

# What is the “as syntax” pointed out by tslint?

Asked 2 years, 3 months ago   Active 6 days ago   Viewed 8k times



I upgraded tslint and now it complains about:

12

ERROR: src/**Metronome**/**JobFetcher**.ts[13, 32]: **Type** assertion using the '<>' syntax is forbidden. **Use** the 'as' syntax instead.



The offending code looks like:

1

```
const jobs = <JobConfig[]> <any> await rp(fetchJobsOptions);
```

What is the as syntax though and why should I use it?

typescript

tslint

asked May 26 '17 at 13:01



k0pernikus

21.4k

27

124

214

## 2 Answers



Refactor your code like this:

30

```
const jobs = await rp(fetchJobsOptions) as JobConfig[];
```



As pointed out in the [TypeScript Deep Dive book by Basarat Ali Syed](#), it says about type casting:

-----

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

Originally the syntax that was added was `<foo>` . This is demonstrated below:

```
var foo: any;
var bar = <string> foo; // bar is now of type "string"
```

However there is an ambiguity in the language grammar when using

```
<foo> style assertions in JSX:
var foo = <string>bar;
</string>
```

Therefore it is now recommended that you just use `as` for consistency.

## Type Assertion vs. Casting

The reason why it's not called "type casting" is that casting generally implies some sort of runtime support. However type assertions are purely a compile time construct and a way for you to provide hints to the compiler on how you want your code to be analyzed.

edited May 26 '17 at 13:20



crashmstr

24.5k 7 54 75

answered May 26 '17 at 13:01



k0pernikus

21.4k 27 124 214

If you want to suppress the error, you can as well go to `tslint.json` and include

0

```
...
"rules": {
  "no-angle-bracket-type-assertion": false,
  ...
}
...
```

provided you don't mind consistence as said.

edited Aug 27 at 9:08

answered Jul 4 at 14:54

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).