

every

Retorna un Observable que emite un valor booleano que determina si todas las emisiones de la fuente cumplen o no una condición

► Signatura

Descripción

Si todos los valores emitidos por el Observable fuente cumplen la condición especificada, `every` emite *true*. Si hay algún valor que no cumpla la condición, se emite *false*.

Ejemplos

Comprobar si todos los valores emitidos son numéricos

[StackBlitz](#)

```
import { of } from "rxjs";
import { every } from "rxjs/operators";

const number$ = of(1, 2, 3, 4);

number$.pipe(every((n) => Number.isInteger(n))).subscribe(console.log);
// Salida: true
```

[Copy](#)

Comprobar si todos los valores emitidos son menores a 2

[StackBlitz](#)

```
import { of } from "rxjs";
import { every } from "rxjs/operators";

const number$ = of(1, 2, 3, 4);

number$.pipe(every((n) => n < 2)).subscribe(console.log);
// Salida: false
```

Copy

Comprobar si todas las peticiones tienen un status 200 (todo OK)

StackBlitz

```
import { of } from "rxjs";
import { every, concatMap } from "rxjs/operators";
import { ajax } from "rxjs/ajax";

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

function getPokemon(id: number) {
  return ajax(`https://pokeapi.co/api/v2/pokemon/${id}`);
}

pokemonId$
  .pipe(
    concatMap((id) => getPokemon(id)),
    every(({ status }) => status === 200)
  )
  .subscribe(console.log);
// Salida: true
```

Copy

Ejemplo de la documentación oficial

Un simple ejemplo que emite *true* si todos los elementos son menores a 5, *false* en caso contrario