

# Difference between `of` operator and `Observable.of`

Asked 1 year, 2 months ago Active 1 year, 2 months ago Viewed 2k times



So what is the difference between `of` operator and `Observable.of` and what is the recommended way to create an observable?

2

```
import { Observable } from 'rxjs/Observable';
const obs$ = Observable.of(3);
```



or



1

```
import { of as observableOf } from 'rxjs';
const obs$ = observableOf(3);
```



rxjs

rxjs6

edited May 24 '18 at 11:09

asked May 24 '18 at 11:01



Rachid Oussanaa

5,211 9 42 65

What is the relation between Operator and Observable? – Ramesh Rajendran May 24 '18 at 11:02

@RameshRajendran I'm asking the difference between `Observable.of` and `'of'` not between `Observable` and `of` operator – Rachid Oussanaa May 24 '18 at 11:04

4 If you import `Observable`, you can use all global operators right away. Your second snippet requires modification of `import` every time you need to use another operator. IIRC, before RxJS 6 it was recommended way because tree shaking was broken, but in 6+ you can do what you did in the first snippet. Also, second form is more "functional", as you can compose it using `pipe` operator, but first form is closer to JS way. – alx May 24 '18 at 11:05

@alx "but in 6+ you can do what you did in the first snippet." No, you can't. "first form is closer to JS way" Why not the second one? – a better oliver May 24 '18 at 11:48

[stackoverflow.com/questions/50218784/...](https://stackoverflow.com/questions/50218784/...) – martin May 24 '18 at 12:01

## 1 Answer



5



The second way is the preferred way, and going forward (from rxjs v7) the only way. Currently in v6 you can still use the first by including the rxjs-compat package, but writing new code, you should stick to the last method mentioned.

answered May 31 '18 at 22:38



Per Hornshøj-  
Schierbeck

8,965 17 71 95

Got a question that you can't ask on public Stack Overflow? [Learn more](#) about sharing private information with Stack Overflow for Teams.

