***Angular Pipes***

***============***

***- Angular have to present data that comes from various data sources.***

***- The datatype supported by angular is not similar as the data source provided to application.***

***- The data will not be presented in the same format that we are expecting.***

***- You can transform the data into various formats by using "Angular Pipes".***

***"Transforming Data using Angular Pipes"***

***- Pipe is a TypeScript class which implements "PipeTransform" base class.***

***- Pipe provide a "transform()" method that transforms your data.***

***- Pipes are used to transforms and format strings, currency, dates and other display data.***

***- Pipes are uses on template expressions by using "|"***

***{{ yourData | pipeName | pipeName }}***

***- Based on when a pipe is applied to data, Pipes are categorized into 2 groups***

***a) Pure Pipe***

***b) Impure Pipe***

***- Pure pipe is the pipe which applies the transformation when any change detected in value or by passing a parameter into pipe.***

***- Impure pipe is the pipe that applies the transformation on every change instead of verifying the parameter.***

***- Angular provides several built-in Pipes and also allows to create custom pipes***

***"Angular Built - in Pipes"***

***- AsyncPipe***

***- CurrencyPipe***

***- DatePipe***

***- DecimalPipe***

***- I18nPluralPipe***

***- I18nSelectPipe***

***- JsonPipe***

***- KeyValuePipe***

***- LowerCasePipe***

***- UpperCasePipe***

***- TitleCasePipe***

***- PercentPipe***

***- SlicePipe***

***1. UpperCasePipe : It converts all letters into uppercase.***

***2. LowerCasePipe : It converts all letters into lowercase.***

***3. TitleCasePipe : It converts every word first letter into***

***uppercase.***

***Syntax:***

***{{ yourData | uppercase }}***

***{{ yourData | lowercase }}***

***{{ yourData | titlecase }}***

***4. DecimalPipe : It is used to display numeric value with***

***thousands separator and fractions.***

***Decimal Pipe reference name is "number".***

***Syntax:***

***{{ data | number:{minIntegerDigits}.{minFractionDigits}-{maxFractionDigits}***

***EX:***

***public price = 45000.40;***

***{{ price }} // 45000.4***

***{{ price | number }} // 45,000.4***

***{{pirce | number: '5.2-2' }} // 45,000.40***

***- lowercase***

***- uppercase***

***- titlecase***

***- decimal [number]***

***5. CurrencyPipe: It is used to display numbers in a currency format, with fractions and currency symbol.***

***Syntax:***

***{{ data | currency: 'Format': 'digitsInfo' }}***

***Format = USD, INR, you can use HTML entities***

***[&#8377;]***

***DigitsInfo: {minInteger}.{minFraction}-{maxFraction}***

***6. DatePipe : It is used to display date value in various date***

***formats.***

***Predefined Formats:***

***- short***

***- medium***

***- long***

***- full***

***- shortDate***

***- mediumDate***

***- longDate***

***- fullDate***

***- shortTime***

***- mediumTime***

***- longTime***

***- fullTime***

***Ex:***

***{{ product.Mfd | date : 'fullDate' }}***

***Custom Date Format is built by using***

***MM - Month Number***

***MMM - Short Month Name***

***MMMM - Full Month Name***

***yy - Year 2 Digits***

***yyyy - Year 4 Digits***

***d - One digit date***

***dd - two digits date***

***Ex:***

***{{product.Mfd | date:'dd-MMMM-yyyy'}}***

***7. PercentPipe: It transforms a number into a percentage string.***

***{{ data | percent: 'digitsInfo' }}***

***Ex:***

***Sales : 0.259***

***{{ product.Sales | percent: '2.2-4' }}***

***25.90%***

***8. SlicePipe***

***It creates a new Array or string containing a subset (slice) of the elements.***

***Syntax:***

***{{ collection | slice:startIndex:endIndex }}***

***Ex:***

***public products = ['TV', 'Mobile', 'Shoe', 'Shirt', 'Jeans'];***

***<ol>***

***<li \*ngFor="let item of products | slice:1:3">***

***{{item}}***

***</li>***

***</ol>***

***O/P:***

***1. Mobile***

***2. Shoe***

***9. JsonPipe : It is used to convert the data into JSON format***

***So that it can be transported accross requests.***

***Syntax:***

***{{ data | json }}***

***Ex:***

***public product = {***

***Name: 'Samsung TV',***

***Price: 45000.40,***

***Mfd: new Date('2020/04/20'),***

***Sales: 0.259***

***};***

***<div>***

***<pre>***

***{{product | json}}***

***</pre>***

***</div>***

***10. KeyValuePipe: It is used to transform an object or a map into an array of key and value pairs.***

***It provides the properties "key" & "value", which return keys and values form a collection.***

***Syntax:***

***{{ iterator | keyvalue }}***

***item.key item.value***

***Ex:***

***1. pipedemo.component.ts***

***export class PipedemoComponent***

***{***

***public products = ['TV', 'Mobile', 'Shoe', 'Shirt', 'Jeans'];***

***public product: {[key: number]: string} = {1: 'Samsung TV', 2: 'Mobile'};***

***}***

***2. pipedemo.component.html***

***<div class="container">***

***<div>***

***<ul class="list-unstyled">***

***<li \*ngFor="let item of products | keyvalue">***

***{{item.key}} : {{item.value}}***

***</li>***

***</ul>***

***</div>***

***<div \*ngFor="let item of product | keyvalue">***

***{{item.key}} : {{item.value}}***

***</div>***

***</div>***

***11. I18nSelectPipe***

***It is a Generic selector that can make decision dynamically according to state or value and defines the result when the relative condition matches.***

***{{ data | i18Select: messageObject }}***

***EX:***

***1. pipedemo.component.ts***

***export class PipedemoComponent{***

***public products = [***

***{Name: 'Samsung TV', Price: 34000.33, ShippedTo: 'Delhi'},***

***{Name: 'Nike Casuals', Price: 1300.44, ShippedTo: 'Hyderabad'},***

***{Name: 'Shirt', Price: 2300.44, ShippedTo: 'Mumbai'}***

***];***

***public statusMessage = {***

***'Hyderabad': 'Delivery in 2 Days',***

***'Delhi': 'Delivery in 5 Days',***

***'Mumbai': 'Not Deliverable'***

***};***

***}***

***2. pipedemo.component.html***

***<div class="container">***

***<table class="table table-hover">***

***<thead>***

***<tr>***

***<th>Name</th>***

***<th>Price</th>***

***<th>City</th>***

***<th>Delivery Time</th>***

***</tr>***

***</thead>***

***<tbody>***

***<tr \*ngFor="let item of products">***

***<td>{{item.Name}}</td>***

***<td>{{item.Price}}</td>***

***<td>{{item.ShippedTo}}</td>***

***<td>{{item.ShippedTo | i18nSelect:statusMessage}}</td>***

***</tr>***

***</tbody>***

***</table>***

***</div>***