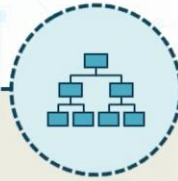


Building a Nested Component



- A component can be used within another component resulting in a nested component, where the outer component is the parent, and the inner component is the child
- Child component uses logic from the parent component
- Nested component is added as a subfolder of the parent component folder
- Nested components are used to pass data to and from the child component



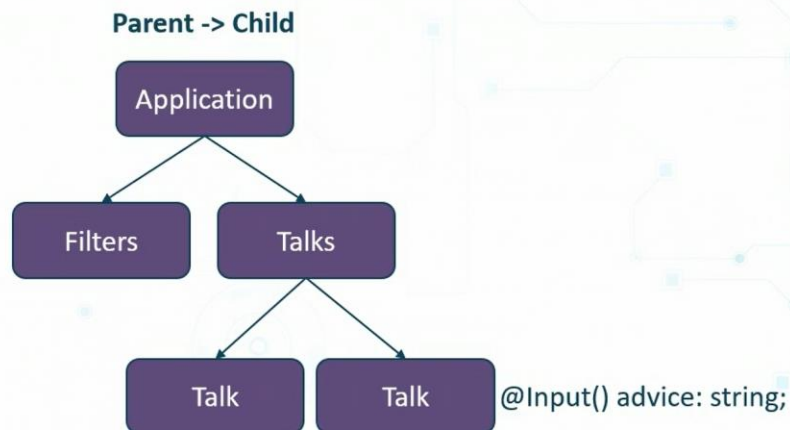
Passing Data to Nested Component Using @Input



- @Input decorator is used for passing data to the nested component
- @Input decorator binds the expression into the directive



Passing Data to Nested Component Using @Input



Moving Data from Nested Component using @Output

@Output:

- For sending data from child to parent component
- Is used if the parent-child hierarchy is small
- Notifies the parent component about the changes

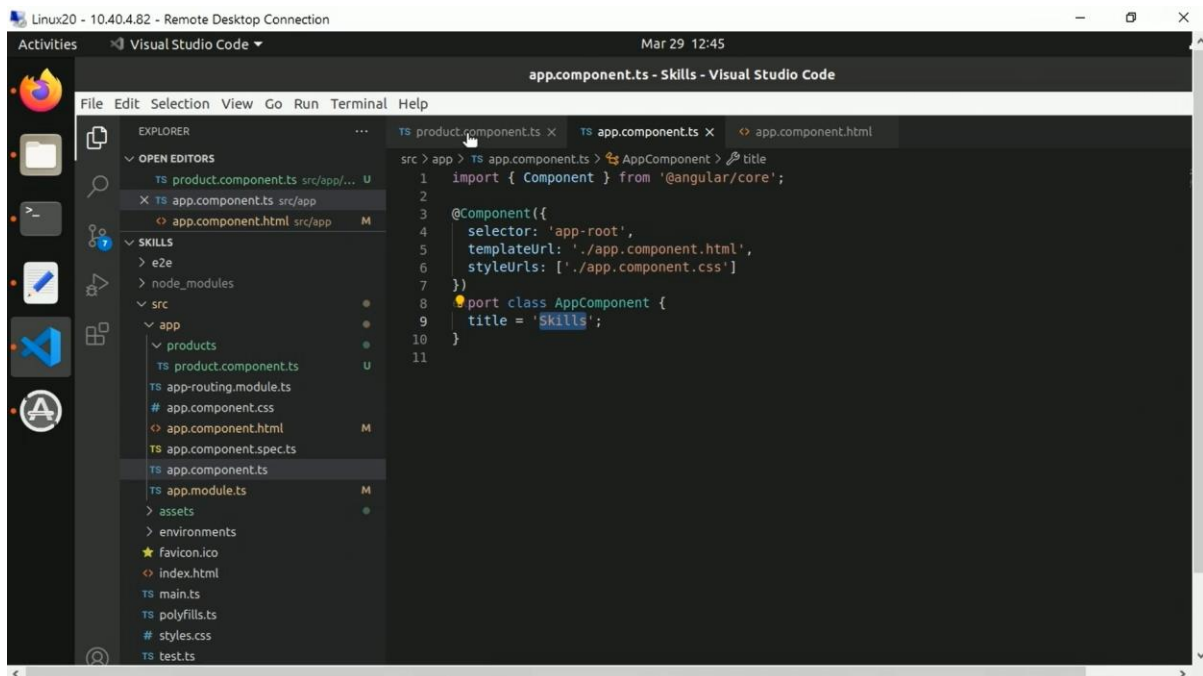
@EventEmitter:

- Is used to emit data from the child component back to the parent component
- Emits an event up to a parent component using event binding

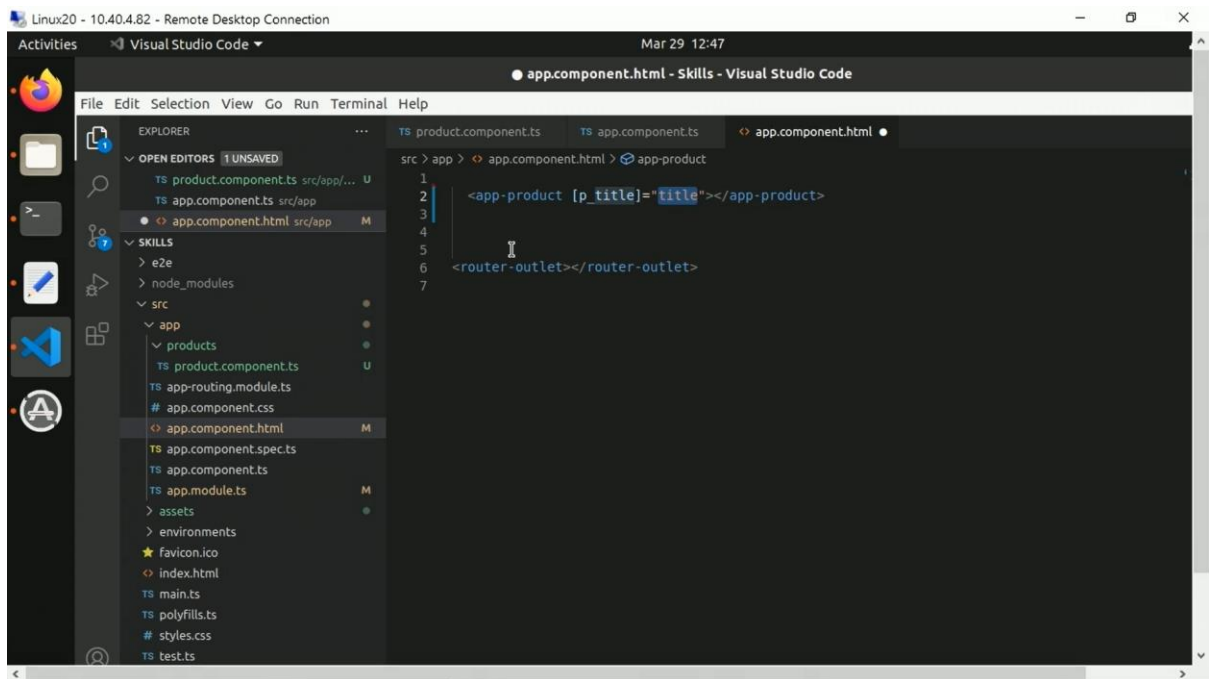
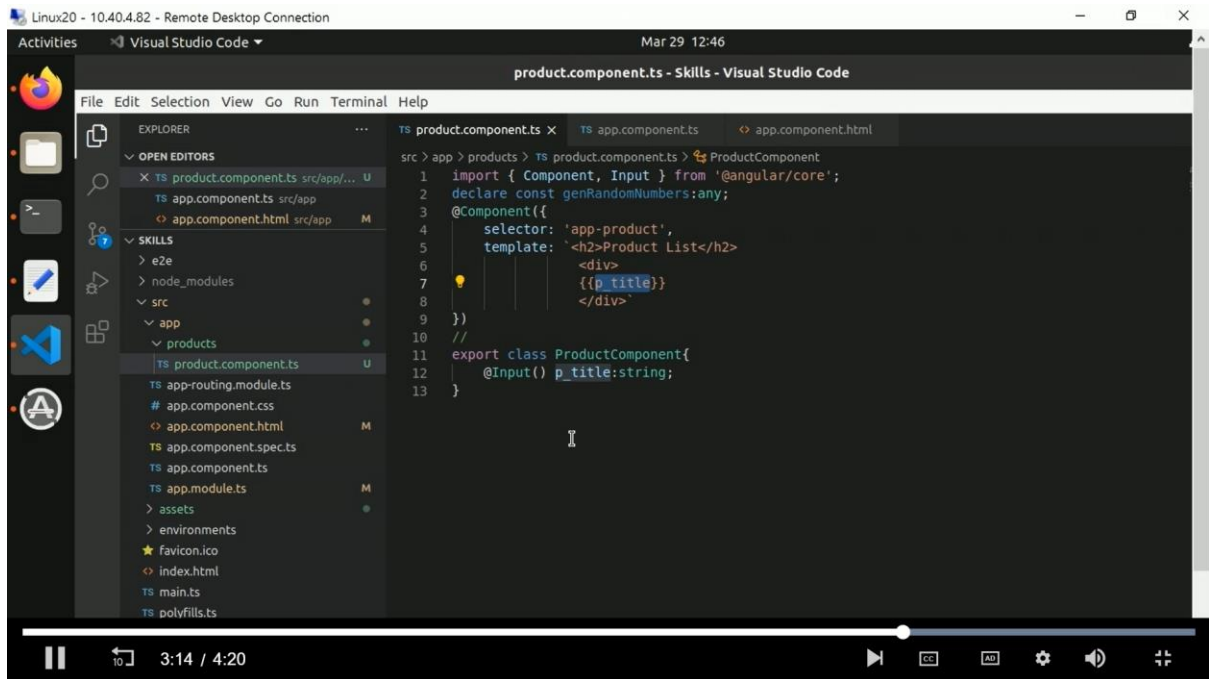
ViewChild in Angular

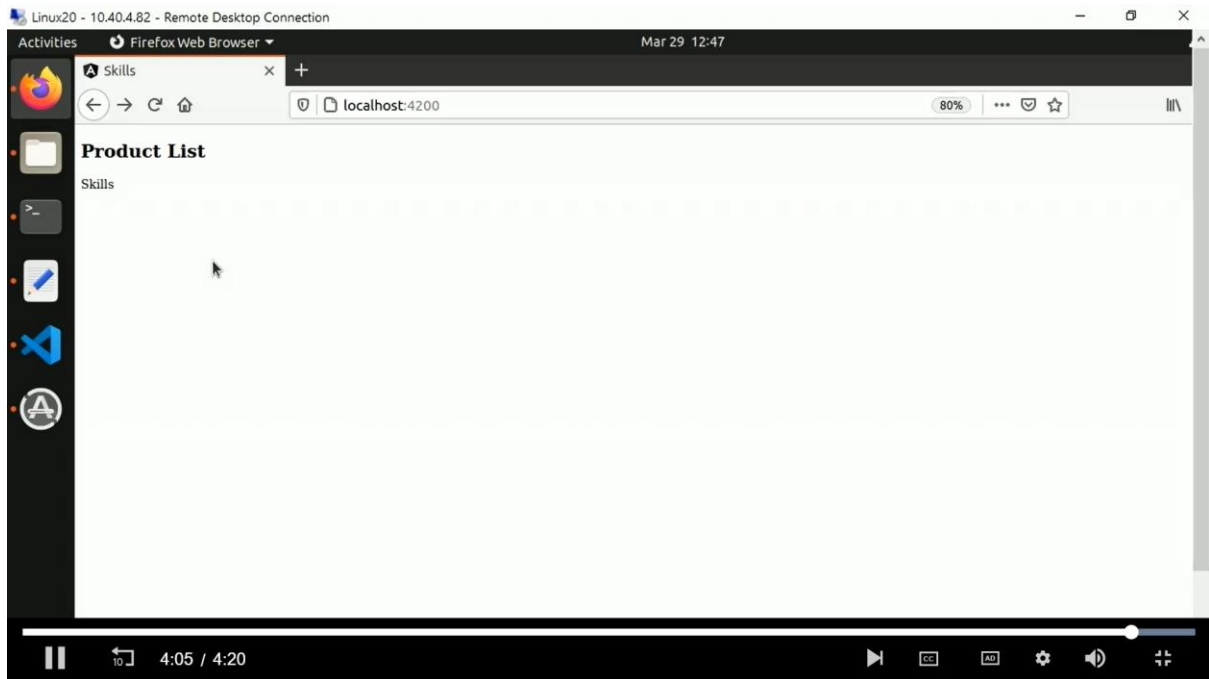
- Used for configuring a view query
- Used to select a specific element from the view
- Requires inputs like selector, read, and static
- Syntax:
 - @ViewChild(selector {read: readValue, static: staticValue}) property;

Passing data to the nested components using @Input

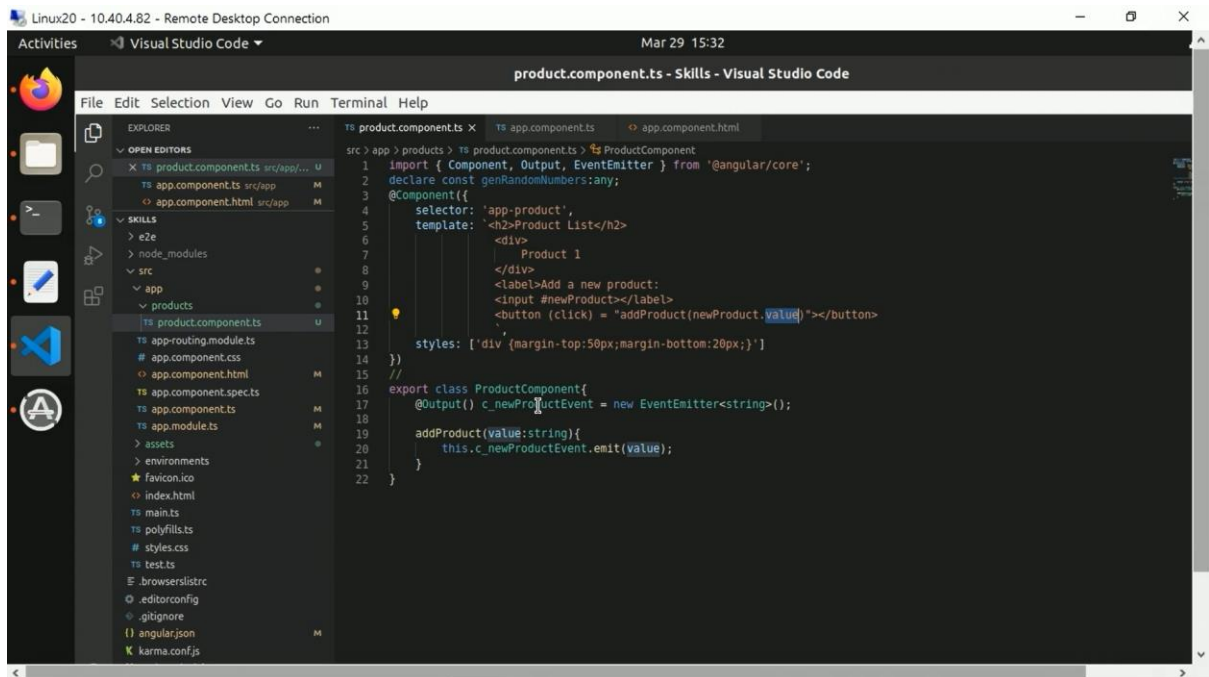


Product.com.ts as child

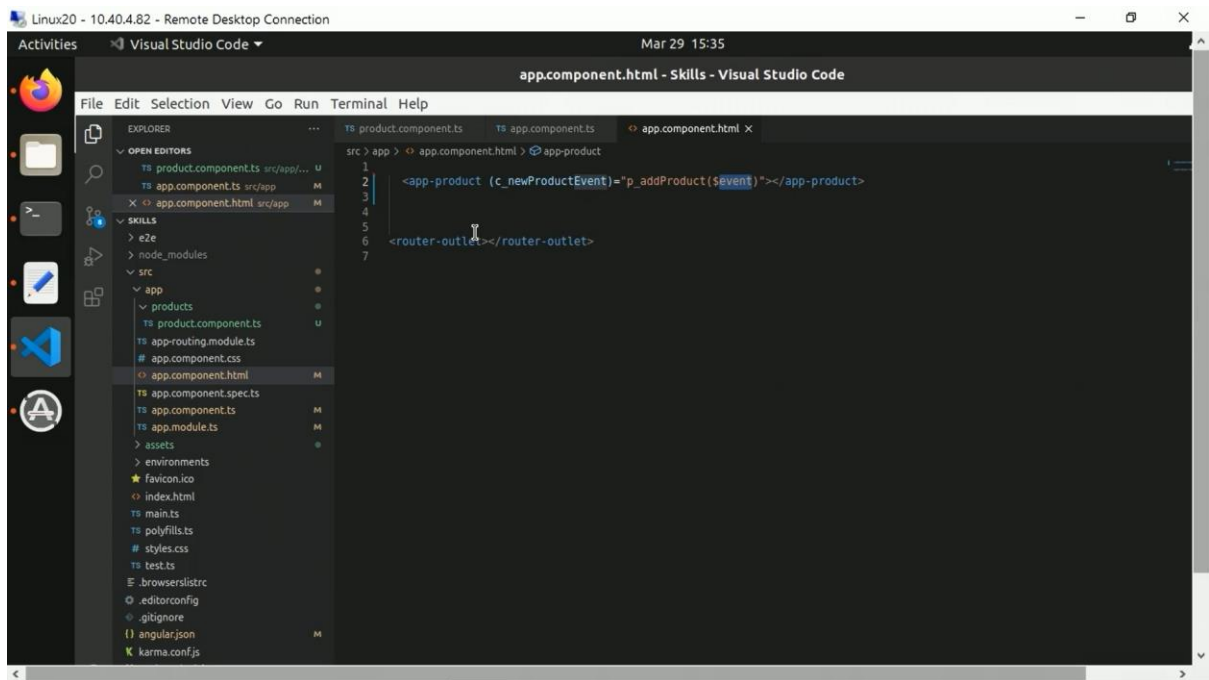
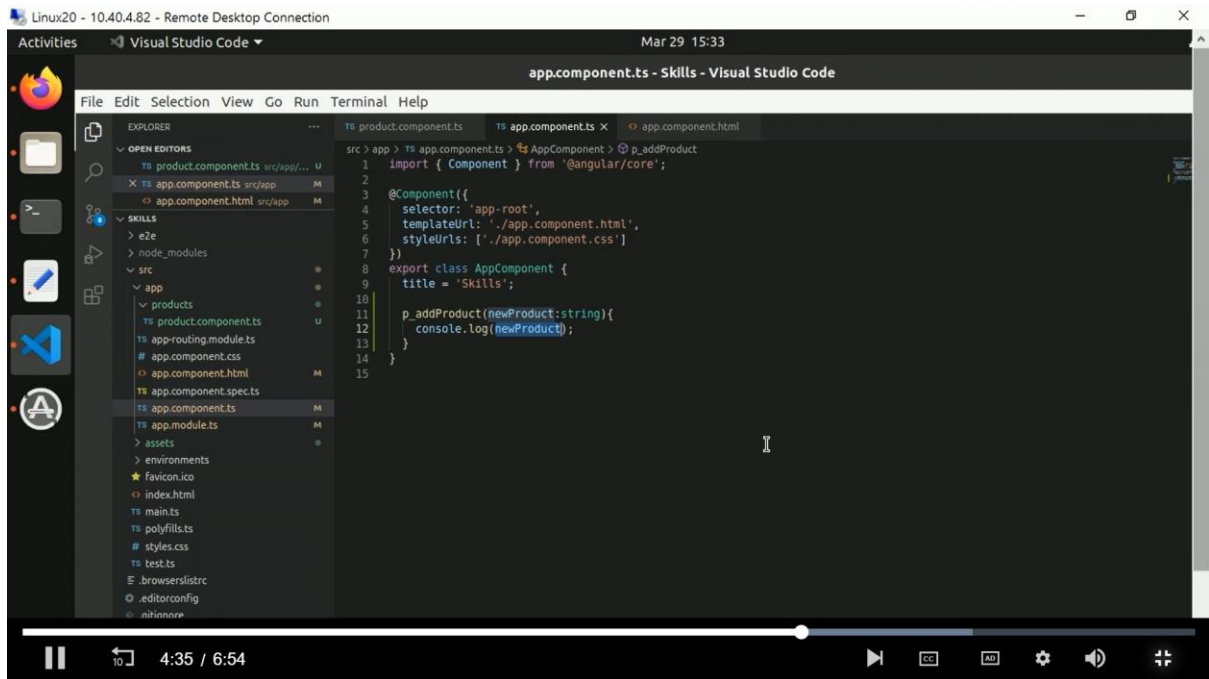


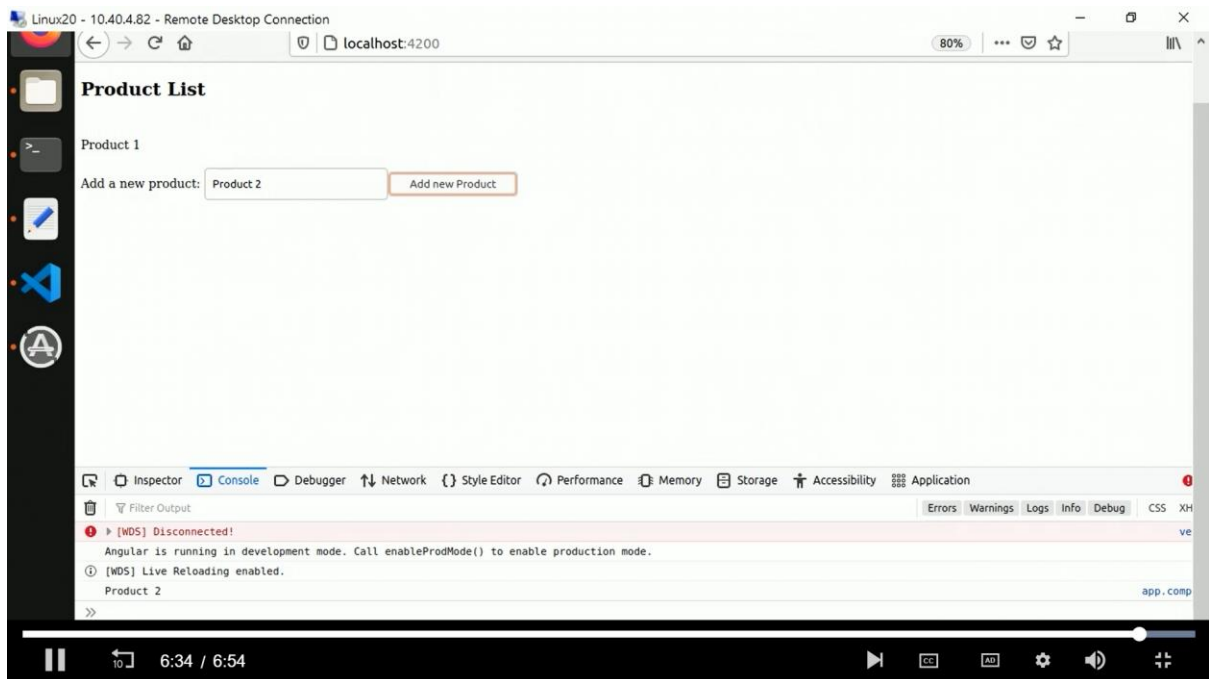
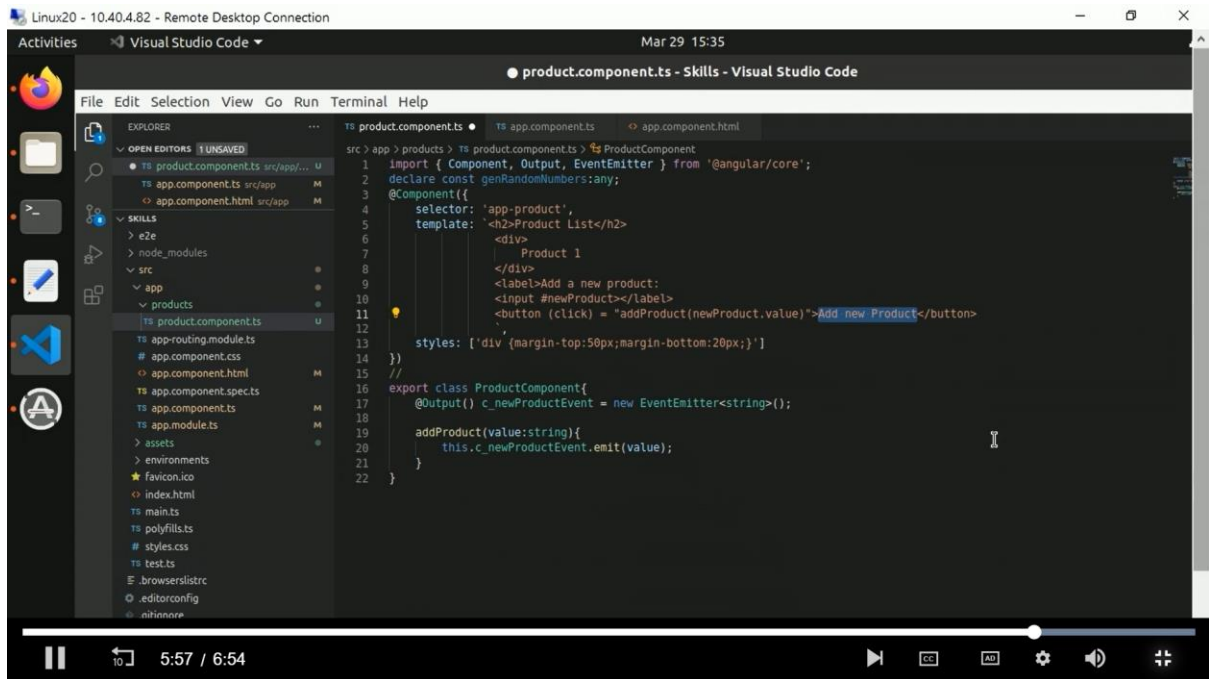


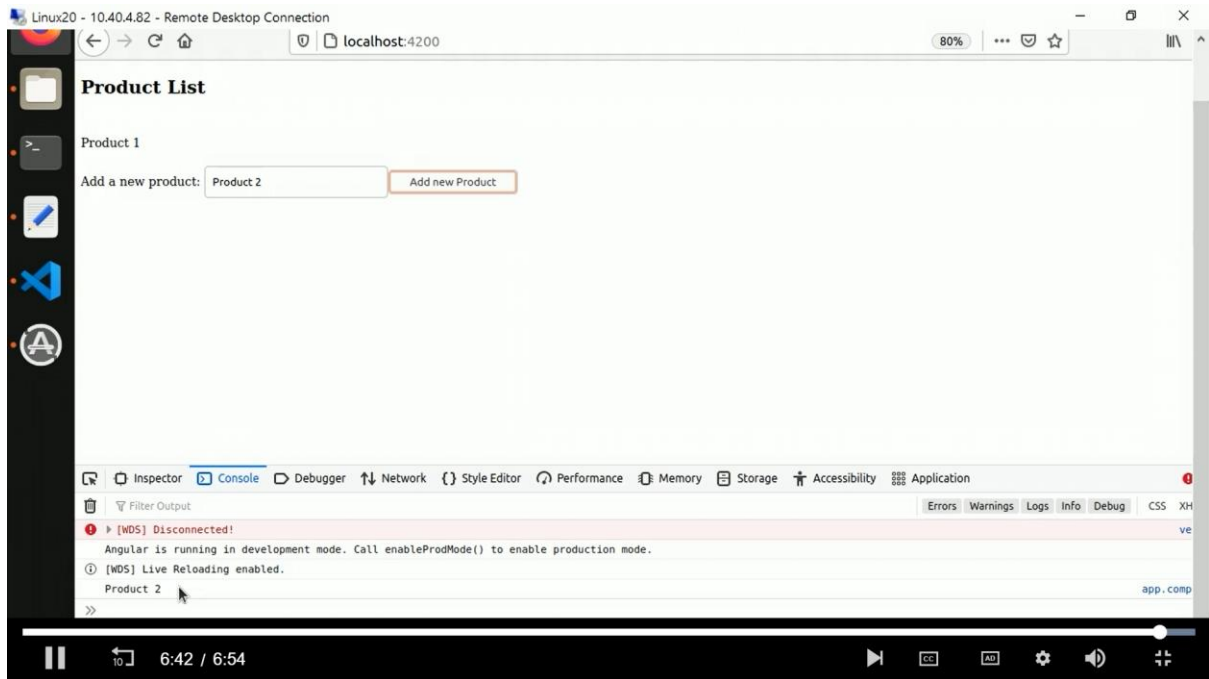
Moving data from nested component using @Output
child



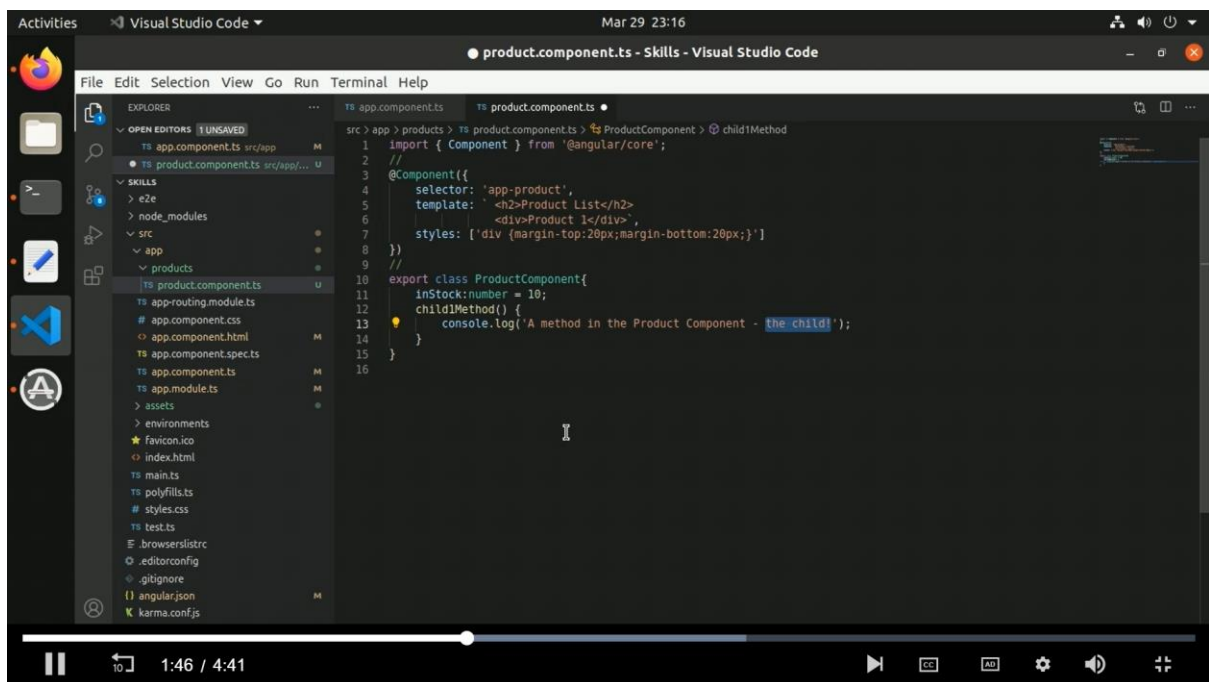
Parent com

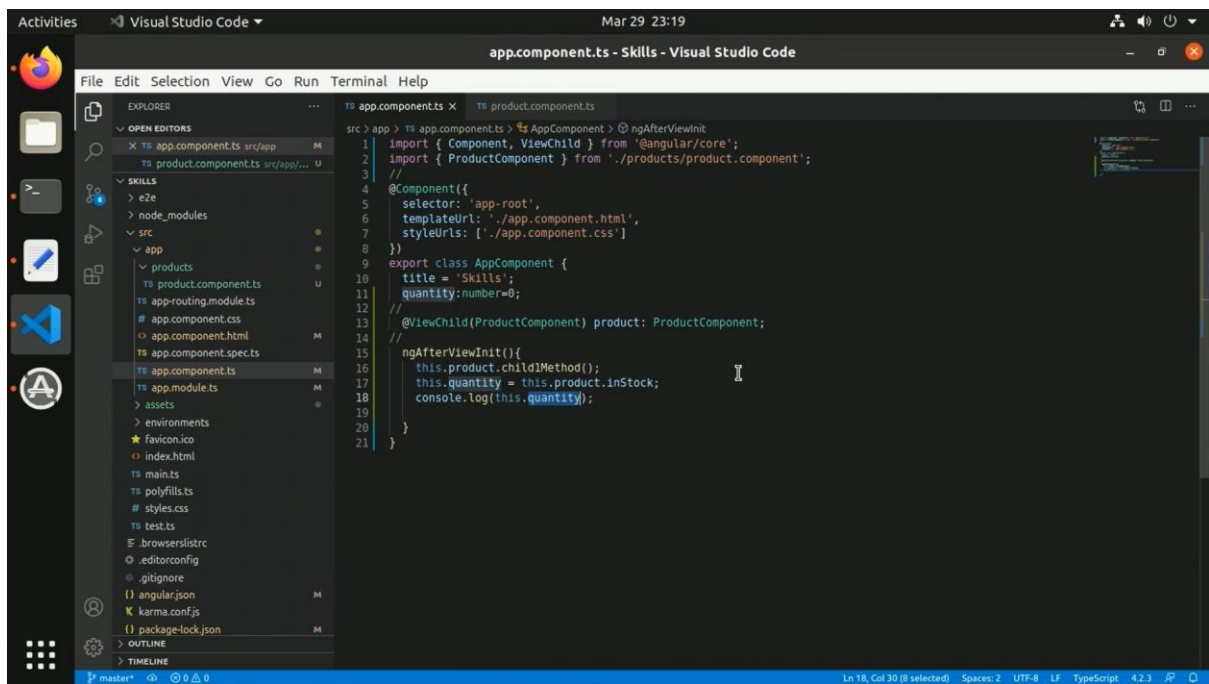




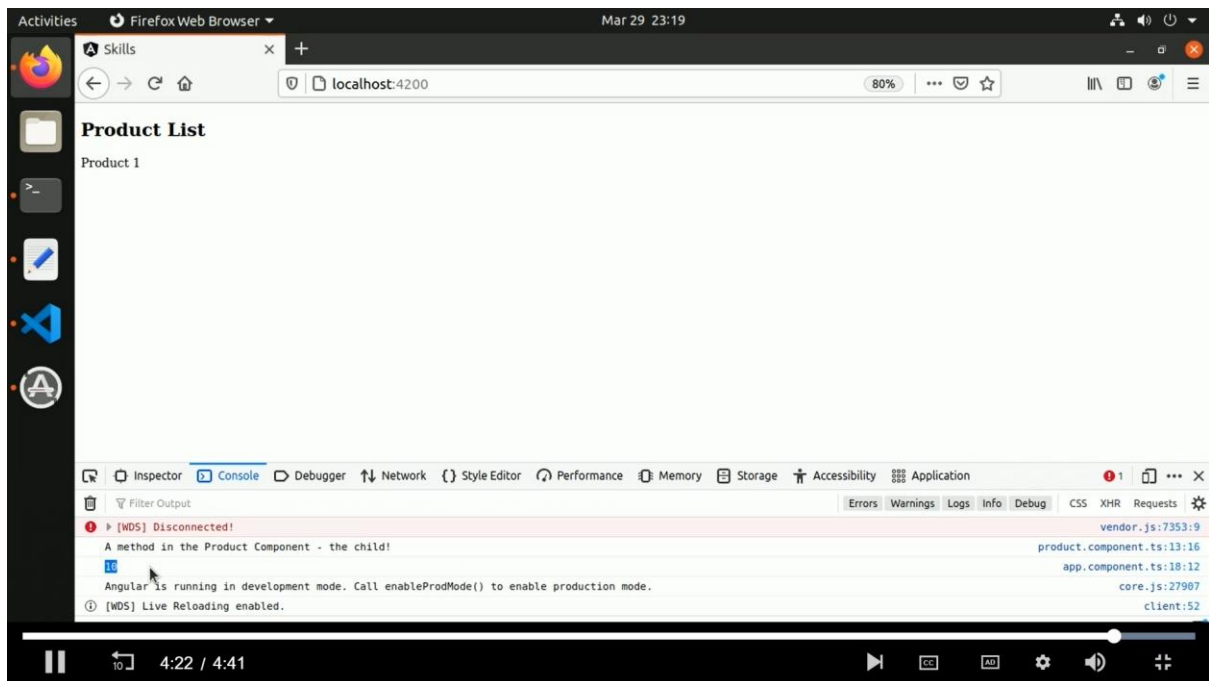


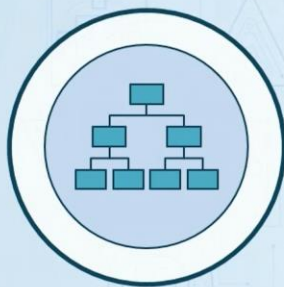
Implementing View Child In Angular Components





```
src > app > ts app.component.ts > AppComponent > @ ngAfterViewInit
1 import { Component, ViewChild } from '@angular/core';
2 import { ProductComponent } from './products/product.component';
3 //
4 @Component({
5   selector: 'app-root',
6   templateUrl: './app.component.html',
7   styleUrls: ['./app.component.css']
8 })
9 export class AppComponent {
10   title = 'Skills';
11   quantity:number=0;
12 //
13 @ViewChild(ProductComponent) product: ProductComponent;
14 //
15 ngAfterViewInit(){
16   this.product.childMethod();
17   this.quantity = this.product.inStock;
18   console.log(this.quantity);
19 }
20
21 }
```





Building a nested component

Passing data to nested components using @Input

Moving data from nested components using @Output

Implementing ViewChild in Angular

Coming Up Next . . .



The RxJS library



HTTP requests

Next Up
Reflect on what you've learned



1:28 / 1:32

