

What is `return of ()` syntax

I'm looking at this snippet of code:

13



2

```
checkGuards(): Observable<boolean> {  
  if (this.checks.length === 0) return of (true);  
  const checks$ = from(this.checks);  
  const runningChecks$ = mergeMap.call(checks$, (s: any) => {  
    if (s instanceof CanActivate) {  
      return andObservables(  
        from([this.runCanActivateChild(s.path), this.runCanActivate(s.route)]));  
    } else if (s instanceof CanDeactivate) {  
      // workaround https://github.com/Microsoft/TypeScript/issues/7271  
      const s2 = s as CanDeactivate;  
      return this.runCanDeactivate(s2.component, s2.route);  
    }  
  });  
}
```

It's typescript. On the first line there is:

```
... return of (true);
```

What's this syntax?



asked Feb 16 '17 at 15:49



Max Koretskyi aka Wizard

51.8k 22 104 225

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up


OR SIGN IN WITH



Google



@AngularFrance, thanks, can you post it as answer? Actually the code is from angular sources. You're the second person saying that their code is not good to learn from – [Max Koretskyi aka Wizard](#) Feb 16 '17 at 16:23

Just did. From Angular sources you say? Interesting. Could you say which file? I'd like to take a look. Note that I said it wasn't the usual practice, not bad practice. Certainly the guys from the core team know what they're doing. ;) – [AngularChef](#) Feb 16 '17 at 16:29 

1 @AngularFrance, it's [here](#) – [Max Koretskyi aka Wizard](#) Feb 16 '17 at 16:36

Thanks, I'll take a look. – [AngularChef](#) Feb 16 '17 at 16:46

3 Answers



18



Most of the short function names in this snippet are **RxJS operators**.

Instead of:

- `Observable.of()`
- `Observable.from()`
- `Observable.mergeMap()`

You have:

- `of()`
- `from()`
- `mergeMap()`

Look at the top of the snippet, you should see the imports for these RxJS operators.

Note that this is NOT the usual practice (I think it makes the code harder to read).

answered Feb 16 '17 at 16:27



[AngularChef](#)

8,228

5

40

55

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google



Likely its a function called `of` in which is being passed `true` . The whitespace is irrelevant to the parenthesis.

2

answered Feb 16 '17 at 15:52

[Daniel A. White](#)

153k 40 302 381

`of` isn't a keyword of TypeScript, there must be a function somewhere named `of`

1

```
function of<T>(value: T) {  
    // ...  
}
```

answered Feb 16 '17 at 15:53

[Trident D'Gao](#)

8,464 10 62 120

1 Actually it is. in the `for.. of` statement. [For.. of](#) – [Leonardo Chaia](#) Feb 16 '17 at 16:02

1 `true true true true` – [Trident D'Gao](#) Feb 16 '17 at 16:32

well it's not `for ... of` definitely, see the [answer](#) – [Max Koretskyi aka Wizard](#) Feb 16 '17 at 16:40

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google

Facebook