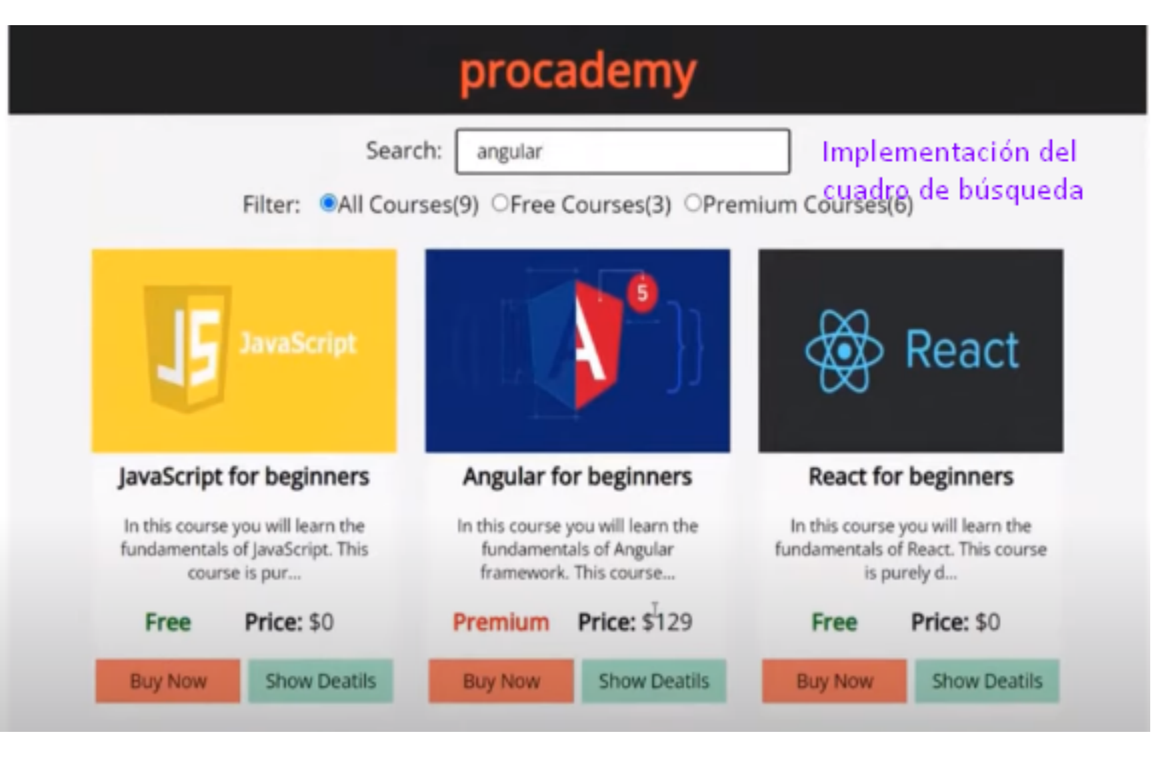
Implementing Search Functionality



procademy

Search: angular

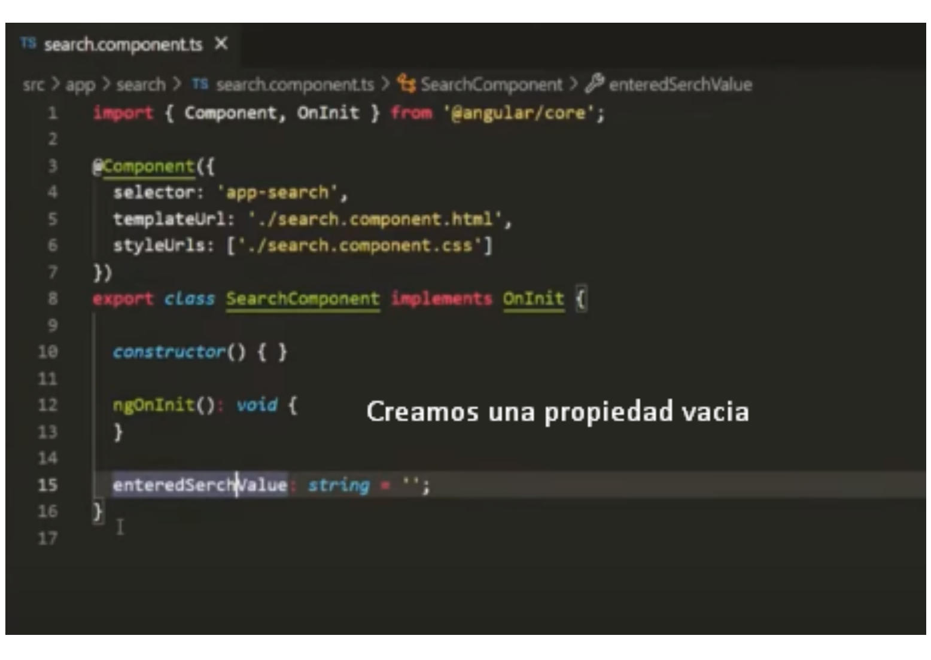
Filter: ●All Courses(9) ○Free Courses(3) ○Premium Courses(6)

JavaScript for beginners
In this course you will learn the fundamentals of JavaScript. This course is pur...
Free Price: $0

Angular for beginners
In this course you will learn the fundamentals of Angular framework. This course...
Premium Price: $129

React for beginners
In this course you will learn the fundamentals of React. This course is purely d...
Free Price: $0

Implementación del cuadro de búsqueda

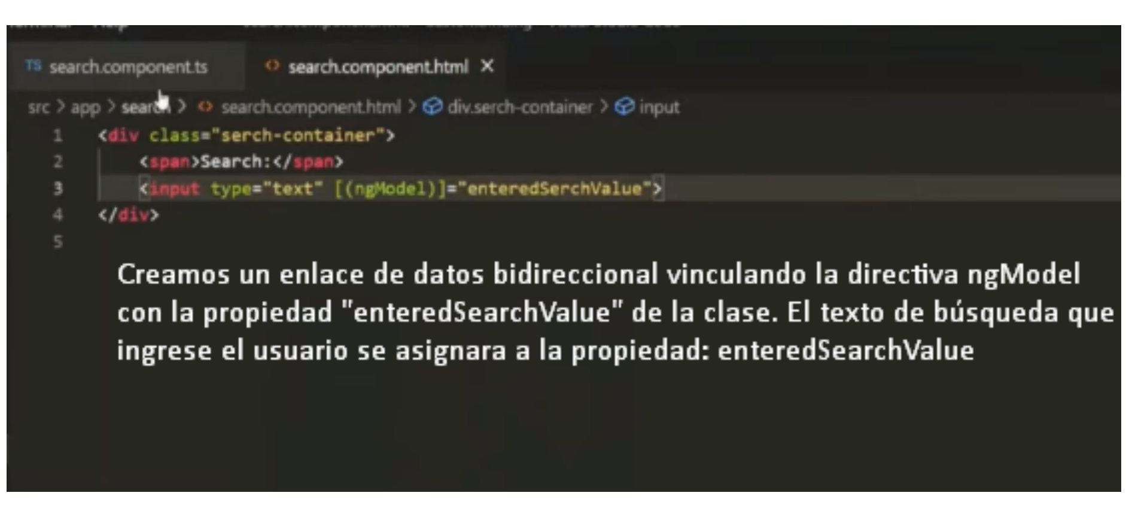


search.component.ts

src > app > search > search.component.ts > SearchComponent > enteredSerchValue

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

@Component({
  selector: 'app-search',
  templateUrl: './search.component.html',
  styleUrls: ['./search.component.css']
})
export class SearchComponent implements OnInit {

  constructor() { }

  ngOnInit(): void {
  }

  enteredSerchValue: string = '';
}
```

Creamos una propiedad vacia



search.component.html

src > app > search > search.component.html > div.serch-container > input

```
<div class="serch-container">
    <span>Search:</span>
    <input type="text" [(ngModel)]="enteredSerchValue">
</div>
```

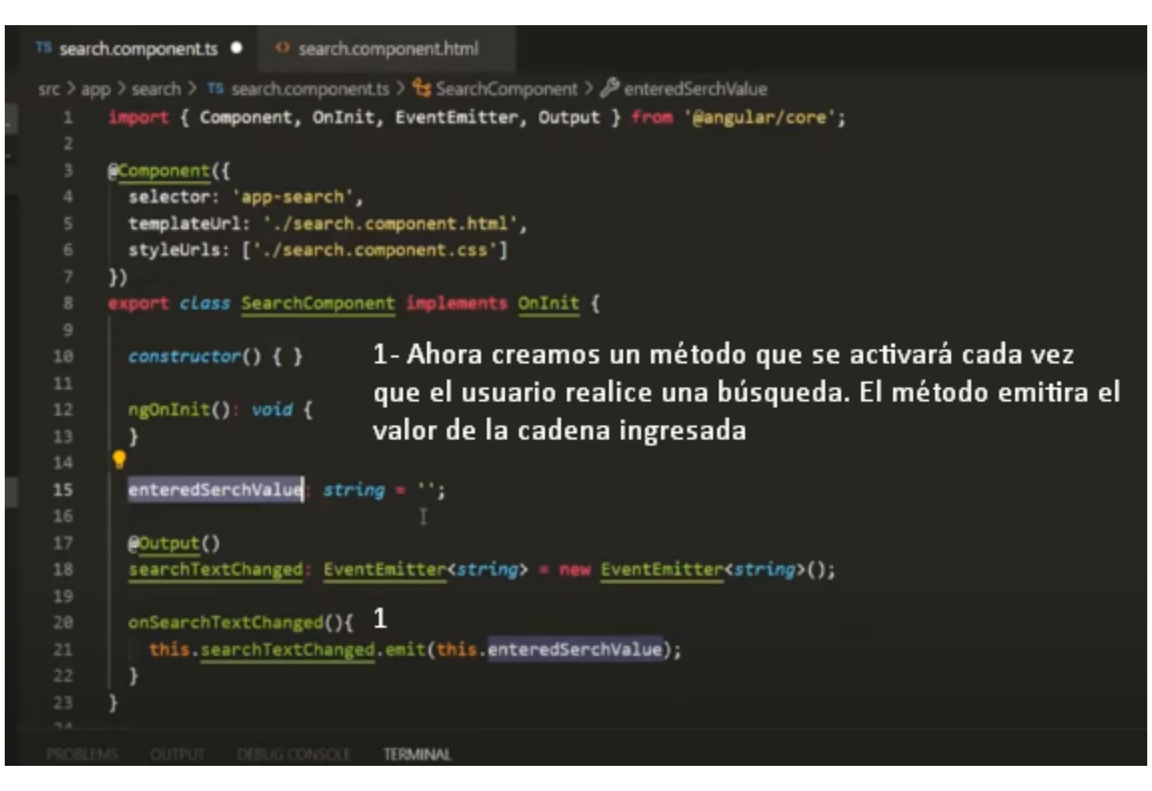
Creamos un enlace de datos bidireccional vinculando la directiva ngModel con la propiedad "enteredSearchValue" de la clase. El texto de búsqueda que ingrese el usuario se asignara a la propiedad: enteredSearchValue

search.component.ts

src > app > search > search.component.ts > SearchComponent > searchTextChanged

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

@Component({
  selector: 'app-search',
  templateUrl: './search.component.html',
  styleUrls: ['./search.component.css']
})
export class SearchComponent implements OnInit {

  constructor() { }

  ngOnInit(): void {
  }

  enteredSerchValue: string = '';

  @Output()
  searchTextChanged: EventEmitter<string> = new EventEmitter<string>();
}
```

Creamos un emisor de eventos personalizados, el mismo emitira datos de tipo String hacia el componente padre y para lograr eso utilizamos el decorador @Output(). Con esto hemos creado nuestro Vinculo de eventos Personalizado

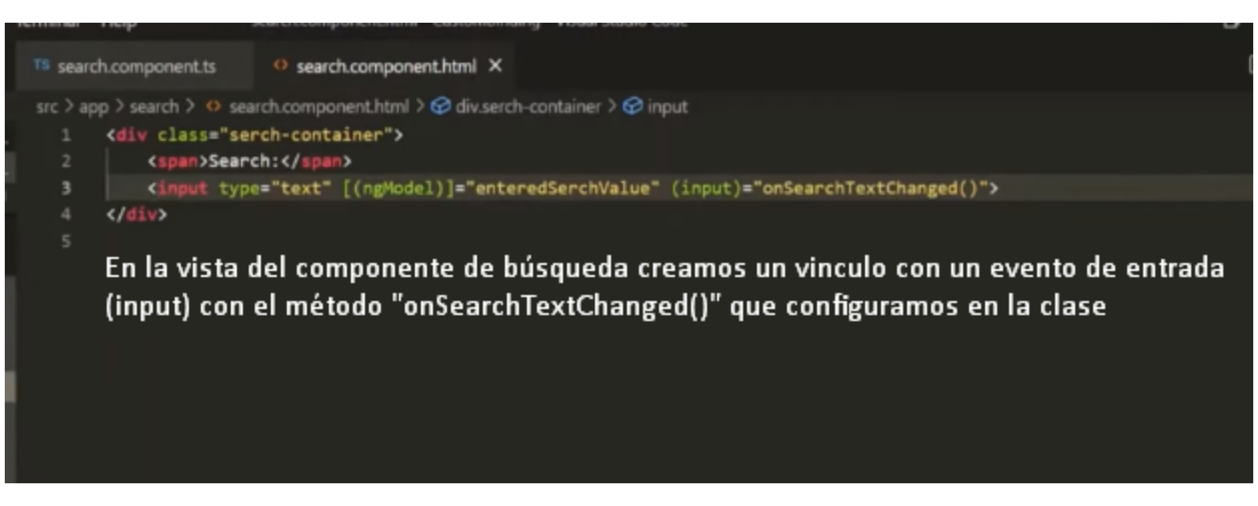


search.component.ts

src > app > search > search.component.ts > SearchComponent > enteredSerchValue

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

@Component({
  selector: 'app-search',
  templateUrl: './search.component.html',
  styleUrls: ['./search.component.css']
})
export class SearchComponent implements OnInit {

  constructor() { }

  ngOnInit(): void {
  }

  enteredSerchValue: string = '';

  @Output()
  searchTextChanged: EventEmitter<string> = new EventEmitter<string>();

  onSearchTextChanged(){
    this.searchTextChanged.emit(this.enteredSerchValue);
  }
}
```

1- Ahora creamos un método que se activará cada vez que el usuario realice una búsqueda. El método emitira el valor de la cadena ingresada

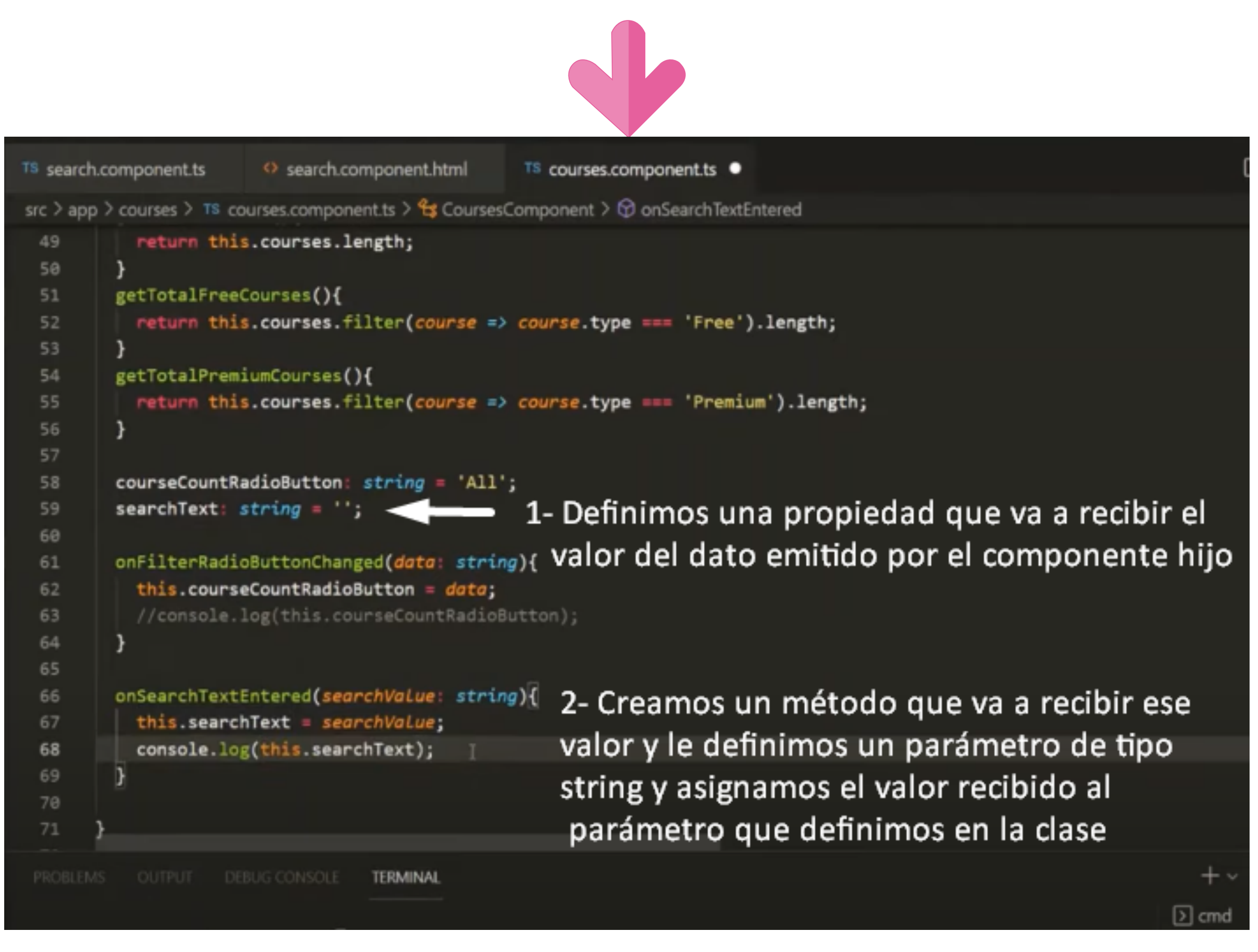


search.component.html

src > app > search > search.component.html > div.serch-container > input

```
<div class="serch-container">
    <span>Search:</span>
    <input type="text" [(ngModel)]="enteredSerchValue" (input)="onSearchTextChanged()">
</div>
```

En la vista del componente de búsqueda creamos un vinculo con un evento de entrada (input) con el método "onSearchTextChanged()" que configuramos en la clase



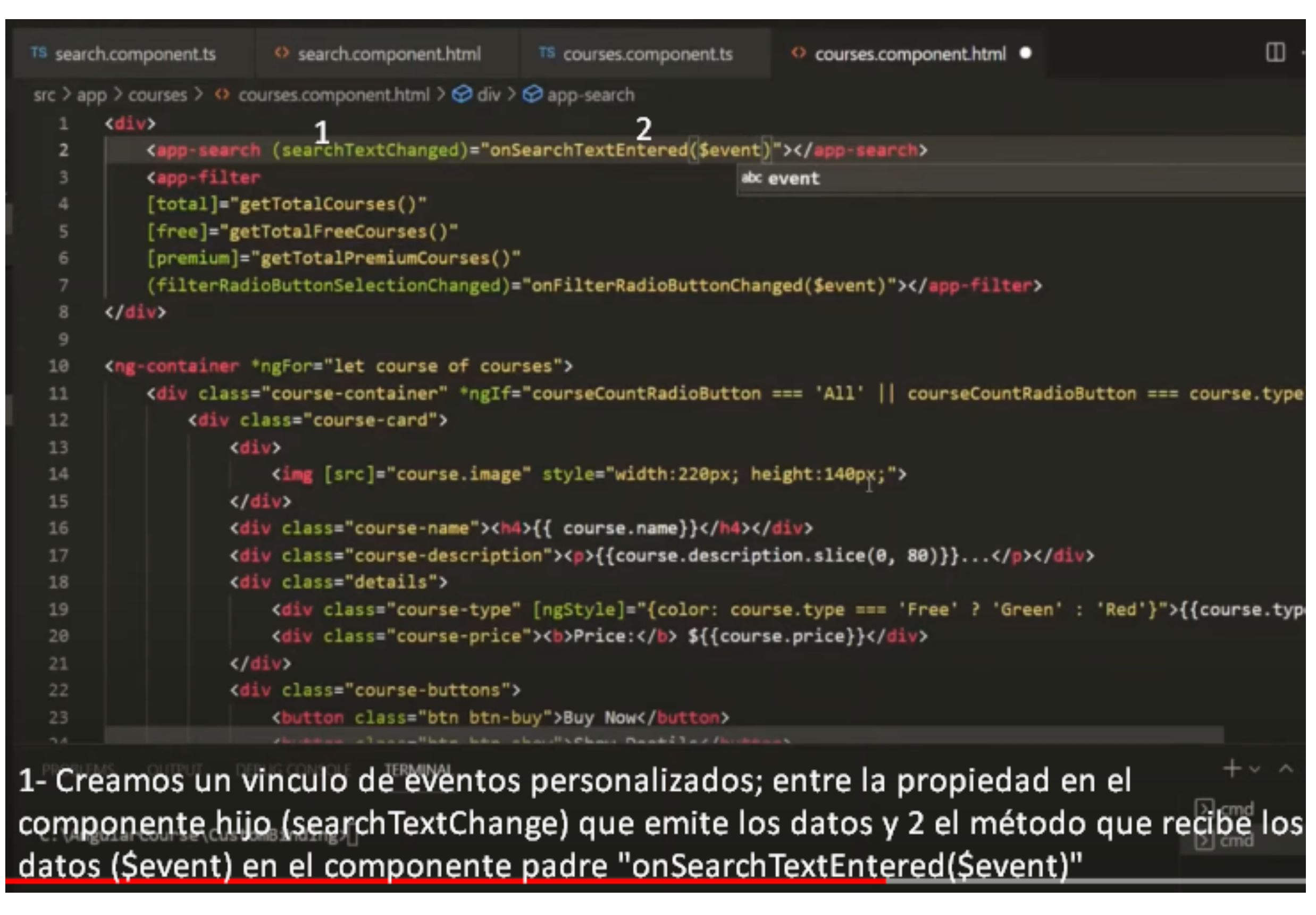
courses.component.ts

src > app > courses > courses.component.ts > CoursesComponent > onSearchTextEntered

```
      return this.courses.length;
    }
    getTotalFreeCourses(){
      return this.courses.filter(course => course.type === 'Free').length;
    }
    getTotalPremiumCourses(){
      return this.courses.filter(course => course.type === 'Premium').length;
    }

    courseCountRadioButton: string = 'All';
    searchText: string = '';

    onFilterRadioButtonChanged(data: string){
      this.courseCountRadioButton = data;
      //console.log(this.courseCountRadioButton);
    }

    onSearchTextEntered(searchValue: string){
      this.searchText = searchValue;
      console.log(this.searchText);
    }

}
```

1- Definimos una propiedad que va a recibir el valor del dato emitido por el componente hijo

2- Creamos un método que va a recibir ese valor y le definimos un parámetro de tipo string y asignamos el valor recibido al parámetro que definimos en la clase

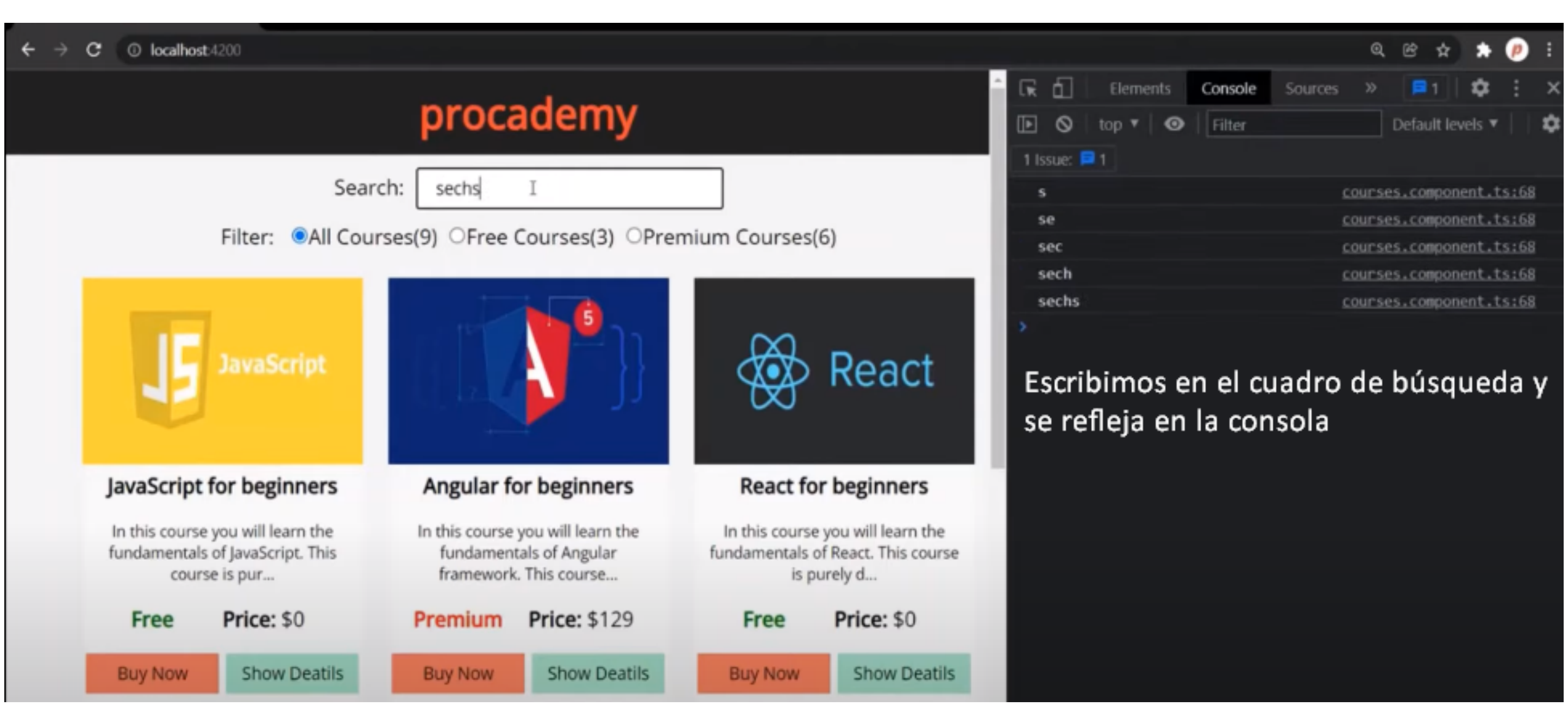


courses.component.html

src > app > courses > courses.component.html > div > app-search

```
<div>
    <app-search (searchTextChanged)="onSearchTextEntered($event)"></app-search>
    <app-filter
    [total]="getTotalCourses()"
    [free]="getTotalFreeCourses()"
    [premium]="getTotalPremiumCourses()"
    (filterRadioButtonSelectionChanged)="onFilterRadioButtonChanged($event)"></app-filter>
</div>

<ng-container *ngFor="let course of courses">
    <div class="course-container" *ngIf="courseCountRadioButton === 'All' || courseCountRadioButton === course.type
        <div class="course-card">
            <div>
                <img [src]="course.image" style="width:220px; height:140px;">
            </div>
            <div class="course-name"><h4>{{ course.name}}</h4></div>
            <div class="course-description"><p>{{course.description.slice(0, 80)}}...</p></div>
            <div class="details">
                <div class="course-type" [ngStyle]="{color: course.type === 'Free' ? 'Green' : 'Red'}">{{course.typ
                <div class="course-price"><b>Price:</b> ${{course.price}}</div>
            </div>
            <div class="course-buttons">
                <button class="btn btn-buy">Buy Now</button>
```

1- Creamos un vinculo de eventos personalizados; entre la propiedad en el componente hijo (searchTextChange) que emite los datos y 2 el método que recibe los datos ($event) en el componente padre "onSearchTextEntered($event)"



localhost:4200

procademy

Search: sechs

Filter: ●All Courses(9) ○Free Courses(3) ○Premium Courses(6)

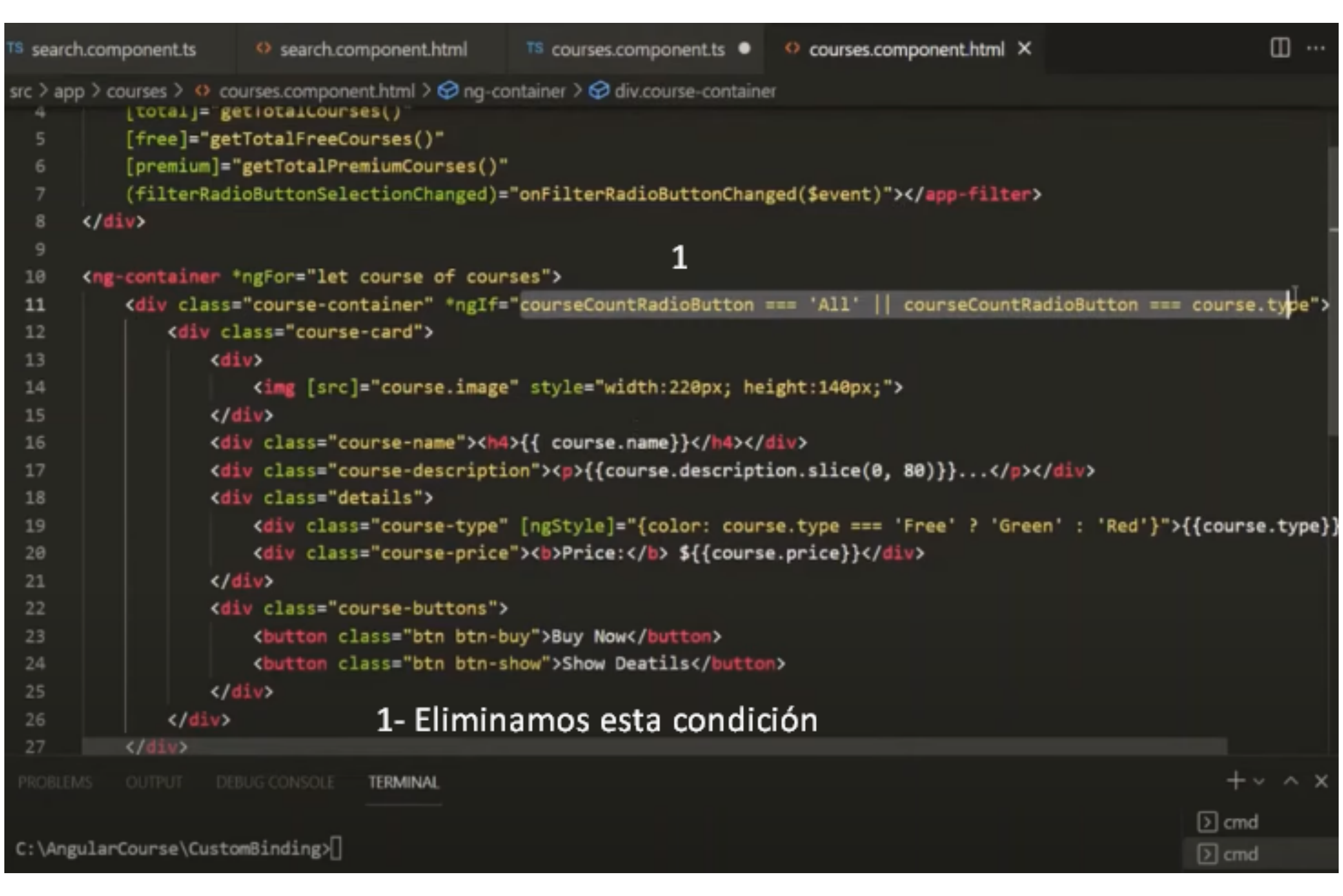
JavaScript for beginners
In this course you will learn the fundamentals of JavaScript. This course is pur...
Free Price: $0

Angular for beginners
In this course you will learn the fundamentals of Angular framework. This course...
Premium Price: $129

React for beginners
In this course you will learn the fundamentals of React. This course is purely d...
Free Price: $0

```
s        courses.component.ts:68
se       courses.component.ts:68
sec      courses.component.ts:68
sech     courses.component.ts:68
sechs    courses.component.ts:68
```

Escribimos en el cuadro de búsqueda y se refleja en la consola



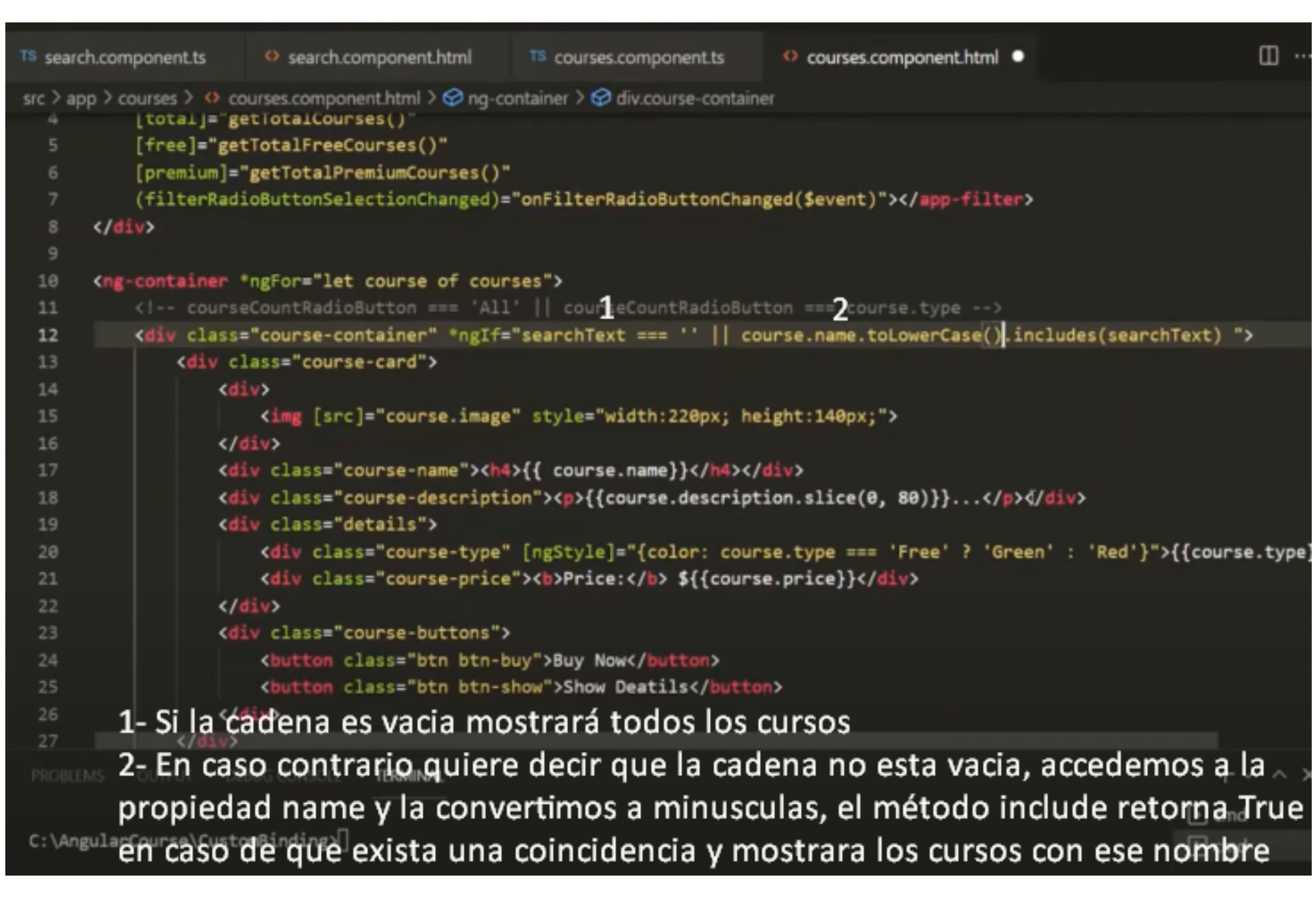
courses.component.html

src > app > courses > courses.component.html > ng-container > div.course-container

```
    [total]="getTotalCourses()"
    [free]="getTotalFreeCourses()"
    [premium]="getTotalPremiumCourses()"
    (filterRadioButtonSelectionChanged)="onFilterRadioButtonChanged($event)"></app-filter>
</div>

<ng-container *ngFor="let course of courses">
    <div class="course-container" *ngIf="courseCountRadioButton === 'All' || courseCountRadioButton === course.type">
        <div class="course-card">
            <div>
                <img [src]="course.image" style="width:220px; height:140px;">
            </div>
            <div class="course-name"><h4>{{ course.name}}</h4></div>
            <div class="course-description"><p>{{course.description.slice(0, 80)}}...</p></div>
            <div class="details">
                <div class="course-type" [ngStyle]="{color: course.type === 'Free' ? 'Green' : 'Red'}">{{course.type}}
                <div class="course-price"><b>Price:</b> ${{course.price}}</div>
            </div>
            <div class="course-buttons">
                <button class="btn btn-buy">Buy Now</button>
                <button class="btn btn-show">Show Deatils</button>
            </div>
        </div>
    </div>
```

```
C:\AngularCourse\CustomBinding>
```

1- Eliminamos esta condición



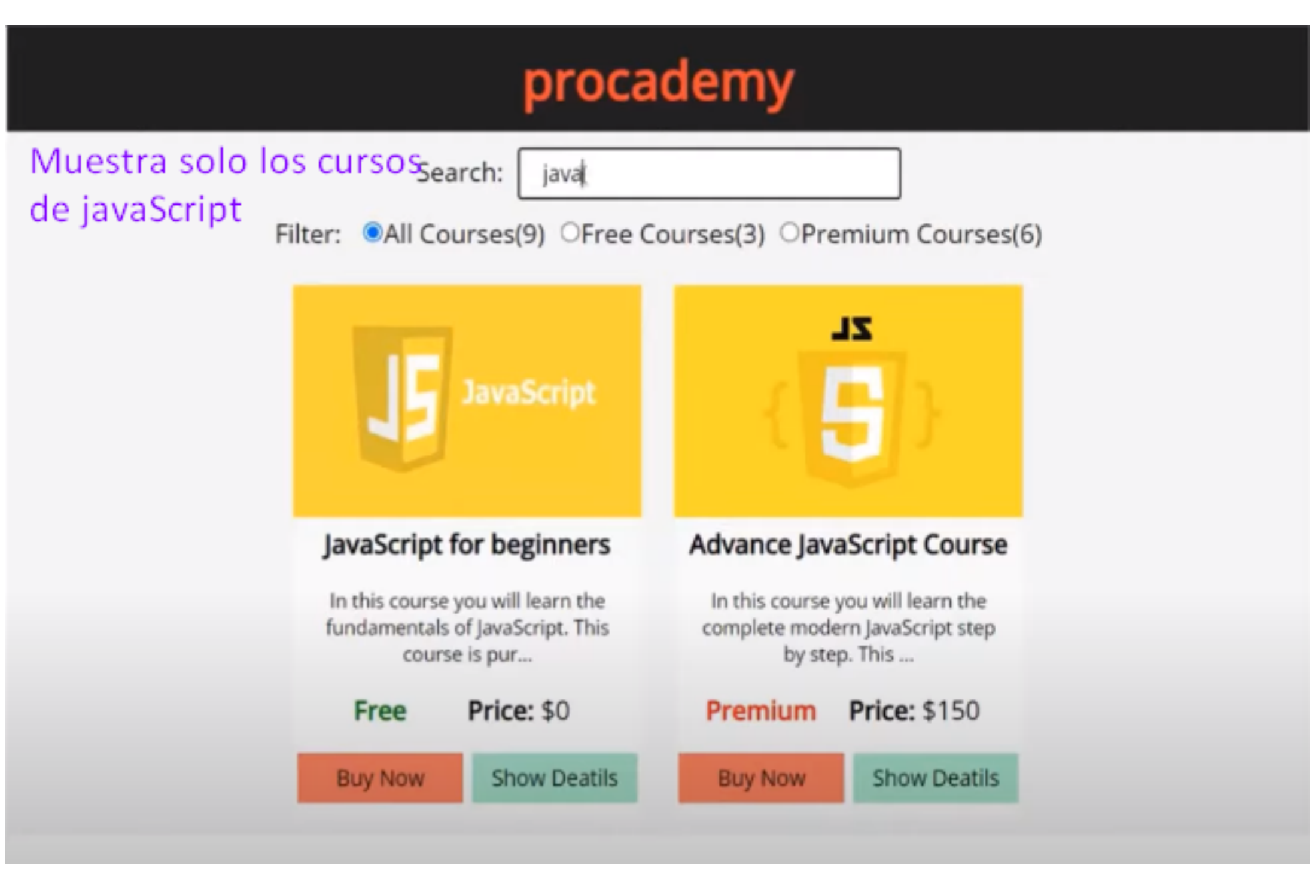
courses.component.html

src > app > courses > courses.component.html > ng-container > div.course-container

```
    [total]="getTotalCourses()"
    [free]="getTotalFreeCourses()"
    [premium]="getTotalPremiumCourses()"
    (filterRadioButtonSelectionChanged)="onFilterRadioButtonChanged($event)"></app-filter>
</div>

<ng-container *ngFor="let course of courses">
    <!-- courseCountRadioButton === 'All' || courseCountRadioButton === course.type -->
    <div class="course-container" *ngIf="searchText === '' || course.name.toLowerCase().includes(searchText) ">
        <div class="course-card">
            <div>
                <img [src]="course.image" style="width:220px; height:140px;">
            </div>
            <div class="course-name"><h4>{{ course.name}}</h4></div>
            <div class="course-description"><p>{{course.description.slice(0, 80)}}...</p></div>
            <div class="details">
                <div class="course-type" [ngStyle]="{color: course.type === 'Free' ? 'Green' : 'Red'}">{{course.type}
                <div class="course-price"><b>Price:</b> ${{course.price}}</div>
            </div>
            <div class="course-buttons">
                <button class="btn btn-buy">Buy Now</button>
                <button class="btn btn-show">Show Deatils</button>
```

1- Si la cadena es vacia mostrará todos los cursos

2- En caso contrario quiere decir que la cadena no esta vacia, accedemos a la propiedad name y la convertimos a minusculas, el método include retorna True en caso de que exista una coincidencia y mostrara los cursos con ese nombre



procademy

Search: java

Filter: ●All Courses(9) ○Free Courses(3) ○Premium Courses(6)

JavaScript for beginners
In this course you will learn the fundamentals of JavaScript. This course is pur...
Free Price: $0

Advance JavaScript Course
In this course you will learn the complete modern JavaScript step by step. This ...
Premium Price: $150

Muestra solo los cursos de javaScript