Building Custom Pipes:

1. Add product-filter.pipe.ts file under products folder. Include the PipeTranform implementation for custom pipe.

product-filter.pipe.ts

```
import { PipeTransform, Pipe } from 'angular2/core';
import { IProduct } from './product';

@Pipe({
    name: 'productFilter'
})
export class ProductFilterPipe implements PipeTransform {

    transform(value: IProduct[], args: string[]): IProduct[] {
        let filter: string = args[0] ? args[0].toLocaleLowerCase() : null;
        return filter ? value.filter((product: IProduct) =>
            product.productName.toLocaleLowerCase().indexOf(filter) != -1) :
value;
    }
}
```

2. Once PipeTrasform implementation is ready add this in you product-list.component.ts file as mentioned below:

product-list.component.ts

```
import { Component, OnInit } from 'angular2/core';
import { IProduct } from './product';
import { ProductFilterPipe } from './product-filter.pipe';
@Component({
    selector: 'pm-products',
    templateUrl: 'app/products/product-list.component.html',
    styleUrls: ['app/products/product-list.component.css'],
    pipes: [ProductFilterPipe]

})
export class ProductListComponent implements OnInit{
    pageTitle: string = 'Product List';
    imageWidth: number = 50;
    imageMargin: number = 2;
    showImage: boolean = false;
    listFilter: string = 'cart';
    products: IProduct[] = [
        {
            "productId": 2,
            "productName": "Garden Cart",
            "productCode": "GDN-0023",
            "releaseDate": "March 18, 2016",
            "description": "15 gallon capacity rolling garden cart",
            "price": 32.99,
            "starRating": 4.2,
            "imageUrl": "https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcRZhmstt7Kkdk5kyMv9mxJyh9sPUBvY9poTrxcl-
SCAUa3WmUVZQA"
        },
        {
            "productId": 5,
            "productName": "Hammer",
            "productCode": "TBX-0048",
            "releaseDate": "May 21, 2016",
            "description": "Curved claw steel hammer",
            "price": 8.9,
            "starRating": 4.8,
            "imageUrl": "https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcSlXJdOiLq-d9v2-
UDfTlQFF3IuhdNMoE1ok1pAgwqubZaZqQJ9"
        }
    ];

    toggleImage(): void{
        this.showImage = !this.showImage;
    }

    ngOnInit(): void{
        console.log('Init Invoked!')
    }

}
```

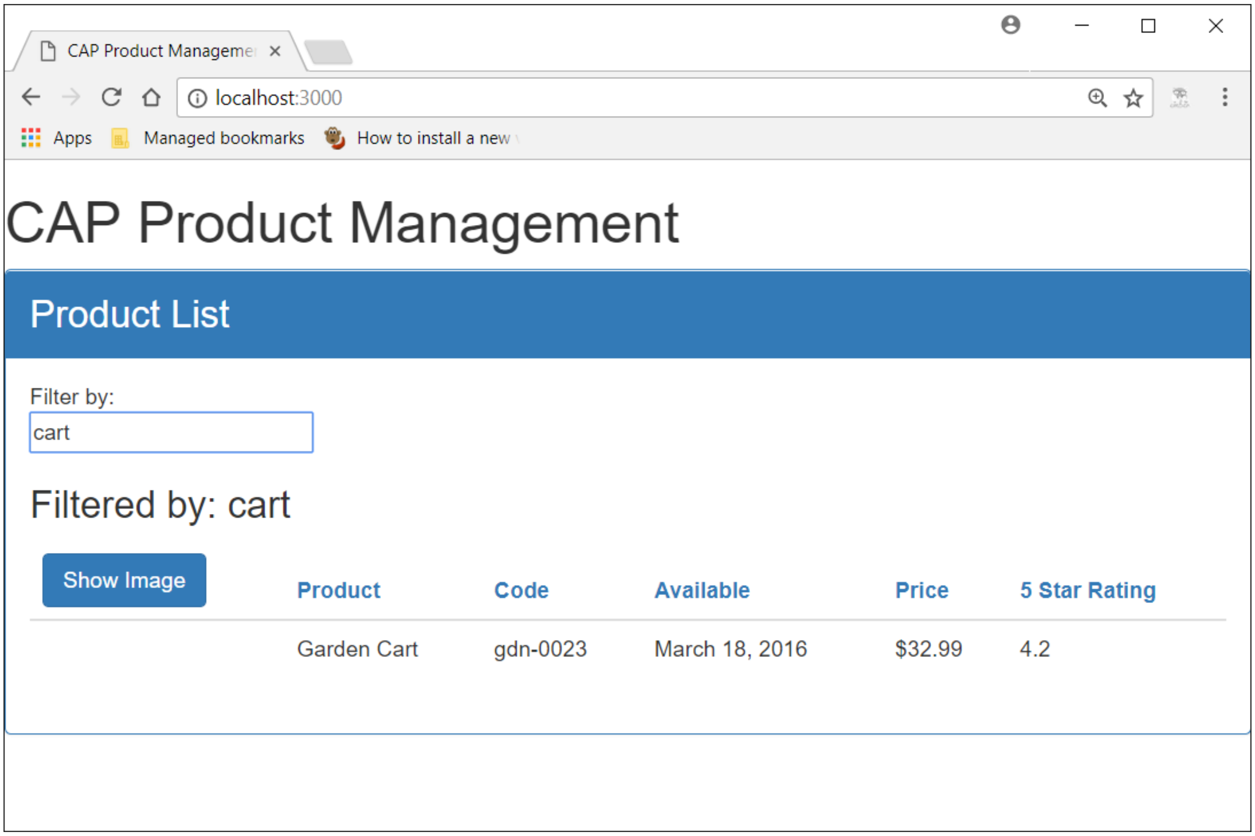
3. Use the custom pipe which we have created in product-list.component.html file along with ngFor directives to filter products.

product-list.component.html

```
<div class='panel panel-primary'>
    <div class='panel-heading'>
        {{pageTitle}}
    </div>
    <div class='panel-body'>
        <div class='row'>
            <div class='col-md-2'>Filter by:</div>
            <div class='col-md-4'>
                <input type='text' [(ngModel)]='listFilter'/>
            </div>
        </div>
        <div class='row'>
            <div class='col-md-6'>
                <h3>Filtered by: {{listFilter}} </h3>
            </div>
        </div>
        <div class='table-responsive'>
            <table class='table' *ngIf='products && products.length'>
                <thead>
                    <tr>
                        <th>
                            <button class='btn btn-primary'
                             (click) = 'toggleImage()'>
                                {{showImage ? 'Hide' : 'Show'}} Image
                            </button>
                        </th>
                        <th>Product</th>
                        <th>Code</th>
                        <th>Available</th>
                        <th>Price</th>
                        <th>5 Star Rating</th>
                    </tr>
                </thead>
                <tbody>
                    <tr *ngFor='#product of products | productFilter:listFilter'>
                        <td>
                            <img *ngIf='showImage'
                            [src]='product.imageUrl'
                            [title]='product.productName'
                            [style.width.px]='imageWidth'
                            [style.margin.px]='imageMargin'>
                        </td>
                        <td>{{product.productName}}</td>
                        <td>{{product.productCode | lowercase }}</td>
                        <td>{{product.releaseDate}}</td>
                        <td>{{product.price | currency:'USD':true:'1.2-2'}}</td>
                        <td>{{product.starRating}}</td>
```

```
                   </tr>
                </tbody>
            </table>
        </div>
    </div>
</div>
```

Now you can check your browser



CAP Product Management

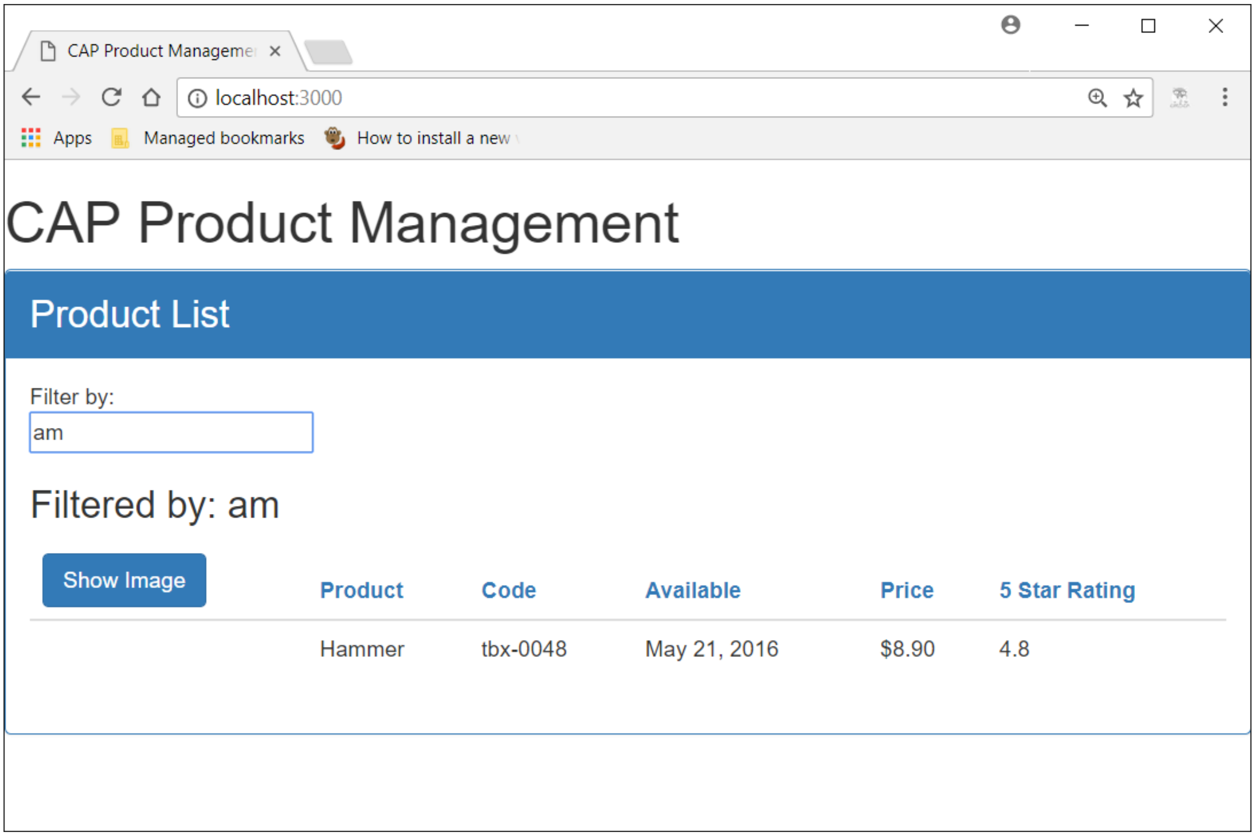
Product List

Filter by:

cart

Filtered by: cart

Product	Code	Available	Price	5 Star Rating
Garden Cart	gdn-0023	March 18, 2016	$32.99	4.2



CAP Product Management

Product List

Filter by:

am

Filtered by: am

Product	Code	Available	Price	5 Star Rating
Hammer	tbx-0048	May 21, 2016	$8.90	4.8

Building Nested Component

1. Add star.component.ts file under shared folder under app/shared directory. And include the below content.

   star.component.ts

```
import { Component, OnChanges } from 'angular2/core';

@Component({
    selector: 'ai-star',
    templateUrl: 'app/shared/star.component.html',
    styleUrls: ['app/shared/star.component.css']
})
export class StarComponent implements OnChanges {

    starWidth: number;
    rating: number = 4;

    ngOnChanges(): void {
    this.starWidth = this.rating * 86 / 5;
    }
}
```

2. Add below html content in your star.component.css file under shared folder.

```
.crop {
    overflow: hidden;
}
div {
    cursor: pointer;
}
```

3. Add below html content in your star.component.html file under shared folder.

star.component.html

```
<div class="crop"
    [style.width.px]="starWidth"
    [title]="rating">
    <div style="width: 86px">
        <span class="glyphicon glyphicon-star"></span>
        <span class="glyphicon glyphicon-star"></span>
        <span class="glyphicon glyphicon-star"></span>
        <span class="glyphicon glyphicon-star"></span>
        <span class="glyphicon glyphicon-star"></span>
    </div>
</div>
```

4. Do the below changes in your product-list.component.ts file

```
import { Component, OnInit } from 'angular2/core';
import { IProduct } from './product';
import { ProductFilterPipe } from './product-filter.pipe';
import { StarComponent } from '../shared/star.component';
@Component({
    selector: 'pm-products',
    templateUrl: 'app/products/product-list.component.html',
    styleUrls: ['app/products/product-list.component.css'],
    pipes: [ProductFilterPipe],
    directives: [StarComponent]

})
```

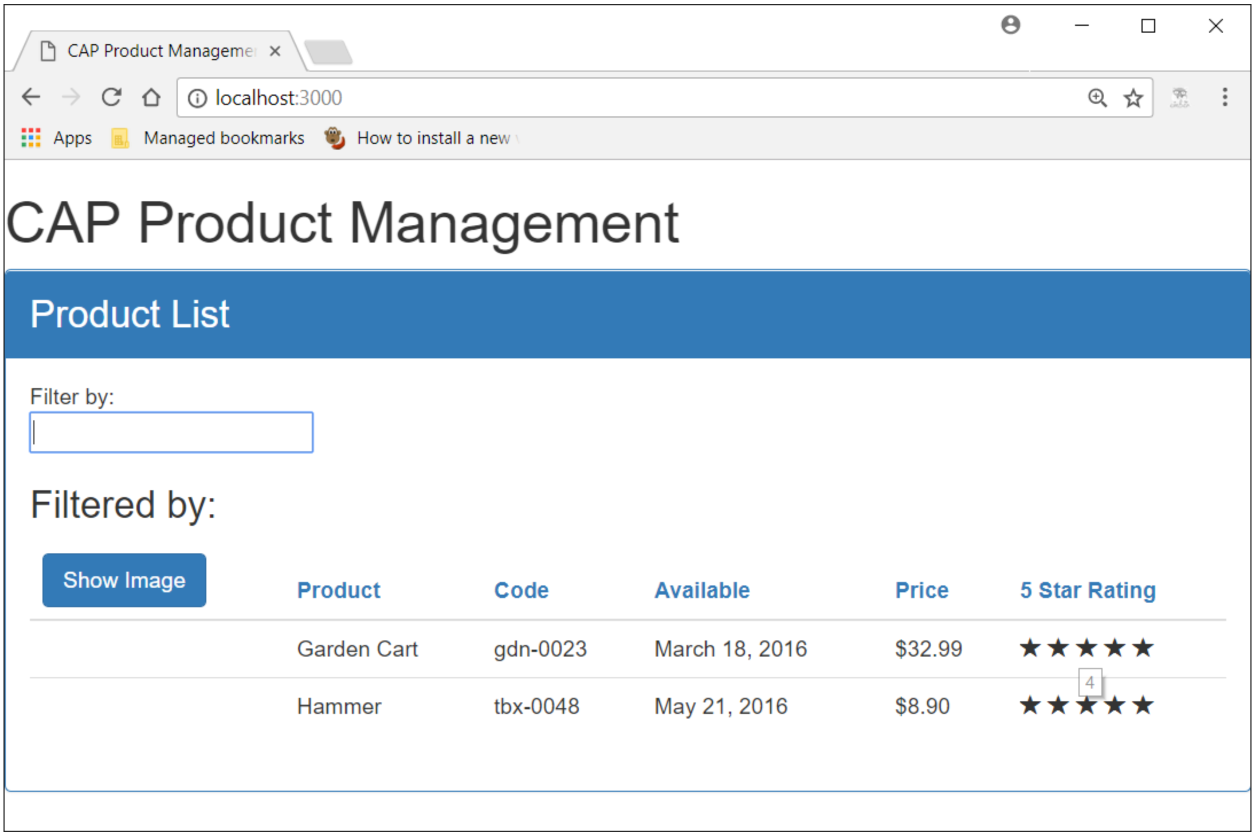
```
export class ProductListComponent implements OnInit{
    pageTitle: string = 'Product List';
    imageWidth: number = 50;
    imageMargin: number = 2;
    showImage: boolean = false;
    listFilter: string = 'cart';
    products: IProduct[] = [
        {
            "productId": 2,
            "productName": "Garden Cart",
            "productCode": "GDN-0023",
            "releaseDate": "March 18, 2016",
            "description": "15 gallon capacity rolling garden cart",
            "price": 32.99,
            "starRating": 4.2,
            "imageUrl": "https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcRZhmstt7Kkdk5kyMv9mxJyh9sPUBvY9poTrxcl-
SCAUa3WmUVZQA"
        },
        {
            "productId": 5,
            "productName": "Hammer",
            "productCode": "TBX-0048",
            "releaseDate": "May 21, 2016",
            "description": "Curved claw steel hammer",
            "price": 8.9,
            "starRating": 4.8,
            "imageUrl": "https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcSlXJdOiLq-d9v2-
UDfTlQFF3IuhdNMoE1ok1pAgwqubZaZqQJ9"
        }
    ];

    toggleImage(): void{
        this.showImage = !this.showImage;
    }

    ngOnInit(): void{
        console.log('Init Invoked!')
    }

}
```

5. Include <ai-star> tag in your product-list.component.html file.

product-list.component.html

```
<div class='panel panel-primary'>
    <div class='panel-heading'>
        {{pageTitle}}
    </div>
    <div class='panel-body'>
        <div class='row'>
            <div class='col-md-2'>Filter by:</div>
            <div class='col-md-4'>
                <input type='text' [(ngModel)]='listFilter'/>
            </div>
        </div>
        <div class='row'>
            <div class='col-md-6'>
                <h3>Filtered by: {{listFilter}} </h3>
            </div>
        </div>
        <div class='table-responsive'>
            <table class='table' *ngIf='products && products.length'>
                <thead>
                    <tr>
                        <th>
                            <button class='btn btn-primary'
                             (click) = 'toggleImage()'>
                                {{showImage ? 'Hide' : 'Show'}} Image
                            </button>
                        </th>
                        <th>Product</th>
                        <th>Code</th>
                        <th>Available</th>
                        <th>Price</th>
                        <th>5 Star Rating</th>
                    </tr>
                </thead>
                <tbody>
                   <tr *ngFor='#product of products | productFilter:listFilter'>
                        <td>
                            <img *ngIf='showImage'
                            [src]='product.imageUrl'
                            [title]='product.productName'
                            [style.width.px]='imageWidth'
                            [style.margin.px]='imageMargin'>
                        </td>
                        <td>{{product.productName}}</td>
                        <td>{{product.productCode | lowercase }}</td>
                        <td>{{product.releaseDate}}</td>
                        <td>{{product.price | currency:'USD':true:'1.2-2'}}</td>
                        <td>
                            <ai-star></ai-star>
                        </td>
                    </tr>
                 </tbody>
             </table>
         </div>
     </div>
</div>
```

You can see star in your browser in place of ratings.



CAP Product Management

Product List

Filter by:

Filtered by:

Product	Code	Available	Price	5 Star Rating
Garden Cart	gdn-0023	March 18, 2016	$32.99	★★★★★
Hammer	tbx-0048	May 21, 2016	$8.90	★★★★★

Passing Data to the Nested Components:

6. Include @Input and @Output decorator in your star.component.ts file under shared folder.

star.component.ts

```
import { Component, OnChanges, Input,
        Output, EventEmitter } from 'angular2/core';

@Component({
    selector: 'ai-star',
    templateUrl: 'app/shared/star.component.html',
    styleUrls: ['app/shared/star.component.css']
})
export class StarComponent implements OnChanges {

    starWidth: number;
    @Input() rating: number ;
    @Output() ratingClicked: EventEmitter<string>
            = new EventEmitter<string>();


    ngOnChanges(): void {
    this.starWidth = this.rating * 86 / 5;
    }
    onClick() {
        this.ratingClicked.emit(`The rating ${this.rating} was clicked!`);
    }

}
```

7. Also modify star template as mentioned below.

star.component.html

```
<div class="crop"
    [style.width.px]="starWidth"
    [title]="rating"
    (click) = 'onClick()'>
    <div style="width: 86px">
        <span class="glyphicon glyphicon-star"></span>
        <span class="glyphicon glyphicon-star"></span>
        <span class="glyphicon glyphicon-star"></span>
        <span class="glyphicon glyphicon-star"></span>
        <span class="glyphicon glyphicon-star"></span>
    </div>
</div>
```

8. Add one more method in your product-list.component.ts file.

product-list.component.ts

```
import { Component, OnInit } from 'angular2/core';
import { IProduct } from './product';
import { ProductFilterPipe } from './product-filter.pipe';
import { StarComponent } from '../shared/star.component';
@Component({
    selector: 'pm-products',
    templateUrl: 'app/products/product-list.component.html',
    styleUrls: ['app/products/product-list.component.css'],
    pipes: [ProductFilterPipe],
    directives: [StarComponent]

})
export class ProductListComponent implements OnInit{
    pageTitle: string = 'Product List';
    imageWidth: number = 50;
    imageMargin: number = 2;
    showImage: boolean = false;
    listFilter: string = 'cart';
    products: IProduct[] = [
        {
            "productId": 2,
            "productName": "Garden Cart",
            "productCode": "GDN-0023",
            "releaseDate": "March 18, 2016",
            "description": "15 gallon capacity rolling garden cart",
            "price": 32.99,
            "starRating": 4.2,
            "imageUrl": "https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcRZhmstt7Kkdk5kyMv9mxJyh9sPUBvY9poTrxcl-
SCAUa3WmUVZQA"
        },
        {
            "productId": 5,
            "productName": "Hammer",
            "productCode": "TBX-0048",
            "releaseDate": "May 21, 2016",
            "description": "Curved claw steel hammer",
            "price": 8.9,
            "starRating": 4.8,
            "imageUrl": "https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcSlXJdOiLq-d9v2-
UDfTlQFF3IuhdNMoE1ok1pAgwqubZaZqQJ9"
        }
    ];

    toggleImage(): void{
        this.showImage = !this.showImage;
    }

    ngOnInit(): void{
        console.log('Init Invoked!')
    }

    onRatingClicked(message: string): void{
        this.pageTitle = 'Product List: ' + message;
    }

}
```

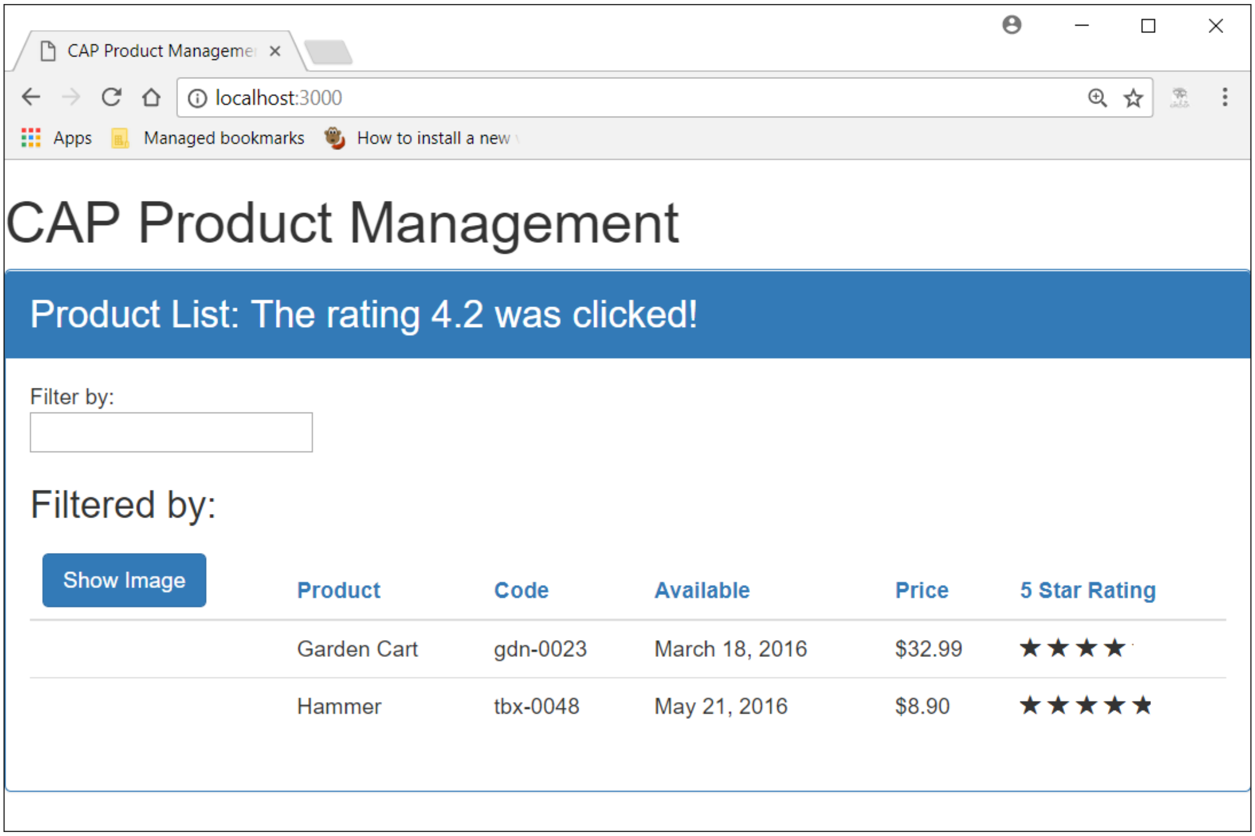
9. Now you can add event in your product-list.component.html file.

product-list.component.html

```
<div class='panel panel-primary'>
    <div class='panel-heading'>
        {{pageTitle}}
    </div>
    <div class='panel-body'>
        <div class='row'>
            <div class='col-md-2'>Filter by:</div>
            <div class='col-md-4'>
                <input type='text' [(ngModel)]='listFilter'/>
            </div>
        </div>
        <div class='row'>
            <div class='col-md-6'>
                <h3>Filtered by: {{listFilter}} </h3>
            </div>
        </div>
        <div class='table-responsive'>
            <table class='table' *ngIf='products && products.length'>
                <thead>
```

```
                    <tr>
                        <th>
                            <button class='btn btn-primary'
                             (click) = 'toggleImage()'>
                                {{showImage ? 'Hide' : 'Show'}} Image
                            </button>
                        </th>
                        <th>Product</th>
                        <th>Code</th>
                        <th>Available</th>
                        <th>Price</th>
                        <th>5 Star Rating</th>
                    </tr>
                </thead>
                <tbody>
                   <tr *ngFor='#product of products | productFilter:listFilter'>
                        <td>
                            <img *ngIf='showImage'
                            [src]='product.imageUrl'
                            [title]='product.productName'
                            [style.width.px]='imageWidth'
                            [style.margin.px]='imageMargin'>
                        </td>
                        <td>{{product.productName}}</td>
                        <td>{{product.productCode | lowercase }}</td>
                        <td>{{product.releaseDate}}</td>
                        <td>{{product.price | currency:'USD':true:'1.2-2'}}</td>
                        <td>

                            <ai-star [rating]='product.starRating'
                             (ratingClicked)='onRatingClicked($event)'></ai-star>
                        </td>
                   </tr>
                </tbody>
            </table>
        </div>
    </div>
</div>
```

Now you can see the below output in your browser with correct start rating and appropriate message when you click ratings.



CAP Product Management

Product List: The rating 4.2 was clicked!

Filter by:

Filtered by:

	Product	Code	Available	Price	5 Star Rating
	Garden Cart	gdn-0023	March 18, 2016	$32.99	★★★★
	Hammer	tbx-0048	May 21, 2016	$8.90	★★★★★

Services and Dependency Injections:

10. Add file called product.service.ts under products folder. And include the below content within it.

product.service.ts

```
import {Injectable} from 'angular2/core';
import {IProduct} from './product';

@Injectable()
export class ProductService{
    getProducts(): IProduct[]{
        return[
            {
                "productId": 2,
                "productName": "Garden Cart",
                "productCode": "GDN-0023",
                "releaseDate": "March 18, 2016",
                "description": "15 gallon capacity rolling garden cart",
                "price": 32.99,
                "starRating": 4.2,
                "imageUrl": "https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcRZhmstt7Kkdk5kyMv9mxJyh9sPUBvY9poTrxcl-
SCAUa3WmUVZQA"
            },
            {
                "productId": 5,
                "productName": "Hammer",
                "productCode": "TBX-0048",
                "releaseDate": "May 21, 2016",
                "description": "Curved claw steel hammer",
                "price": 8.9,
                "starRating": 4.8,
                "imageUrl": "https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcSlXJdOiLq-d9v2-
UDfTlQFF3IuhdNMoE1ok1pAgwqubZaZqQJ9"
            }
        ]
    }

}
```

11. Register this ProductService in your app.component.ts file.

app.component.ts

```
import { Component } from 'angular2/core';
import { ProductListComponent } from './products/product-list.component';
import {ProductService} from './products/product.service';

@Component({
    selector: 'pm-app',
    template: `
        <div>
            <h1>{{pageTitle}}</h1>
            <pm-products></pm-products>
        </div>
    `,
    directives: [ProductListComponent],
    providers: [ProductService]
})

export class AppComponent{
    pageTitle: string= 'CAP Product Management';

}
```

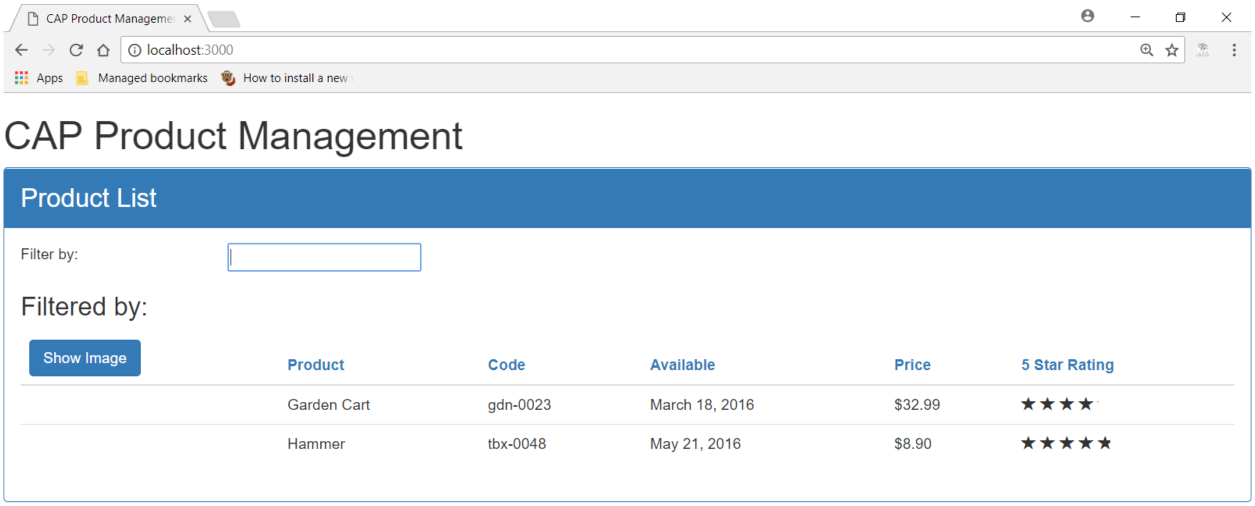
Injecting Service to product.component.ts

12. Now inject the service which we have created into product.component.ts file.

product.component.ts

```
import { Component, OnInit } from 'angular2/core';
import { IProduct } from './product';
import { ProductFilterPipe } from './product-filter.pipe';
import { StarComponent } from '../shared/star.component';
import { ProductService } from './product.service';
@Component({
    selector: 'pm-products',
    templateUrl: 'app/products/product-list.component.html',
    styleUrls: ['app/products/product-list.component.css'],
    pipes: [ProductFilterPipe],
    directives: [StarComponent]

})
export class ProductListComponent implements OnInit{
    pageTitle: string = 'Product List';
    imageWidth: number = 50;
    imageMargin: number = 2;
    showImage: boolean = false;
    listFilter: string ;
    products: IProduct[];

    constructor(private _productService: ProductService){

    }

    toggleImage(): void{
        this.showImage = !this.showImage;
    }

    ngOnInit(): void{
       this.products = this._productService.getProducts();
    }

    onRatingClicked(message: string): void{
        this.pageTitle = 'Product List: ' + message;
    }

}
```

Now you can see the output in your browser.



CAP Product Management

Product List

Filter by:

Filtered by:

Show Image

Product	Code	Available	Price	5 Star Rating
Garden Cart	gdn-0023	March 18, 2016	$32.99	★★★★
Hammer	tbx-0048	May 21, 2016	$8.90	★★★★★

Retrieving Data Using http:

13. Add http library in your index.html file, this file already included with rxJs library so no need to include that.

index.html

```
<!DOCTYPE html>
<html>

<head>
    <title>CAP Product Management</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <link href="node_modules/bootstrap/dist/css/bootstrap.css" rel="stylesheet"
/>
    <link href="app/app.component.css" rel="stylesheet" />

    <!-- 1. Load libraries -->
    <!-- IE required polyfills, in this exact order -->
    <script
src="node_modules/angular2/es6/dev/src/testing/shims_for_IE.js"></script>
    <script src="node_modules/es6-shim/es6-shim.min.js"></script>
    <script src="node_modules/systemjs/dist/system-polyfills.js"></script>

    <script src="node_modules/angular2/bundles/angular2-polyfills.js"></script>
    <script src="node_modules/systemjs/dist/system.src.js"></script>
    <script src="node_modules/rxjs/bundles/Rx.js"></script>
    <script src="node_modules/angular2/bundles/angular2.dev.js"></script>

     <!-- required for http -->
    <script src="node_modules/angular2/bundles/http.dev.js"></script>

    <!-- 2. Configure SystemJS -->
    <script>
        System.config({
        packages: {
          app: {
            format: 'register',
            defaultExtension: 'js'
          }
        }
      });
      System.import('app/main')
            .then(null, console.error.bind(console));
    </script>
</head>

<body>
    <pm-app>Loading App..</pm-app>

</body>

</html>
```

14. Include HTTP_PROVIDERS service in your app.component.ts file as mentioned below:

app.component.ts

```
import { Component } from 'angular2/core';
import { ProductListComponent } from './products/product-list.component';
import {ProductService} from './products/product.service';
import { HTTP_PROVIDERS } from 'angular2/http';
import 'rxjs/Rx'; //Load all features
@Component({
    selector: 'pm-app',
    template: `
```

```
        <div>
            <h1>{{pageTitle}}</h1>
            <pm-products></pm-products>
        </div>
    `,
    directives: [ProductListComponent],
    providers: [ProductService, HTTP_PROVIDERS]
})

export class AppComponent{
    pageTitle: string= 'CAP Product Management';

}
```

15. Sending an Http Request
    a. Now you can remove hard coded product list from product.service.ts file. And replace that with http request.

product.service.ts

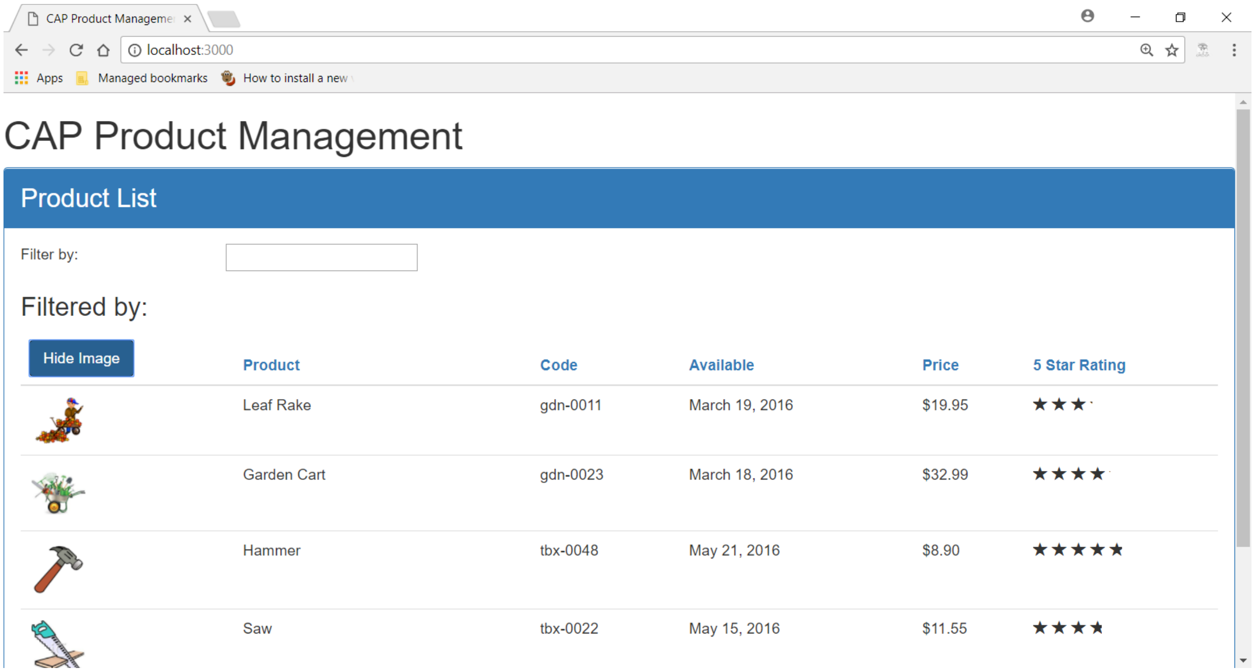
```
import {Injectable} from 'angular2/core';
import {IProduct} from './product';
import {Http, Response} from 'angular2/http';
import {Observable} from 'rxjs/Observable';

@Injectable()
export class ProductService{
    private  _productUrl = 'api/products/products.json';

    constructor(private _http : Http){

    }

    getProducts(): Observable<IProduct[]>{
        return this._http.get(this._productUrl)
            .map((response:Response) => <IProduct[]>response.json())
            .do( data => console.log("All: " +JSON.stringify(data)))
            .catch(this.handleError);

    }

    private handleError(error:Response){
        console.error(error);
        return Observable.throw(error.json().error || 'Server Error');
    }
}
```

16. Subscribe your product in your product-list.component.ts file as mentioned below.

```
import { Component, OnInit } from 'angular2/core';
import { IProduct } from './product';
import { ProductFilterPipe } from './product-filter.pipe';
import { StarComponent } from '../shared/star.component';
import { ProductService } from './product.service';
@Component({
    selector: 'pm-products',
    templateUrl: 'app/products/product-list.component.html',
    styleUrls: ['app/products/product-list.component.css'],
    pipes: [ProductFilterPipe],
    directives: [StarComponent]

})
export class ProductListComponent implements OnInit{
    pageTitle: string = 'Product List';
    imageWidth: number = 50;
    imageMargin: number = 2;
    showImage: boolean = false;
    listFilter: string ;
    products: IProduct[];
    errorMessage: string;

    constructor(private _productService: ProductService){

    }

    toggleImage(): void{
        this.showImage = !this.showImage;
    }

    ngOnInit(): void{
       this._productService.getProducts()
       .subscribe(
            products => this.products= products,
            error => this.errorMessage = <any>error);
    }

    onRatingClicked(message: string): void{
        this.pageTitle = 'Product List: ' + message;
    }

}
```

17. Check whether products.json file has been include in your api/products directory.

Now you can see the below output



CAP Product Management

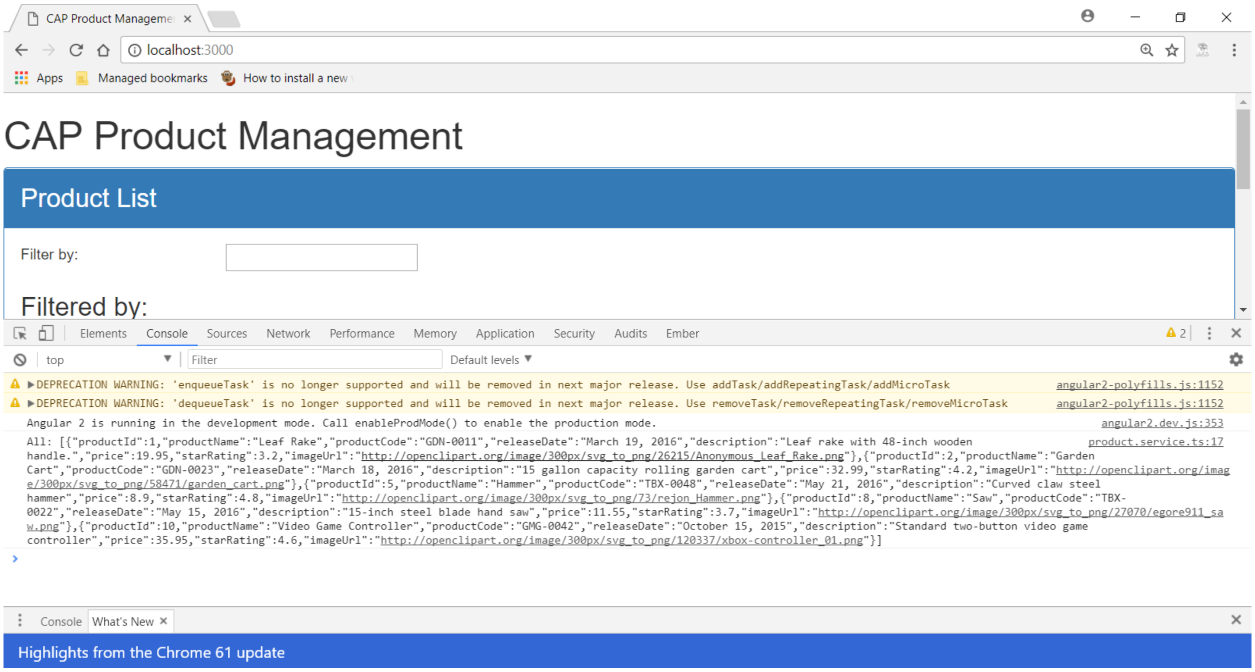
Product List

Filter by:

Filtered by:

Hide Image	Product	Code	Available	Price	5 Star Rating
	Leaf Rake	gdn-0011	March 19, 2016	$19.95	★★★
	Garden Cart	gdn-0023	March 18, 2016	$32.99	★★★★
	Hammer	tbx-0048	May 21, 2016	$8.90	★★★★★
	Saw	tbx-0022	May 15, 2016	$11.55	★★★★

Also check your console window in the browser:



CAP Product Management

Product List

Filter by:

Filtered by:

```
▶DEPRECATION WARNING: 'enqueueTask' is no longer supported and will be removed in next major release. Use addTask/addRepeatingTask/addMicroTask   angular2-polyfills.js:1152
▶DEPRECATION WARNING: 'dequeueTask' is no longer supported and will be removed in next major release. Use removeTask/removeRepeatingTask/removeMicroTask   angular2-polyfills.js:1152
Angular 2 is running in the development mode. Call enableProdMode() to enable the production mode.   angular2.dev.js:353
All: [{"productId":1,"productName":"Leaf Rake","productCode":"GDN-0011","releaseDate":"March 19, 2016","description":"Leaf rake with 48-inch wooden   product.service.ts:17
handle.","price":19.95,"starRating":3.2,"imageUrl":"http://openclipart.org/image/300px/svg_to_png/26215/Anonymous_Leaf_Rake.png"},{"productId":2,"productName":"Garden
Cart","productCode":"GDN-0023","releaseDate":"March 18, 2016","description":"15 gallon capacity rolling garden cart","price":32.99,"starRating":4.2,"imageUrl":"http://openclipart.org/imag
e/300px/svg_to_png/58471/garden_cart.png"},{"productId":5,"productName":"Hammer","productCode":"TBX-0048","releaseDate":"May 21, 2016","description":"Curved claw steel
hammer","price":8.9,"starRating":4.8,"imageUrl":"http://openclipart.org/image/300px/svg_to_png/73/rejon_Hammer.png"},{"productId":8,"productName":"Saw","productCode":"TBX-
0022","releaseDate":"May 15, 2016","description":"15-inch steel blade hand saw","price":11.55,"starRating":3.7,"imageUrl":"http://openclipart.org/image/300px/svg_to_png/27070/egore911_sa
w.png"},{"productId":10,"productName":"Video Game Controller","productCode":"GMG-0042","releaseDate":"October 15, 2015","description":"Standard two-button video game
controller","price":35.95,"starRating":4.6,"imageUrl":"http://openclipart.org/image/300px/svg_to_png/120337/xbox-controller_01.png"}]
>
```

Highlights from the Chrome 61 update