

How to use DecimalPipe from component in angular?

Asked 9 months ago Active 9 months ago Viewed 2k times



3

I have a requirement that I have to convert the number using the decimal pipe from ts

Instead of use decimal pipe like this



```
<td>{{rmanFmvRulesDef.max | number : '1.2-2'}}</td>
```



1

I want to manipulate it from the component, can anyone please help me ?

Thanks



angular



pipe



angular7



angular-pipe

edited Dec 17 '18 at 7:45



Mukyuu

2,305

3

14

29

asked Dec 17 '18 at 7:25



Soumya Gangamwar

469

4

27

2 Answers



9

As usual in angular you can rely on DI. You can override the transform function in the ts.



```
import { DecimalPipe } from '@angular/common';
class MyService {
  constructor(private _decimalPipe: DecimalPipe) {
  }
  transformDecimal(num) {
    return this._decimalPipe.transform(num, '1-2.2');
  }
}
```

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

```
providers: [DecimalPipe,...]
```

answered Dec 17 '18 at 8:56



parag badala

116 2

Adding the pipe to providers instead of declarations was one imported missing information. Thanks for this! – liqSTAR Aug 22 at 10:37

▲ You could see example on [Stackblitz](#)

1

Just simply call by: `element.price = format.formatCurrency(element.price);`

Once you've define the helper. You could define by: `<h2 *ngFor="let item of item">{{ item.price }}</h2>` as example.

Component:

```
import { accounting } from 'accounting';

export function formatCurrency(value) {
  return accounting.formatMoney(value, '', 0, ' ', '.');
}

export function unformatCurrency(value) {
  return accounting.unformat(value);
}
```

Note: `return accounting.formatMoney(value, '', 0, ' ', '.');` You can change ' ' and . with whatever splitter that you want , or space. For how many number behind decimal you could update the 0 number to how many digits you want.

answered Dec 17 '18 at 7:32



Mukyuu

2,305 3 14 29

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