

# Atelier Angular :

## La recherche dans Angular

### Objectifs :

1. Rechercher les produits par catégorie,
2. Rechercher par nom des produits en utilisant une api,
3. Rechercher par nom des produits en utilisant l'événement keyup,
4. Rechercher par nom des produits en utilisant un pipe,
5. Rechercher des produits en utilisant le module ng2-search-filter.

### Rechercher les produits par catégorie

1. Créer le composant Web rechercheParCategorie en tapant :

**ng g c rechercheParCategorie --skipTests=true --inlineStyle=true**

2. Editer le fichier **app-routing.module.ts** et y ajouter la Route « rechercheParCategorie » :

```
{path: "rechercheParCategorie", component : RechercheParCategorieComponent}
```

3. Editer le fichier app.component.html pour ajouter le lien vers le composant rechercheParCategorie:

```
<a class="dropdown-item" routerLink="/rechercheParCategorie" >Recherche Par  
Catégorie</a>
```

4. Tester le lien en cliquant sur « Recherche Par Catégorie »

5. Ajouter à la classe ProduitService la méthode rechercheParCategorie() :

```
rechercherParCategorie(idCat: number):Observable< Produit[]> {  
    const url = `${this.apiUrl}/prodscat/${idCat}`;  
    return this.http.get<Produit[]>(url);  
}
```

6. Copier dans recherche-par-categorie.component.html le contenu du fichier produits.component.html
7. Ajouter la liste déroulante contenant les catégories au fichier rechercheParCategorie.component.html :

```
<div class="col-sm-4 col-md-4 col-lg-4 mb-4">  
    <label>Categories </label>  
    <select class="form-control form-control-lg" id="idCat" name="idCat"  
    [(ngModel)]="IdCategorie" (change)="onChange()">  
        <option *ngFor="let cat of categories" value="{{cat.idCat}}">  
            {{cat.nomCat}}  
        </option>  
    </select>  
</div>
```

8. Modifier la méthode ngOnInit() de la classe RechercheParCategorieComponent :

```
produits! : Produit[];  
IdCategorie! : number;  
categories! : Categorie[];
```

```
ngOnInit(): void {  
    this.produitService.listeCategories().  
    subscribe(cats => {this.categories = cats._embedded.categories;  
        console.log(cats);  
    });  
}
```

9. Ajouter la méthode onChange() à la classe RechercheParCategorieComponent :

```
onChange() {  
    this.produitService.rechercherParCategorie(this.IdCategorie).  
    subscribe(prods =>{this.produits=prods});  
}
```

## Rechercher par nom des produits en utilisant une api

10. Au niveau du projet Spring boot, ajouter une api que recherche les produits dont le nom contient un texte donné :

```
@RequestMapping(value="/prodsByName/{nom}",method = RequestMethod.GET)
public List<Produit> findByNomProduitContains(@PathVariable("nom") String nom) {
    return produitService.findByNomProduitContains(nom);
}
```

11. Créer le composant Web rechercheParNom en tapant :

**ng g c rechercheParNom --skipTests=true --inlineStyle=true**

12. Editer le fichier **app-routing.module.ts** et y ajouter la Route « rechercheParNom » :

```
{path: "rechercheParNom", component : RechercheParNomComponent},
```

13. Editer le fichier **app.component.html** pour ajouter le lien vers le composant rechercheParCategorie:

```
<a class="dropdown-item" routerLink="/rechercheParNom" >Recherche Par Nom</a>
```

14. Copier dans **recherche-par-categorie.component.html** le contenu du fichier **produits.component.html** et remplacer la liste des catégories par un input :

```
<div class="container">

  <form class="form-inline">
    <div class="form-group mx-sm-3 mb-2 mt-2 ">
      <label>Nom Produit</label>
      <input type="text" class="form-control" [(ngModel)]="nomProduit" name="nomProduit" class="form-control">
      <button type="submit" (click)="rechercherProds()" class="btn btn-primary mb-2 ml-2">Rechercher</button>
    </div>
  </form>

  <div class="card shadow mb-4">
    <div class="card-body">
      <table class="table table-striped">
        <thead>
          <tr> <th>Id</th><th>Nom</th> <th>Prix</th> <th>Date
Création</th><th>Catégorie</th> </tr>
        </thead>
        <tbody *ngFor="let produit of produits">
```

```

        <tr>
          <td>{{produit.idProduit}}</td>
          <td>{{produit.nomProduit}}</td>
          <td>{{produit.prixProduit}}</td>
          <td>{{produit.dateCreation | date:'dd/MM/yyyy'}}</td>
          <td>{{produit.categorie?.nomCat}}</td>
        </tr>
      </tbody>
    </table>
  </div>
</div>
</div>

```

15. Ajouter la méthode `rechercherParNom()` à la classe `ProduitService` :

```

rechercherParNom(nom: string):Observable< Produit[]> {
  const url = `${this.apiUrl}/prodsByName/${nom}`;
  return this.http.get<Produit[]>(url);
}

```

16. Ajouter la méthode `rechercherProds ()` à la classe `RechercheParNomComponent` :

```

rechercherProds(){
  this.produitService.rechercherParNom(this.nomProduit).
    subscribe(prods => {
      this.produits = prods;
      console.log(prods)});
}

```

17. Tester votre travail

## Rechercher par nom des produits en utilisant l'événement keyup

18. Modifier le fichier recherche-par-nom.component.html comme suit :

```
<form class="form-inline">
  <div class="form-group mx-sm-3 mb-2 mt-2 ">
    <label>Nom Produit en tapant</label>
    <input class="form-control" #searchTerm (keyup)="onKeyUp(searchTerm.value)">
  </div>
</form>
```

19. Ajouter la méthode onKeyUp à la classe RechercheParNomComponent :

```
allProduits! : Produit[];
searchTerm!: string;

ngOnInit(): void {
  this.produitService.listeProduit().subscribe(prods => {
    console.log(prods);
    this.allProduits = prods;
  });
}

onKeyUp(filterText : string){
  this.products = this.allProduits.filter(item =>
item.nomProduit.toLowerCase().includes(filterText));
}
```

20. Tester votre travail

## Rechercher par nom des produits en utilisant un pipe

21. Créer le pipe searchFilter :

```
ng g pipe searchFilter
```

22. Modifier la méthode transform comme suit :

```
transform(list: any[], filterText: string): any {
  return list ? list.filter(item =>
item.nomProduit.toLowerCase().includes(filterText)) : [];
}
```

23. Modifier le fichier recherche-par-nom.component.html comme suit :

```
<form class="form-inline">
  <div class="form-group mx-sm-3 mb-2 mt-2 ">
    <label>Nom Produit en tapant</label>
  <input #nom [(ngModel)]="searchTerm" name="searchTerm" class="form-control" >
  </div>
</form>
```

```
<tbody *ngFor="let produit of produits | searchFilter: searchTerm ">
```

24. Tester votre travail

## Rechercher des produits en utilisant le module ng2-search-filter

25. Installez le module ng2-search-filter

```
npm install ng2-search-filter
```

26. Ajouter le module ng2-search-filter au fichier app.module.ts

```
imports: [
  ...
  Ng2SearchPipeModule
]
```

27. Modifier le fichier recherche-par-nom.component.html comme suit :

```
<tbody *ngFor="let produit of produits | filter:searchTerm ">
```

28. Tester votre travail