

Is it possible to export Arrow functions in ES6/7?

Asked 3 years, 6 months ago Active 7 days ago Viewed 36k times



The export statement below gives a syntax error

73

```
export default const hello = () => console.log("say hello")
```



why ?



I'm only able to export named functions

6

```
export function hello() {  
  console.log("hello")  
}
```

What is the reason?

javascript

ecmascript-6

arrow-functions

edited Apr 6 '16 at 18:07

asked Apr 6 '16 at 17:52



jozzy

1,215

1

11

11

4 What does the error actually say? – Andy Apr 6 '16 at 17:58

2 Works fine here: astexplorer.net/#/0fv5UXttsP . – Felix Kling Apr 6 '16 at 18:00

Read here: developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/... – John Apr 6 '16 at 18:03

1 what is the syntax error? – JordanHendrix Apr 6 '16 at 18:15

1 You cannot name a default export. – Felix Kling Apr 6 '16 at 18:25

2 Answers



Is it possible to export Arrow functions in ES6/7?

118

Yes. `export` doesn't care about the value you want to export.



The export statement below gives a syntax error ... why?



You cannot have a **default** export *and* give it a **name** ("default" is already the name of the export).

Either do

```
export default () => console.log("say hello");
```

or

```
const hello = () => console.log("say hello");  
export default hello;
```

edited Apr 6 '16 at 18:29

answered Apr 6 '16 at 18:27



Felix Kling

598k

138

913

976

3 How does the below work then `export default hello = () => { console.log("why the downvote")}` – [jozzy](#) Apr 6 '16 at 18:29

3 `x = y` is an assignment expression which resolves to the value of `y`. It's not a variable declaration. You can put `x = y` anywhere you can put an expression. **Note:** This will throw in strict mode if `x` is not defined beforehand. – [Felix Kling](#) Apr 6 '16 at 18:30 ✎

2 what about regular exports ? is it possible to do something like: `export () => { /*body*/ }` as `getUsers`; ? or do I have to first define it and then export it? – [Tomasz Mularczyk](#) Mar 24 '17 at 11:16 ✎

36 @Tomasz: `export const getUser = () => {...}`; – [Felix Kling](#) Mar 24 '17 at 13:00 ✎

3 @Burrich: It shouldn't matter how a function was created as long as it can be used in the way it is supposed to be. – [Felix Kling](#) Oct 17 '17 at 16:50



If you don't want a default export, you can simply export a named function with this syntax:

0

```
export const yourFunctionName = () => console.log("say hello");
```

answered Oct 24 at 6:34

[Raphael Pinel](#)

51 1 4