Notifications Powered By Feedify

Here to help crazed fans go right.

Angular 7 Pipes

In Angular 1, filters are used which are later called Pipes onwards Angular2. In Angular 7, it is known as pipe and used to transform data. It is denoted by symbol |

Syntax:

```
{{title | uppercase}}
```

Pipe takes integers, strings, arrays, and date as input separated with |. It transforms the data in the format as required and displays the same in the browser.

Let's see an example using pipes. Here, we display the title text in upper and lower case by using pipes.

Example:

Define a variable named "title" in component.ts file.

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'my-first-app';
}
```

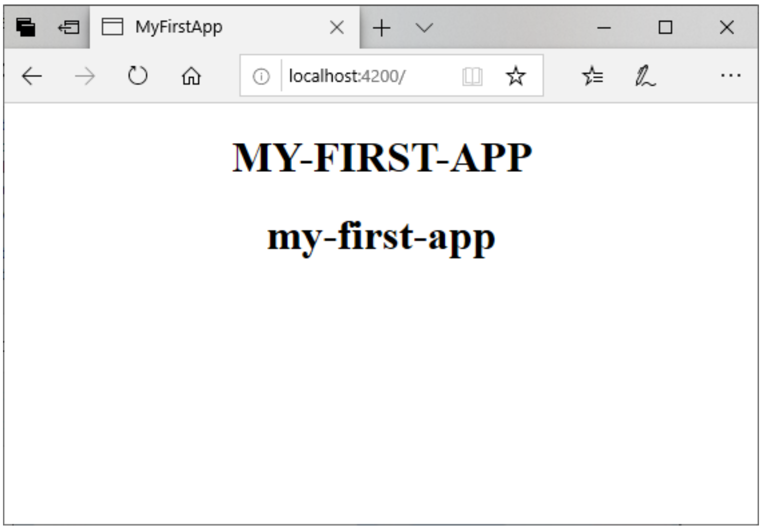
Use the pipe symbol in component.html file:

```
<h1>
   {{ title | uppercase }} <br/></h1>
<h1>
  {{ title | lowercase }} <br/></h1>
```

Output:

Run ng serve and see the result. You will see the following result.

Notifications Powered By Feedify



MY-FIRST-APP

my-first-app

Here, you can see that pipes have changed the tittle in upper and lowercase.

Angular 7 Built-in Pipes

Angular 7 provides some built-in pipes:

- Lowercasepipe
- Uppercasepipe
- Datepipe
- Currencypipe
- Jsonpipe
- Percentpipe
- Decimalpipe
- Slicepipe

You have seen the lowercasepipe and uppercasepipe examples. Now, let's take some examples to see how the other pipes work.

Example:

Define the required variables in component.ts file.

component.ts file:

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'my-first-app';
  todaydate = new Date();
  jsonval = {name: 'Alex', age: '25', address:{a1: 'Paris', a2: 'France'}};
  months = ['Jan', 'Feb', 'Mar', 'April', 'May', 'Jun',
```

Notifications Powered By Feedify

```
  'July', 'Aug', 'Sept', 'Oct', 'Nov', 'Dec'];
}
```

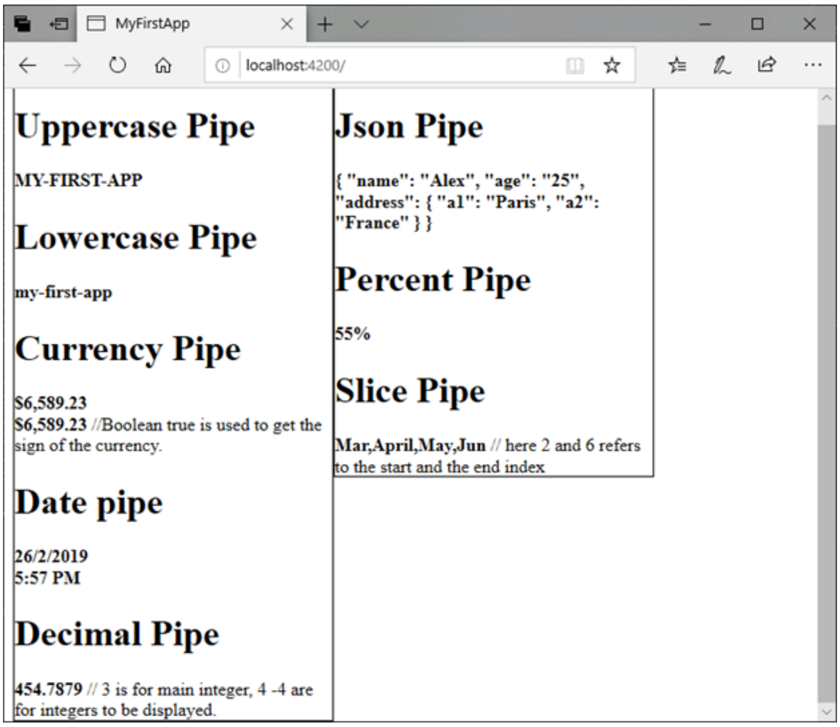
Use the different built-in pipe symbols in component.html file:

component.html file:

```
<div style = "width:100%;">
  <div style = "width:40%;float:left;border:solid 1px black;">
    <h1>Uppercase Pipe</h1>
    <b>{{title | uppercase}}</b><br/>
    <h1>Lowercase Pipe</h1>
    <b>{{title | lowercase}}</b>
    <h1>Currency Pipe</h1>
    <b>{{6589.23 | currency:"USD"}}</b><br/>
    <b>{{6589.23 | currency:"USD":true}}</b> //Boolean true is used to get the sign of the currency.
    <h1>Date pipe</h1>
    <b>{{todaydate | date:'d/M/y'}}</b><br/>
    <b>{{todaydate | date:'shortTime'}}</b>
    <h1>Decimal Pipe</h1>
    <b>{{ 454.78787814 | number: '3.4-4' }}
</b> // 3 is for main integer, 4 -4 are for integers to be displayed.
  </div>
  <div style = "width:40%;float:left;border:solid 1px black;">
    <h1>Json Pipe</h1>
    <b>{{ jsonval | json }}</b>
    <h1>Percent Pipe</h1>
    <b>{{00.54565 | percent}}</b>
    <h1>Slice Pipe</h1>
    <b>{{months | slice:2:6}}</b>
    // here 2 and 6 refers to the start and the end index
  </div>
</div>
```

Output:

You can see the use of all built-in pipes here:



Uppercase Pipe

MY-FIRST-APP

Lowercase Pipe

my-first-app

Currency Pipe

$6,589.23
$6,589.23 //Boolean true is used to get the sign of the currency.

Date pipe

26/2/2019
5:57 PM

Decimal Pipe

454.7879 // 3 is for main integer, 4 -4 are for integers to be displayed.

Json Pipe

{ "name": "Alex", "age": "25", "address": { "a1": "Paris", "a2": "France" } }

Percent Pipe

55%

Slice Pipe

Mar,April,May,Jun // here 2 and 6 refers to the start and the end index

How to create a custom pipe?

To create a custom pipe, create a new ts file and use the code according to the work you have to do. You have to import Pipe, PipeTransform from Angular/Core. Let's create a sqrt custom pipe.

Notifications Powered By Feedify

component.ts file:

```
import {Pipe, PipeTransform} from '@angular/core';
@Pipe ({
  name : 'sqrt'
})
export class SqrtPipe implements PipeTransform {
  transform(val : number) : number {
    return Math.sqrt(val);
  }
}
```

Now, it's turn to make changes in the app.module.ts. Create a class named as SqrtPipe. This class will implement the PipeTransform. The transform method defined in the class will take argument as the number and will return the number after taking the square root.

As, we have created a new file so, we need to add the same in app.module.ts.

Module.ts file:

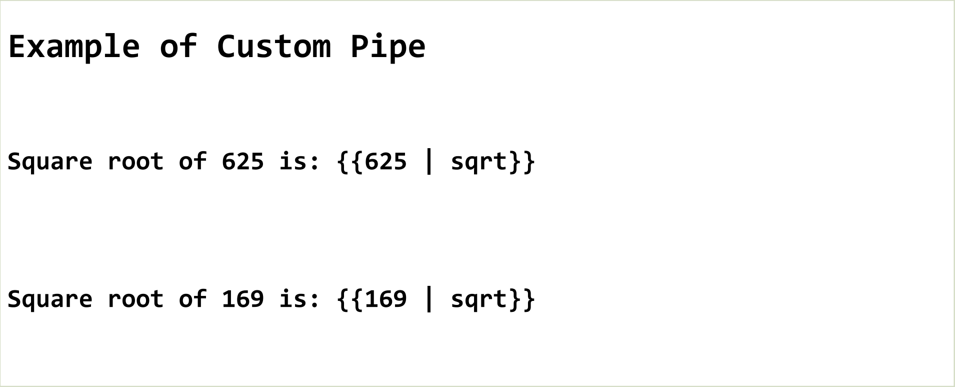
```
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { AppComponent } from './app.component';
import { NewCmpComponent } from './new-cmp/new-cmp.component';
import { ChangeTextDirective } from './change-text.directive';
import { SqrtPipe } from './app.sqrt';
@NgModule({
   declarations: [
      SqrtPipe,
      AppComponent,
      NewCmpComponent,
      ChangeTextDirective
   ],
   imports: [
      BrowserModule
   ],
   providers: [],
   bootstrap: [AppComponent]
})
export class AppModule { }
```

Now, use sqrt pipe in component.html file.

component.html file:

```
<h1>Example of Custom Pipe</h1>
<h2>Square root of 625 is: {{625 | sqrt}}</h2><br/>
<h2>Square root of 169 is: {{169 | sqrt}}</h2>
```

Output:



```
Example of Custom Pipe

Square root of 625 is: {{625 | sqrt}}

Square root of 169 is: {{169 | sqrt}}
```

← prev next →

Notifications Powered By Feedify

It’s better to bundle

Ad Changing addresses? Bundle home & auto with Alls

Allstate

Quote Now

Please Share

Learn Latest Tutorials

Agile	QA	Angular 7	D. Science	ReactJS
Software E.				

Preparation

Aptitude	Reasoning	Verbal A.	Interview	Company

Trending Technologies

AI	AWS	Selenium	IoT	Cloud
Hadoop				

B.Tech / MCA

DBMS	DS	DAA	OS	C. Network
Compiler D.	COA	D. Math.	E. Hacking	C. Graphics

⇧

Notifications Powered By Feedify

Web Tech.	Cyber Sec.	Automata	C	C++
Java	.Net	Python	Programs	Control S.

⇧