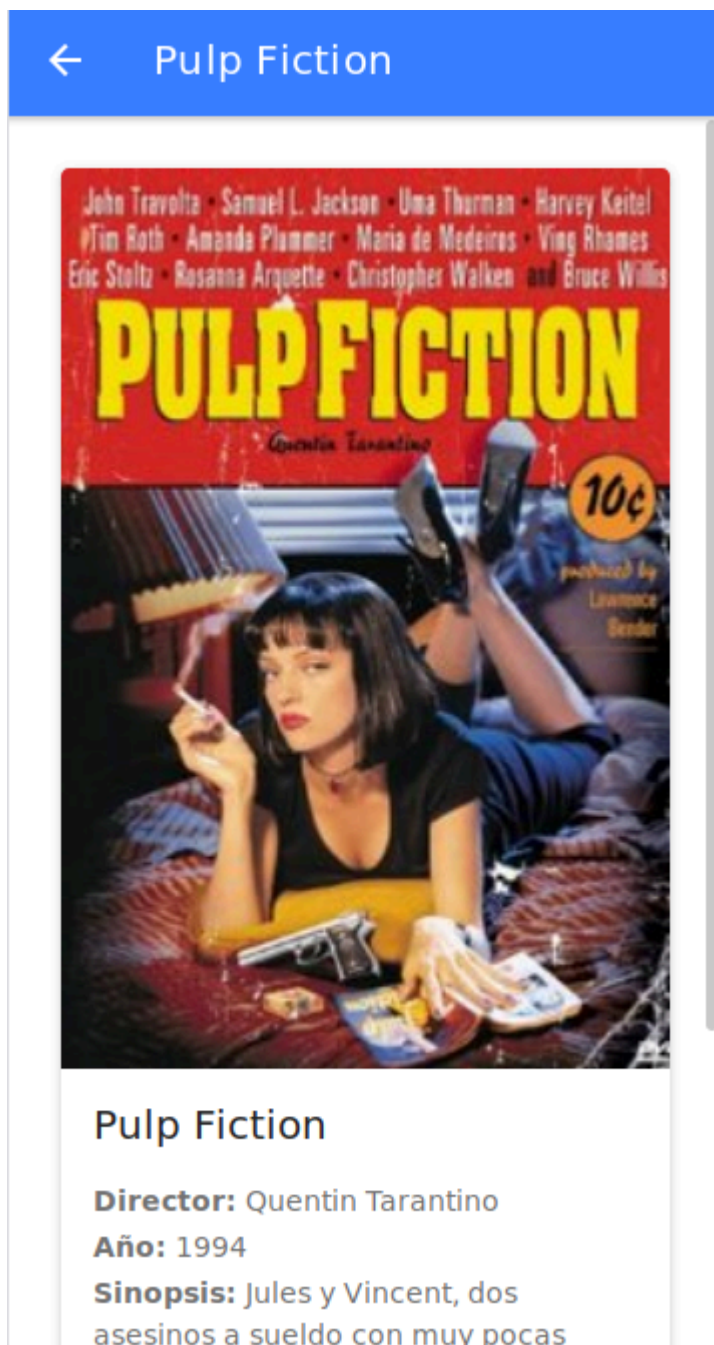
```
1  ionViewDidEnter() {  
2    console.log('ionViewDidEnter VideoclubPage');  
3  }
```

Revisa el orden en el que suceden estos eventos al abrir y cerrar páginas. Emula la aplicación en el navegador y abre las herramientas del desarrollador para ver el log de eventos por la

consola. Si utilizas la opción "ionic serve --lab" tendrás que dejar activado el emulador de una sola plataforma para que no se dupliquen los eventos (puedes seleccionar los emuladores a mostrar desde las opciones de menú de la parte superior derecha).

## Ejercicio 3.3 - Nueva página detalle

Vamos a crear una nueva página llamada "detalle" en la que mostraremos la información ampliada de un película. Esta pantalla tendrá un aspecto similar al de la siguiente captura:



Esta pantalla se ha creado utilizando una tarjeta ( `ion-card` ). Recuerda cambiar también el título del *toolbar* superior para que indique el título de la película y añadir el botón para volver a la página anterior.

## Ejercicio 3.4 - Enlazar la página detalle

Modifica la plantilla de la página " `videoclub` " para que al pulsar sobre cada elemento de la lista se abra la pantalla detalle. Para esto añade a la etiqueta `ion-item` de cada elemento de la lista de películas el siguiente evento:

```
1 (click)="verPaginaDetalle(1)"
```

Este evento llamará al método " `verPaginaDetalle` " (que ahora definiremos) pasándole como parámetro un número fijo. De momento vamos a pasar a la función este número a modo de identificador (puede ser cualquier número), más adelante modificaremos este parámetro para asociarlo con la película.

Modifica la clase " `videoclub.ts` " para añadir el método `verPaginaDetalle` asociado a este evento. Al abrir la nueva página le pasaremos como parámetro el valor recibido (que como hemos dicho de momento es un número):

```
1 verPaginaDetalle(id): void {  
2   this.router.navigate(['/detalle', id]);  
3 }
```

Una vez hecho esto, desde la página " `detalle` " ya podemos recuperar la película sobre la que se ha pulsado en el listado. Para realizar esta acción añade la siguiente instrucción en el evento **ngOnInit** de la clase:

```
1 const id = this.activatedRoute.snapshot.paramMap.get('id');
```

Muestra la variable `id` por consola ( `console.log(id)` ) para asegurarte de que está recibiendo el valor correcto.