# Type of lambda expressions in typescript

▲

4

▼

★

I'm trying to assign a lambda expression which returns a string, to a property, which, according to the API description, accepts the types (String | Object[] | Function).

```
this._popupTemplate.content = (feature) => {
        var template: string;
        //....
        return template;
    }
```

It seems to be working, however, webstorm says "

> assigned expression of type (feature:any) => string is not assignable to type string

"

So I tried using a type assertion: `<string>(feature) => {...}` which seems to have no effect. How can I satisfy webstorm (without suppression the information)?

`typescript`   `lambda`

asked Nov 8 '16 at 14:25

netik
**510**  2  8  25

---

1    Seems litke that `this._popupTemplate.content` is a string and not a `String | Object[] | Function`. Can you double check its typings definitions? Remember that typings definitions **may not be** equal to the API description, it could be outdated. – goenning Nov 8 '16 at 14:30

---

1 Answer

**3**

▼

✓

```
(feature) => {
    var template: string;
    //....
    return template;
}
```

is just easier way to write normal function like this

```
function(feature) {
    var template: string;
    //....
    return template;
}
```

The problem is you're trying to assign the function itself, not it's value. You got to execute it first. Add parentheses around the function (or lambda) and then execute it by adding parentheses with arguments after it. Like this:

```
this._popupTemplate.content = ((feature) => {
        var template: string;
        //....
        return template;
    })(feature);
```

answered Nov 8 '16 at 17:25

Erik Cupal
**1,344**   1   10   17

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

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

Email Sign Up    OR SIGN IN WITH    G Google    Facebook ✕