Using lodash in Angular 4

Asked 1 year, 6 months ago   Active 1 year, 6 months ago   Viewed 9k times

I have previously used lodash in my Angular 4 application by simply importing it and using it as shown here:

10

```
import lodash from 'lodash';

public getSubtotal() : number {
   return lodash.sumBy(this.cartItems, function(item) {
     return item.product.price * item.item.quantity;
   })
}
```

1

I am once again trying to use lodash similarly but am getting an error that lodash is undefined.

```
import lodash from 'lodash';

lodash.indexOf([1, 2, 1, 2], 2);
```

I get the following error:

```
ERROR TypeError: Cannot read property 'indexOf' of undefined
at CustomizeComponent.showTopping (customize.component.ts:39)
at Object.eval [as updateDirectives] (CustomizeComponent.html:285)
at Object.debugUpdateDirectives [as updateDirectives] (core.js:14638)
at checkAndUpdateView (core.js:13785)
at callViewAction (core.js:14136)
at execComponentViewsAction (core.js:14068)
at checkAndUpdateView (core.js:13791)
at callViewAction (core.js:14136)
at execEmbeddedViewsAction (core.js:14094)
at checkAndUpdateView (core.js:13786)
```
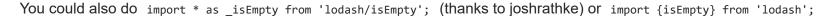
angular   lodash

asked Feb 16 '18 at 23:10

1 Answer

First you need to install the packages [lodash](#) and [@types/lodash](#) (contains type definitions):

16

```
npm i lodash
npm i --save-dev @types/lodash
```

Then you can use lodash e.g. with `import * as _ from 'lodash';` and further do `_.indexOf([1, 2, 1, 2], 2);`

You could also do `import * as _isEmpty from 'lodash/isEmpty';` (thanks to joshrathke) or `import {isEmpty} from 'lodash';`

edited Feb 16 '18 at 23:25   answered Feb 16 '18 at 23:16

pzaenger
3,712   1   20   29

---

5   For what its worth directly importing only the modules you need can make your build significantly smaller. Signifigantly, relative to the Lodash library :)
`import * as _indexOf from 'lodash/indexOf';` – joshrathke Feb 16 '18 at 23:22 ✎

---

1   @joshrathke That's a good point. I will edit my answer to add this point. – pzaenger Feb 16 '18 at 23:23

Worked perfectly. Thank you. –   koque   Feb 16 '18 at 23:39

@pzaenger `import * as _isEmpty from 'lodash/isEmpty';` does not work. Says `This module can only be referenced with ECMAScript...` –
Rich May 2 at 2:40 ✎