

RxJS en Angular - Conceptos y Usos Reales

1. switchMap

Concepto: Cancela la suscripción anterior y se suscribe al nuevo Observable.

Uso real: Búsqueda en tiempo real, como en un campo de autocompletado.

Ejemplo:

```
this.searchForm.valueChanges.pipe(  
  debounceTime(300),  
  switchMap(term => this.api.search(term))  
)  
.subscribe(data => this.results = data);
```

2. mergeMap

Concepto: Mapea a un nuevo Observable y se suscribe a todos en paralelo.

Uso real: Cargar datos de varios endpoints sin cancelar las solicitudes previas.

Ejemplo:

```
from(userIds).pipe(  
  mergeMap(id => this.api.getUserDetails(id))  
)  
.subscribe(user => this.users.push(user));
```

3. forkJoin

Concepto: Ejecuta varios Observables en paralelo y devuelve un único resultado cuando todos finalizan.

Uso real: Obtener datos iniciales de configuración y usuario antes de renderizar.

Ejemplo:

```
forkJoin({  
  config: this.api.getConfig(),  
  user: this.api.getUser()  
}).subscribe(({ config, user }) => {  
  this.config = config;  
  this.user = user;  
});
```

4. combineLatest

Concepto: Combina el último valor emitido por varios Observables.

Uso real: Filtrar resultados combinando el valor de un buscador y un selector de categoría.

Ejemplo:

```
combineLatest([  
  this.searchTerm$,  
  this.selectedCategory$  
)  
.pipe(  
  map(([term, category]) => this.filterItems(term, category))  
)  
.subscribe(filtered => this.filteredItems = filtered);
```