

Ejercicio 2

Descargar este enunciado [pdf](#) o [html](#)

Partiendo del [Ejercicio 1](#), se pide realizar las siguientes tareas:

1. Basándote en el componente `ItemListaCoches.kt` proporcionado, define la pantalla con la lista de coches en `GaleriaCochesScreen.kt` y su correspondiente `GaleriaCochesViewModel.kt` que gestione el filtrado usando Room, así como manteniendo los estados y gestionando los eventos de `GaleriaCochesScreen` en el paquete `.galeriacoches`.



Aviso

La lista **NO debe contener** ningún tipo de animación al añadir o borrar elemento de la misma.

| | | | |
|--|-------------------|--------------------------|-----|
| | Volkswagen Passat | 43210 € | |
| | Ford Mustang | 49500 € antes 55000 € | 10% |
| | Audi A4 | 46000 € | |
| | BMW X5 | 65100 € antes 86800 € | 25% |
| | Mercedes Clase C | 50000 € | |
| | Volkswagen Golf | 29000 € | |
| | Ford Focus | 28500 € antes 30000 € | 5% |
| | Audi A3 | 32000 € | |

2. La lista de coches debe estar **contenida en un Scaffold** con un `CenterAlignedTopAppBar` que **colapse al hacer scroll** en la lista y un `BottomAppBar` que cambiará dependiendo de si tenemos seleccionado o no algún coche en la misma.
 - Si no hay ningún coche seleccionado la `BottomAppBar` debe ofrecer dos componentes `FilterChipWithIcon` definidos en `com.vehiculos.ui.composables.FilterChipCommon.kt`:
 - Uno para ordenar los coches por precio con el icono `Icons.Filled.SortByAlpha`.
 - Otro para mostrar solo los coches con descuento o en oferta con el icono `Icons.Filled.LocalOffer`.
 - Si tenemos un coche seleccionado la `BottomAppBar` debe ofrecer dos botones de acción:
 - Uno para **ver la ficha del coche seleccionado** con el icono `Icons.Filled.Edit`.
 - Otro para **eliminar el coche seleccionado** con el icono `Icons.Filled.Delete`.

| | | | |
|--|------------|--------------------------|----|
| | Ford Focus | 28500 € antes 30000 € | 5% |
| | Audi A3 | 32000 € | |
| <div><div>AZ Precio</div><div> Descuento</div></div> | | | |

| | | | |
|-----------------------------------|------------|--------------------------|----|
| | Ford Focus | 28500 € antes 30000 € | 5% |
| | Audi A3 | 32000 € | |
| <div><div></div><div></div></div> | | | |

3. Tanto la ordenación por precio como el filtrado por coches en oferta se realizarán en el `GaleriaCochesViewModel.kt` llamando a los métodos correspondientes del repositorio que usarán las operaciones definidas en la BD con Room. Para ello deberás definir clase `GaleriaCochesEvent` que será un 'Sum-Type' con los siguientes eventos:
- `OnEditarFicha` que reciba un callback que me permita navegar a la pantalla de edición de coche pasándole el id del coche seleccionado.
 - `OnSeleccionarCoche` que recibirá el `CocheUiState` seleccionado.
 - `OnOrdenarPorPrecio` .
 - `OnFiltrarPorDescuento` .



Pista

Puedes definir el método privado ...

```
private suspend fun getCohes() : List<CocheUiState>
```

en el `GaleriaCochesViewModel.kt` que te devuelva la lista de coches ordenada y filtrada si fuera necesario usando los estados booleanos que en el ViewModel `ordenarPorPrecioState` y `filtrarPorDescuentoState` .