**Pipes in Angular:**

**Reference URLs:**

<https://angular.io/guide/pipes>

<https://www.tektutorialshub.com/angular/angular-custom-pipes/>

Use [pipes](https://angular.io/guide/glossary#pipe) to transform strings, currency amounts, dates, and other data for display. Pipes are simple functions to use in [template expressions](https://angular.io/guide/glossary#template-expression) to accept an input value and return a transformed value. Pipes are useful because you can use them throughout your application, while only declaring each pipe once. For example, you would use a pipe to show a date as **April 15, 1988** rather than the raw string format.

Angular provides built-in pipes for typical data transformations, including transformations for internationalization (i18n), which use locale information to format data. The following are commonly used built-in pipes for data formatting:

* [DatePipe](https://angular.io/api/common/DatePipe): Formats a date value according to locale rules.
* [UpperCasePipe](https://angular.io/api/common/UpperCasePipe): Transforms text to all upper case.
* [LowerCasePipe](https://angular.io/api/common/LowerCasePipe): Transforms text to all lower case.
* [CurrencyPipe](https://angular.io/api/common/CurrencyPipe): Transforms a number to a currency string, formatted according to locale rules.
* [DecimalPipe](https://angular.io/api/common/DecimalPipe): Transforms a number into a string with a decimal point, formatted according to locale rules.
* [PercentPipe](https://angular.io/api/common/PercentPipe): Transforms a number to a percentage string, formatted according to locale rules.

**Pure and Impure Pipes:**

**A pure pipe** is only called when Angular detects a change in the value or the parameters passed to a pipe.

**An impure pipe** is called for every change detection cycle no matter whether the value or parameter(s) changes.

How to Create Custom Pipe in Angular

The [Pipes](https://www.tektutorialshub.com/angular/angular-pipes/) are a great way to transform the appearance of elements in the template. The [Angular](https://www.tektutorialshub.com/angular-tutorial/)comes with some great built-in pipes like Date pipe, Currency pipe, and Number pipe, etc. But if these pipes do not cover your needs, then we can create our own [pipe in Angular](https://www.tektutorialshub.com/angular/angular-pipes/).

To create a custom pipe, first we need to create a pipe class. The pipe class must implement the PipeTransform interface. We also decorate it with @pipe decorator. Give a name to the pipe under name metadata of the @pipe decorator. Finally, we create the transform method, which transforms given value to the desired output.

## How to Create Custom Pipes

To create a Custom Pipe, first, You need to follow these steps

1. Create a pipe class
2. Decorate the class with @pipe decorator.
3. Give a name to the pipe in the name meta data of the @pipe decorator. We will use this name in the template.
4. The pipe class must implement the PipeTransform interface. The interfaces contain only one method transform.
5. The first parameter to the transform method is the value to be transferred. The transform method must transform the value and return the result. You can add any number of additional arguments to the transform method.
6. Declare the pipe class in the Angular Module (app.module.ts)
7. Use the custom pipe just as you use other pipes.