**Lesson19 Container and Nested Components and use @Output variable**

**Notes:**-

**1-we use @Output expression on Event variable to pass variable string value from the child component to parent component**

**Steps:-**

**1-on the EmployeeCount.component.ts we write the following code**

import { Component, OnInit,Input , Output, EventEmitter } from '@angular/core';

import {Employee} from '../../Service/Employee';

import {EmployeeService} from '../../Service/employee.service';

import { DELEGATE\_CTOR } from '@angular/core/src/reflection/reflection\_capabilities';

@Component({

selector: 'app-employee-count',

templateUrl: './employee-count.component.html',

styleUrls: ['./employee-count.component.css']

})

export class EmployeeCountComponent implements OnInit {

ngOnInit(){}

//we use the @Input Expression to make the variable on the Child component accessible to the parent component

@Input() all: number;

@Input() male: number;

@Input() female: number;

// Holds the selected value of the radio button

selectedRadioButtonValue: string = 'All';

//we use the Output event variable to pass the value from the child component to parent component

@Output() countRadioButtonSelectionChanged: EventEmitter<string> =

new EventEmitter<string>();

//we use the function to pass the value on the radio button to the event that will be passed to the parent component

onRadioButtonSelectionChange() {

this.countRadioButtonSelectionChanged

.emit(this.selectedRadioButtonValue);

}}

**3-we create the component called employeeList.component.ts**

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

import {Employee} from '../../Service/Employee';

import {EmployeeService} from '../../Service/employee.service';

import { DELEGATE\_CTOR } from '@angular/core/src/reflection/reflection\_capabilities';

@Component({

selector: 'app-employee',

templateUrl: './employeeList.component.html',

styleUrls: ['./employee.component.css']

})

export class EmployeeListComponent implements OnInit {

constructor(public employeeService:EmployeeService) {}

ngOnInit() {

this.ShowEmployees();}

//This variable is used to carry the value passed from the child component by the event variable to the parent component

selectedEmployeeCountRadioButton: string = 'All';

//This (function/event) is the called when we call the event on the child component

onEmployeeCountRadioButtonChange(selectedRadioButtonValue: string): void {

this.selectedEmployeeCountRadioButton = selectedRadioButtonValue;}

GetFullName(Fname:string,Lname:string){return Fname+” ”+Lname;}

ShowEmployees(){

this.employeeService.GetPersons().subscribe(data =>

this.employeeService.Employees = data as Employee[]);}

GetMaleEmployees():number{

return this.employeeService.Employees.filter(e => e.Gender === "Male").length;}

GetFemaleEmployees():number{

return this.employeeService.Employees.filter(e => e.Gender === "Female").length;}

GetAllEmployees():number{

return this.employeeService.Employees.length;}}

**4-on the employeeList.component.html we write the following code**

<app-employee-count [all]="GetAllEmployees()" [male]="GetMaleEmployees()" [female]="GetFemaleEmployees()"

//This Code is used to link the event on the child component with the method on the parent component

(countRadioButtonSelectionChanged)="onEmployeeCountRadioButtonChange($event)"></app-employee-count>

<table>

<thead>

<tr><th colspan="4">Employee Details</th></tr>

<tr>

<td></td>

<th style="text-align: left;">Full Name</th>

<th style="text-align: left;">Gender</th>

<th style="text-align: left;">Salary</th>

</tr>

</thead>

<tbody>

<ng-container \*ngFor="let employee of employeeService.Employees;">

<tr \*ngIf="selectedEmployeeCountRadioButton=='All' ||

selectedEmployeeCountRadioButton==employee.Gender">

<td>{{count}}</td>

<td>{{GetFullName(employee.Fname,employee.Lname) | employeeTitle:employee.Gender}}</td>

<td>{{employee.Gender}}</td>

<td>{{employee.Salary | currency:'USD':true:'1.4-4'}}</td>

</tr>

</ng-container>

<tr \*ngIf="employeeService.Employees==0|| !employeeService.Employees">

<td colspan="4">

No Employees to display

</td>

</tr>

</tbody>

</table>