

# exhaust

---

Convierte un Observable de orden superior en uno de primer orden ignorando los Observables internos mientras el Observable interno actual no se haya completado

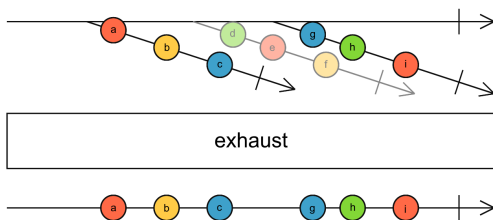
---

## ► Signatura

---

## Descripción

Convierte un Observable de orden superior en uno de primer orden ignorando los Observables internos mientras el Observable interno actual no se haya completado.



`exhaust` se suscribe a un Observable que emite Observables, también conocido como Observable de orden superior. Cuando recibe un Observable interno, el Observable resultante comienza a emitir sus emisiones. Hasta este punto, se comporta igual que `mergeAll`. Sin embargo, `exhaust` ignora cada Observable interno nuevo hasta que el Observable actual no se haya completado. Una vez esté completo, aceptará el siguiente Observable interno y comenzará a emitir sus valores, repitiendo el proceso.

---

## Ejemplos

**Ignorar todos los Observables internos hasta que el Observable interno actual esté completo**

StackBlitz

```
import { exhaust, map, delay } from "rxjs/operators";
import { ajax } from "rxjs/ajax";
import { of } from "rxjs";

const pokemonId$ = of(1, 5, 6);

function getPokemonName(id: number) {
  return ajax.getJSON(`https://pokeapi.co/api/v2/pokemon/${id}`).pipe(
    map(({ name }) => name),
    delay(2000)
  );
}

pokemonId$
  .pipe(
    map((id) => getPokemonName(id)),
    exhaust()
  )
  .subscribe(console.log);
// Salida: bulbasaur
```

Ejemplo de la documentación oficial

**Por cada click, ejecutar un temporizador, únicamente si no hay ningún temporizador activo**