

## Angular Iterations and Conditions

### Shopping.component.ts

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

@Component({
  selector: 'app-shopping',
  templateUrl: './shopping.component.html',
  styleUrls: ['./shopping.component.css']
})
export class ShoppingComponent {
  categories = ["Select a Category", "Electronics", "Footwear", "Fashion"];
  electronics = ["Select Electronic Product", "JBL Speaker", "Earpods"];
  footwear = ["Select Footwear", "Nike Casuals", "Lee Cooper Boot"];
  fashion = ["Select Fasion Wear", "Shirt", "Jeans"];
  products:string[] = [];

  data = [
    {Name: "JBL Speaker", Price: 4500.55, Photo: "assets/speaker.jpg"},
    {Name: "Earpods", Price: 3500.55, Photo: "assets/earpods.jpg"},
    {Name: "Nike Casuals", Price: 6500.55, Photo: "assets/shoe.jpg"},
    {Name: "Lee Cooper Boot", Price: 5500.55, Photo: "assets/shoe1.jpg"},
    {Name: "Shirt", Price: 1500.55, Photo: "assets/shirt.jpg"},
    {Name: "Jeans", Price: 2500.55, Photo: "assets/jeans.jpg"},
  ];

  selectedCategoryName = "Select a Category";

  CategoryChanged(){
    switch(this.selectedCategoryName) {
      case "Electronics":
```

```

        this.products = this.electronics;

        break;

        case "Footwear":

            this.products = this.footwear;

            break;

        case "Fashion":

            this.products = this.fashion;

            break;

        default:

            this.products = ["Please Select a Category"];

            break;

    }
}

selectedProductName:string = "";

searchedProduct:any = {

    Name: "",

    Price: 0,

    Photo: ""

};

ProductChanged(){

    this.searchedProduct = this.data.find(product=>
product.Name==this.selectedProductName);

}

CartItems:any[] = [];

CartItemsCount = 0;

GetCartItemsCount(){

    this.CartItemsCount = this.CartItems.length;

}

AddToCartClick(){

    this.CartItems.push(this.searchedProduct);

```

```

    alert("Item Added to Your Shopping Cart");
    this.GetCartItemsCount();
}
isCartVisible = false;
ToggleCartDisplay(){
    this.isCartVisible = (this.isCartVisible===false)?true:false;
}
RemoveCartItem(index:number){
    let choice = confirm("Are you sure, want to delete?");
    if(choice===true) {
        this.CartItems.splice(index,1);
        this.GetCartItemsCount();
    }
}
}
}

```

### **Shopping.component.html**

```

<div class="container-fluid">
    <div class="row">
        <div class="col">
            <h2 class="bg-danger text-white text-center p-2"> <span class="fa fa-shopping-
bag"></span> Amazon Shopping</h2>
        </div>
    </div>
    <div class="row mt-2">
        <div class="col-3">
            <div class="mt-2">
                <label>Select Category</label>
                <div>
                    <select (change)="CategoryChanged()" [(ngModel)]="selectedCategoryName"
class="form-select">

```

```

        <option *ngFor="let item of categories">
            {{item}}
        </option>
    </select>
</div>
</div>
<div class="mt-2">
    <label>Select Product</label>
    <div>
        <select (change)="ProductChanged()" [(ngModel)]="selectedProductName"
class="form-select">
            <option *ngFor="let item of products">
                {{item}}
            </option>
        </select>
    </div>
</div>
<div class="mt-2">
    <label>Product Preview</label>
    <div class="card">
        <div class="card-header text-center">
            <h2>{{searchedProduct.Name}}</h2>
        </div>
        <div class="card-body text-center">
            <img width="200" [src]="searchedProduct.Photo" height="200">
            <p>{{searchedProduct.Price}}</p>
        </div>
        <div class="card-footer">
            <div class="d-grid">
                <button class="btn btn-danger" (click)="AddToCartClick()" >

```

```

        <span class="fa fa-shopping-cart"></span>

        Add to Cart

    </button>

</div>

</div>

</div>

</div>

</div>

<div class="col-6">

    <table *ngIf="isCartVisible" class="table table-hover caption-top">

        <caption>Review Your Cart Items</caption>

        <thead>

            <tr>

                <th>Name</th>

                <th>Price</th>

                <th>Preview</th>

                <th></th>

            </tr>

        </thead>

        <tbody>

            <tr *ngFor="let item of CartItems; let index=index">

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

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

                <td>

                    <img [src]="item.Photo" width="50" height="50" />

                </td>

                <td>

                    <button class="btn-outline-danger" (click)="RemoveCartItem(index)" >

                        <span class="fa fa-trash"></span>


```

```

        </button>
      </td>
    </tr>
  </tbody>
</table>
</div>
<div class="col-3">
  <div class="d-grid">
    <button class="btn btn-danger" (click)="ToggleCartDisplay()" >
      <span class="fa fa-shopping-cart"></span>
      Your Cart Items
      <span class="badge bg-light text-dark">{{CartItemsCount}}</span>
    </button>
  </div>
</div>
</div>
</div>

```

### Ex: Sending Object as Event Argument

#### Likesdemo.component.ts

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

```

@Component({
  selector: 'app-likesdemo',
  templateUrl: './likesdemo.component.html',
  styleUrls: ['./likesdemo.component.css']
})

```

```
export class LikesdemoComponent {
```

```
  Products = [
```

```

{Name: "JBL Speaker", Photo: "assets/speaker.jpg", Likes:0, Dislikes:0},
{Name: "Nike Casuals", Photo: "assets/shoe.jpg", Likes:0, Dislikes: 0},
{Name: "Shirt", Photo: "assets/shirt.jpg", Likes:0, Dislikes: 0}
];

LikesClick(item:any){
    item.Likes++;
}

DislikesClick(item:any){
    item.Dislikes++;
}
}

```

### Likesdemo.component.html

```

<div class="container">
  <div class="mt-3 card-group">
    <div class="card me-3" *ngFor="let item of Products">
      <img [src]="item.Photo" height="200" class="card-img-top">
      <div class="card-body">
        <h2>{{item.Name}}</h2>
      </div>
      <div class="card-footer text-center">
        <div class="btn-group">
          <button class="btn btn-outline-success" (click)="LikesClick(item)" >
            <span class="fa fa-thumbs-up"></span>
            [{{item.Likes}}] Likes
          </button>
          <button class="btn btn-outline-danger" (click)="DislikesClick(item)">

```

```

        <span class="fa fa-thumbs-down"></span>
        [{{item.Dislikes}}] Dislikes
    </button>
</div>
</div>
</div>
</div>
</div>

```

## Iterations and Conditions

Ex:

### Conditionsdemo.component.ts

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

```
@Component({
```

```
  selector: 'app-conditionsdemo',
```

```
  templateUrl: './conditionsdemo.component.html',
```

```
  styleUrls: ['./conditionsdemo.component.css']
```

```
})
```

```
export class ConditionsdemoComponent {
```

```
  products = [
```

```
    {Name: "Nike Casuals", Price: 4500.55, Photo: "assets/shoe.jpg", Category:
    "Footwear"},
```

```
    {Name: "Lee Cooper Boot", Price: 6000.55, Photo: "assets/shoe1.jpg",
    Category: "Footwear"},
```

```
    {Name: "Shirt", Price: 1300.44, Photo: "assets/shirt.jpg", Category:
    "Fashion"},
```



```

    {Name: "Jeans", Price: 4300.44, Photo: "assets/jeans.jpg", Category:
"Fashion"},

    {Name: "JBL Speaker", Price: 2300.44, Photo: "assets/speaker.jpg",
Category: "Electronics"},

    {Name: "EarPods", Price: 7300.44, Photo: "assets/earpods.jpg", Category:
"Electronics"},

];

categories = ["All", "Electronics", "Footwear", "Fashion"];

selectedCategoryName = "All";

productName = "";

}

```

### Conditionsdemo.component.html

```

<div class="container-fluid">

  <h1 class="text-primary text-center">Amazon Shopping</h1>

  <div class="row">

    <div class="col-2">

      <div class="mt-2">

        <label class="form-label">Select Category</label>

        <div>

          <select class="form-select" [(ngModel)]="selectedCategoryName">

            <option *ngFor="let item of categories">

              {{item}}

            </option>

          </select>

        </div>

      </div>

    </div>

    <div class="mt-2">

```

```

<label class="form-label">Select Category</label>

<div>

  <ul class="list-unstyled">

    <li><input type="radio" [(ngModel)]="selectedCategoryName"
name="Category" value="All" >All</li>

    <li><input type="radio" [(ngModel)]="selectedCategoryName"
name="Category" value="Electronics">Electronics</li>

    <li><input type="radio" [(ngModel)]="selectedCategoryName"
name="Category" value="Footwear">Footwear</li>

    <li><input type="radio" [(ngModel)]="selectedCategoryName"
name="Category" value="Fashion">Fashion</li>

  </ul>

</div>

</div>

</div>

<div class="col-10">

  {{productName}}

  <div class="card-group">

    <ng-container *ngFor="let product of products">

      <div class="card" *ngIf="selectedCategoryName=='All' ||
selectedCategoryName==product.Category" >

        <img [src]="product.Photo" height="200" class="card-img-top">

        <div class="card-header">

          <h3>{{product.Name}}</h3>

        </div>

        <div class="card-body">

          <h5>{{product.Price}}</h5>

        </div>

```

```
        </div>
    </ng-container>
</div>
</div>
</div>
</div>
```

## Attribute Directives

ngModel  
ngStyle  
ngClass

EX: Class Binding

### **Classbinding.component.ts**

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

```
@Component({
  selector: 'app-classbinding',
  templateUrl: './classbinding.component.html',
  styleUrls: ['./classbinding.component.css']
})
```

```
export class ClassbindingComponent{
  flagBackground = false;
  flagText = false;
  flagBorder = false;
}
```

### **Classbinding.component.css**

```
.border-style {
    border:2px solid tomato;
}

.background-style {
    background-color: yellow;
}

.text-style {
    text-align: center;
    color:red;
    padding: 4px;
}
```

### **Classbinding.component.html**

```
<div class="container-fluid">
  <div class="form-switch form-check">
    <div><input type="checkbox" [(ngModel)]="flagBackground" class="form-check-input">Background</div>
    <div><input type="checkbox" [(ngModel)]="flagText" class="form-check-input">Text</div>
    <div><input type="checkbox" [(ngModel)]="flagBorder" class="form-check-input">Border</div>
  </div>
  <h2 [ngClass]="{'background-style': flagBackground, 'text-style': flagText, 'border-style': flagBorder}">Class Binding in Angular</h2>
</div>
```

### **Style Binding Example**

#### **Component.ts**

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

```
@Component({  
  selector: 'app-classbinding',  
  templateUrl: './classbinding.component.html',  
  styleUrls: ['./classbinding.component.css']  
})
```

```
export class ClassbindingComponent {  
  BackgroundColor = '#ff00ff';  
  TextColor = "";  
  Alignment = "";  
}
```

### **Component.html**

```
<div class="container-fluid">  
  <fieldset>  
    <legend>Select Effects</legend>  
    <dl>  
      <dt>Background Color</dt>  
      <dd><input [(ngModel)]="BackgroundColor" type="color" class="form-control-color"></dd>  
      <dt>Text Align</dt>  
      <dd>  
        <input type="radio" [(ngModel)]="Alignment" value="left" name="align">  
Left  
        <input type="radio" [(ngModel)]="Alignment" value="center" name="align"> Center
```

```

        <input type="radio" [(ngModel)]="Alignment" value="right"
name="align"> Right
    </dd>
<dt>Text Color</dt>
<dd>
    <select class="form-select" [(ngModel)]="TextColor">
        <option value="white">White</option>
        <option value="red">Red</option>
    </select>
</dd>
</dl>
</fieldset>
<h2 [ngStyle]='{'background-color': BackgroundColor, 'color': TextColor, 'text-align': Alignment}'>Welcome to Angular</h2>
</div>

```

**Ex: Using a Method return value**

### **Component.ts**

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

```

@Component({
  selector: 'app-classbinding',
  templateUrl: './classbinding.component.html',
  styleUrls: ['./classbinding.component.css']
})

```

```

export class ClassbindingComponent{
  BackgroundColor = '#ff00ff';
  TextColor = '';
}

```

```

Alignment = "";
SetEffects(){
  let effects = {
    'background-color': this.BackgroundColor,
    'text-align': this.Alignment,
    'color': this.TextColor
  }
  return effects;
}
}

```

### Component.html

```

<div class="container-fluid">
  <fieldset>
    <legend>Select Effects</legend>
    <dl>
      <dt>Background Color</dt>
      <dd><input [(ngModel)]="BackgroundColor" type="color" class="form-
control-color"></dd>
      <dt>Text Align</dt>
      <dd>
        <input type="radio" [(ngModel)]="Alignment" value="left" name="align">
Left
        <input type="radio" [(ngModel)]="Alignment" value="center"
name="align"> Center
        <input type="radio" [(ngModel)]="Alignment" value="right"
name="align"> Right
      </dd>
    </dl>
  </fieldset>
</div>

```

```
<dt>Text Color</dt>
<dd>
  <select class="form-select" [(ngModel)]="TextColor">
    <option value="white">White</option>
    <option value="red">Red</option>
  </select>
</dd>
</dl>
</fieldset>
<h2 [ngStyle]='{'background-color': BackgroundColor, 'color': TextColor, 'text-align': Alignment}'>Welcome to Angular</h2>
</div>
```

## Event Binding





