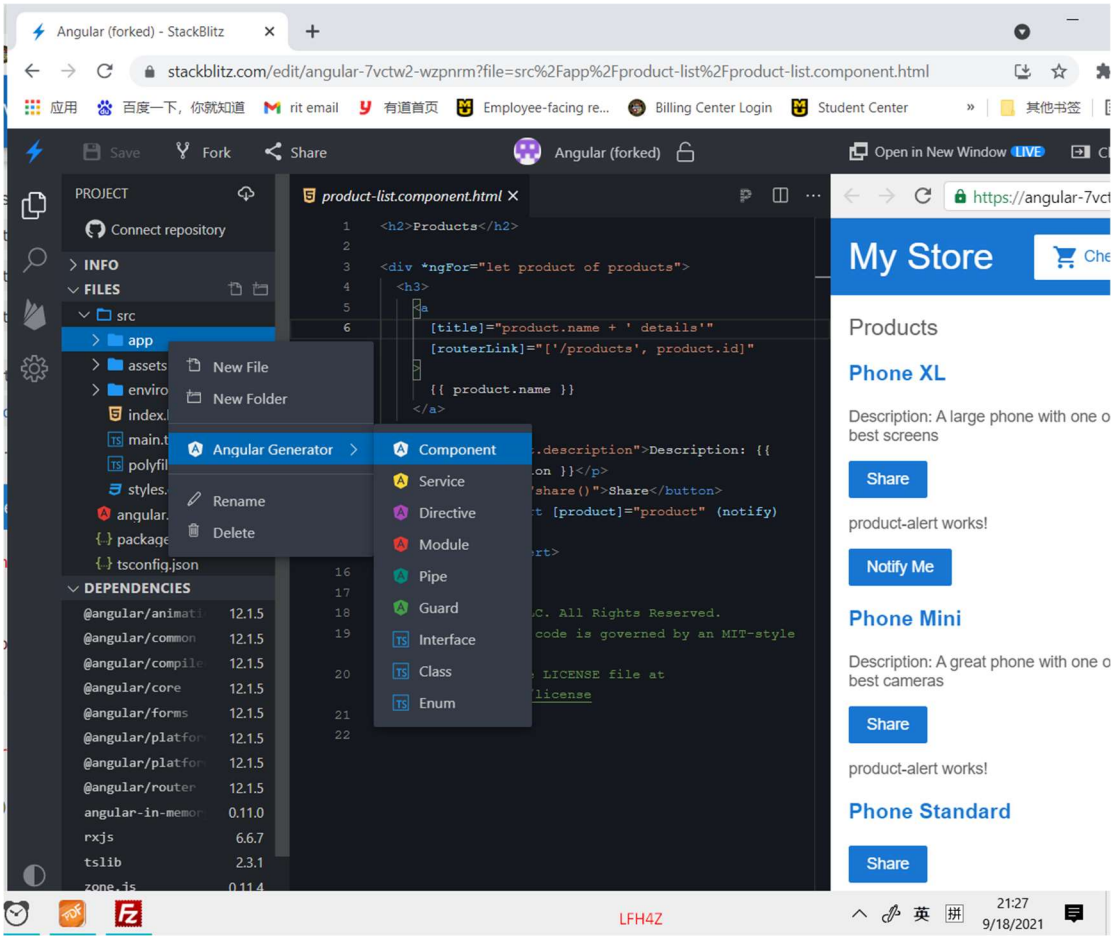
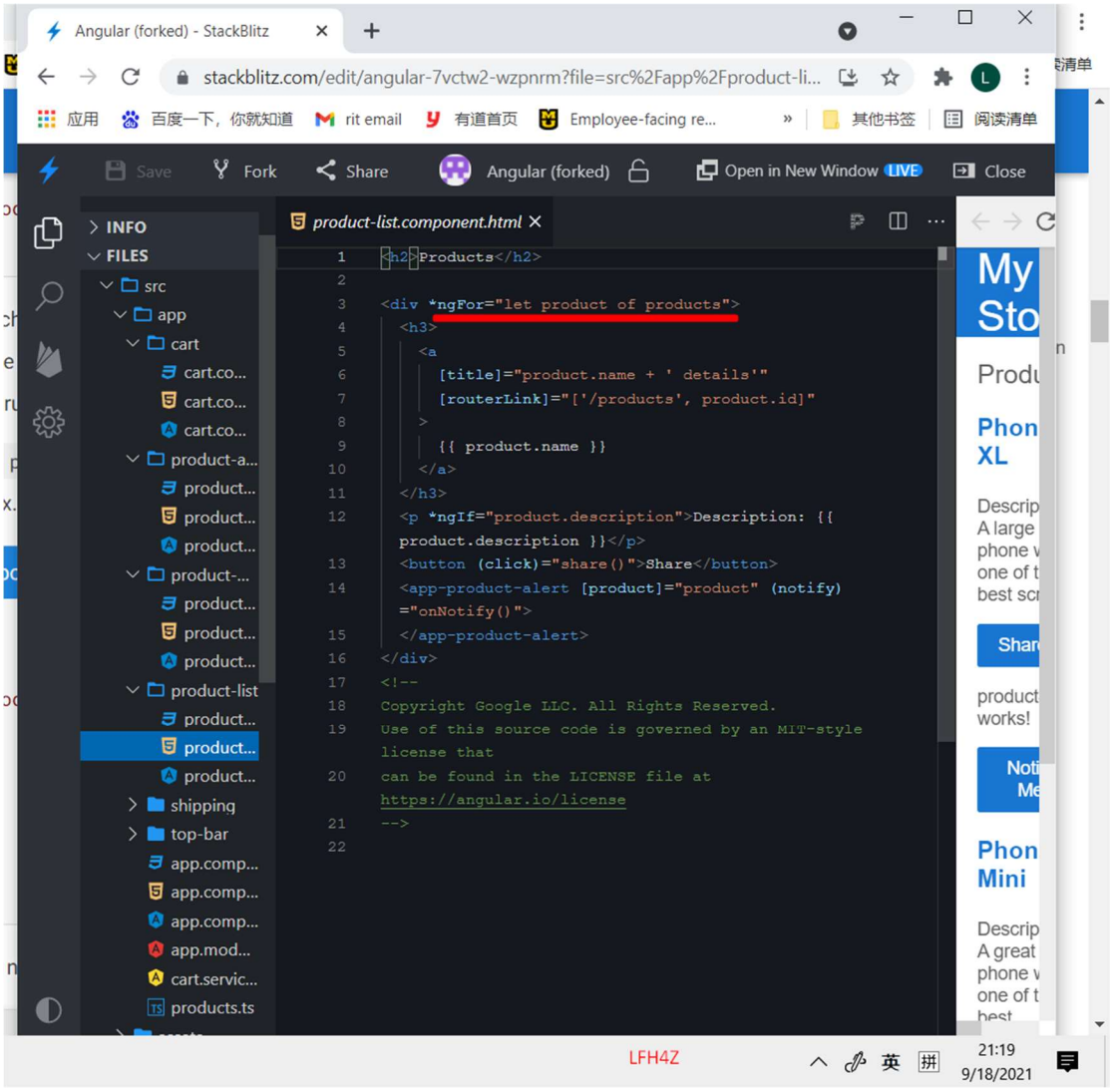
1 Create Product List

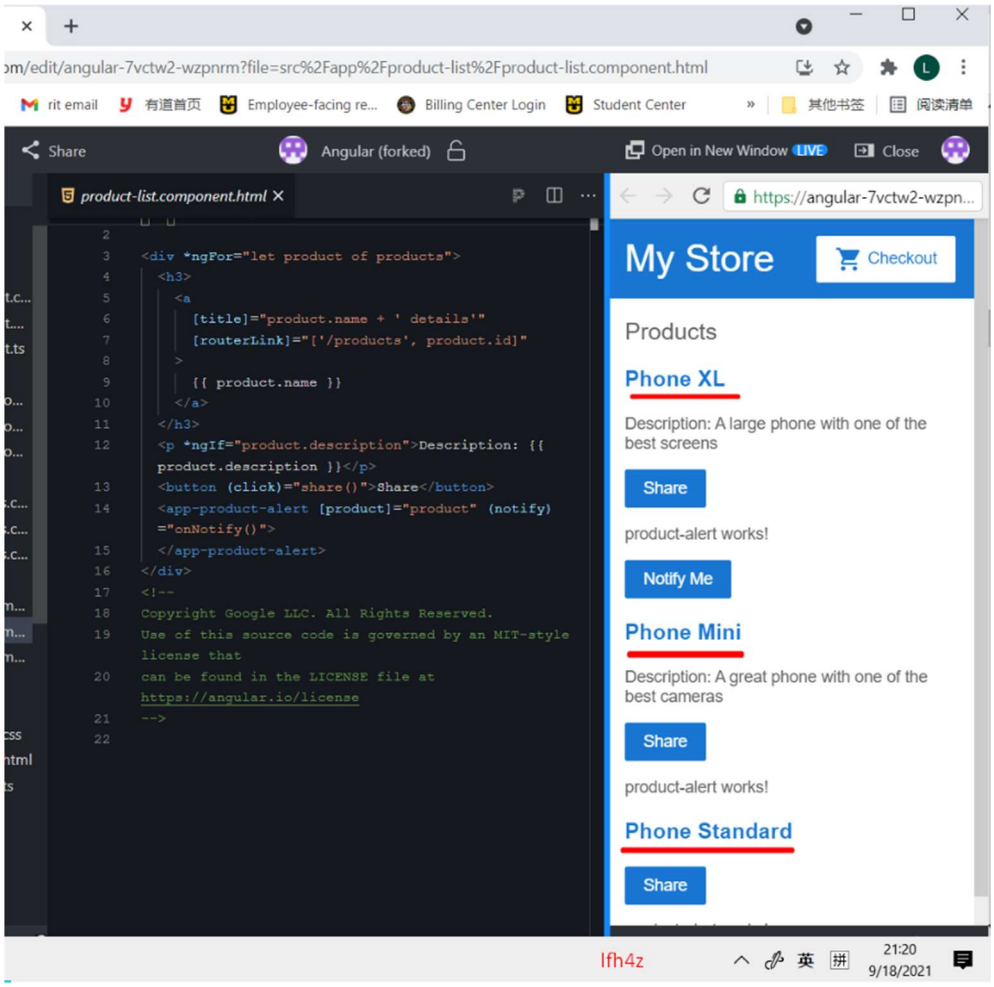
1.0 Set Up the Product List file

1) Create the component of the product list

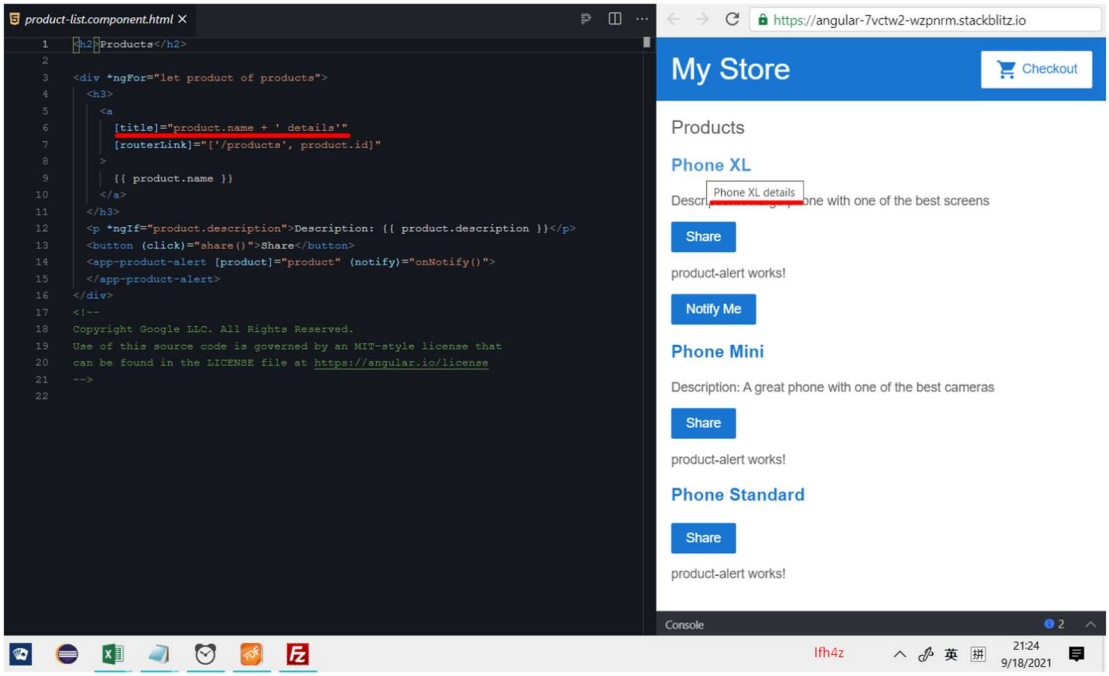


2)Add *ngFor statement to the following content so that the product object in the product.ts file can be gone through and shown on the web page

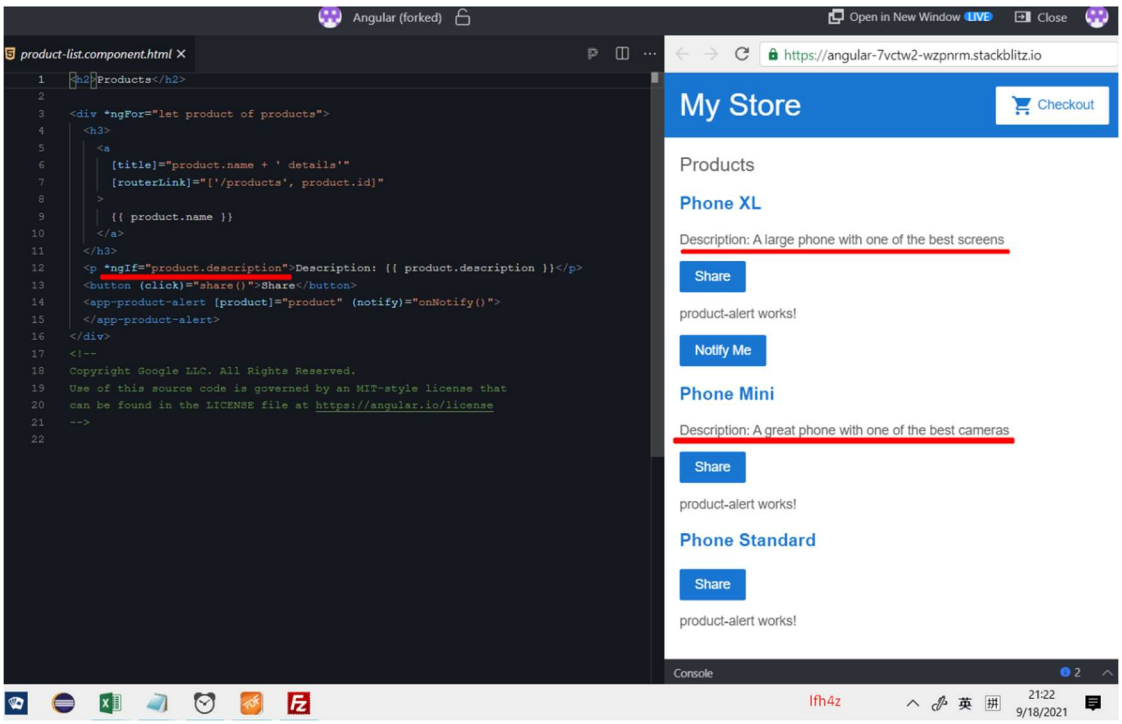




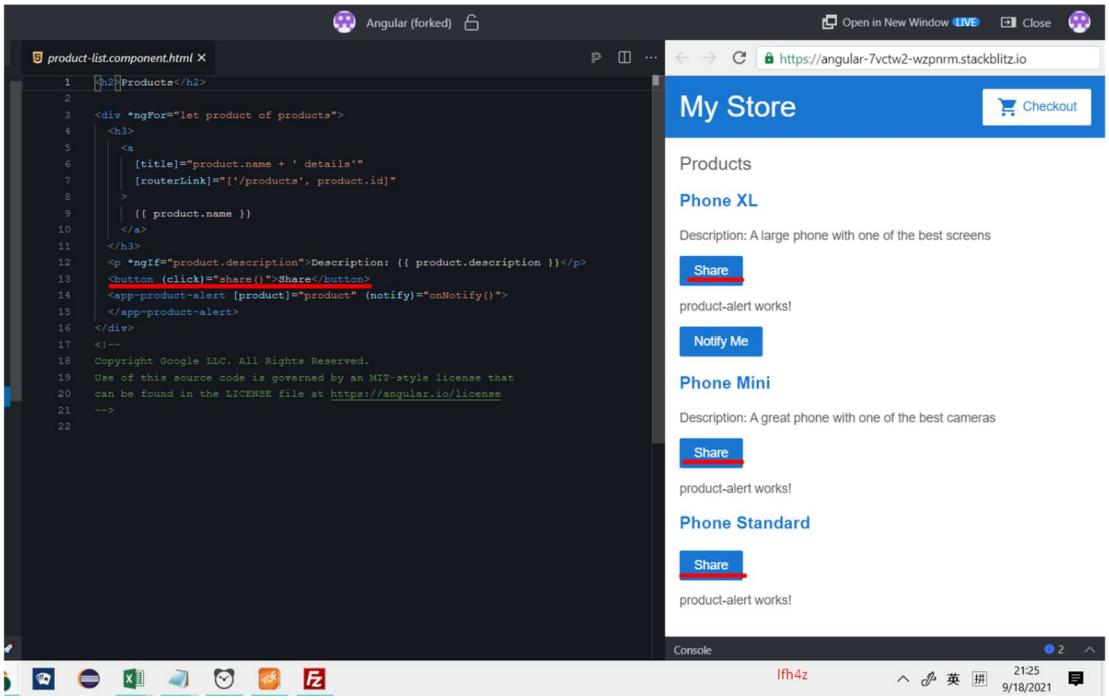
3) Add the following statement so that it will show the information of title when the mouse points to the product name.



4)Add the *ngIf statement so that the object of the product.ts will be gone through. If the object has description variable, the statement of the html file will be executed and the description information will be shown on the product list web page.

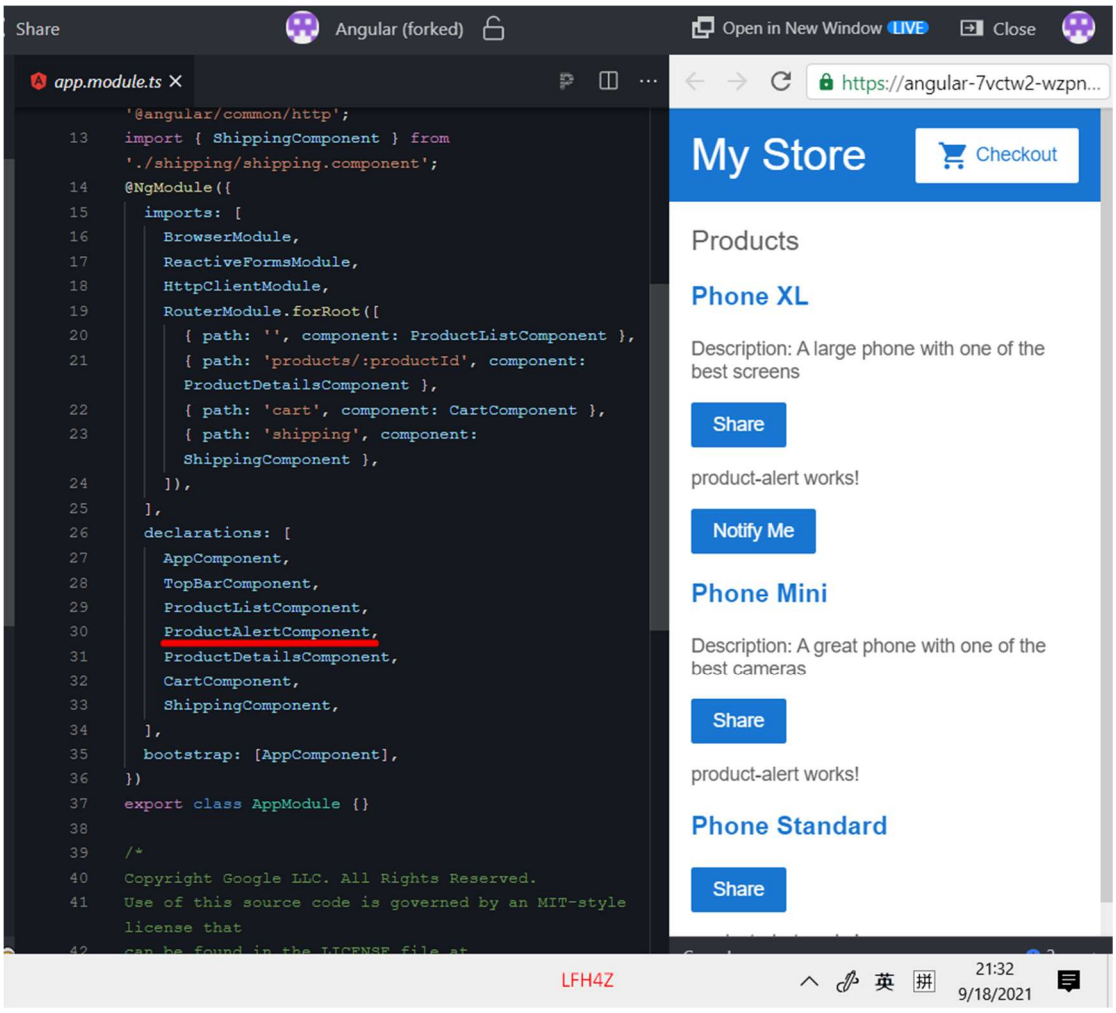


5)Add the button tag to the html file. This file will call on the share() function which defined on the produc-list.component.ts file.

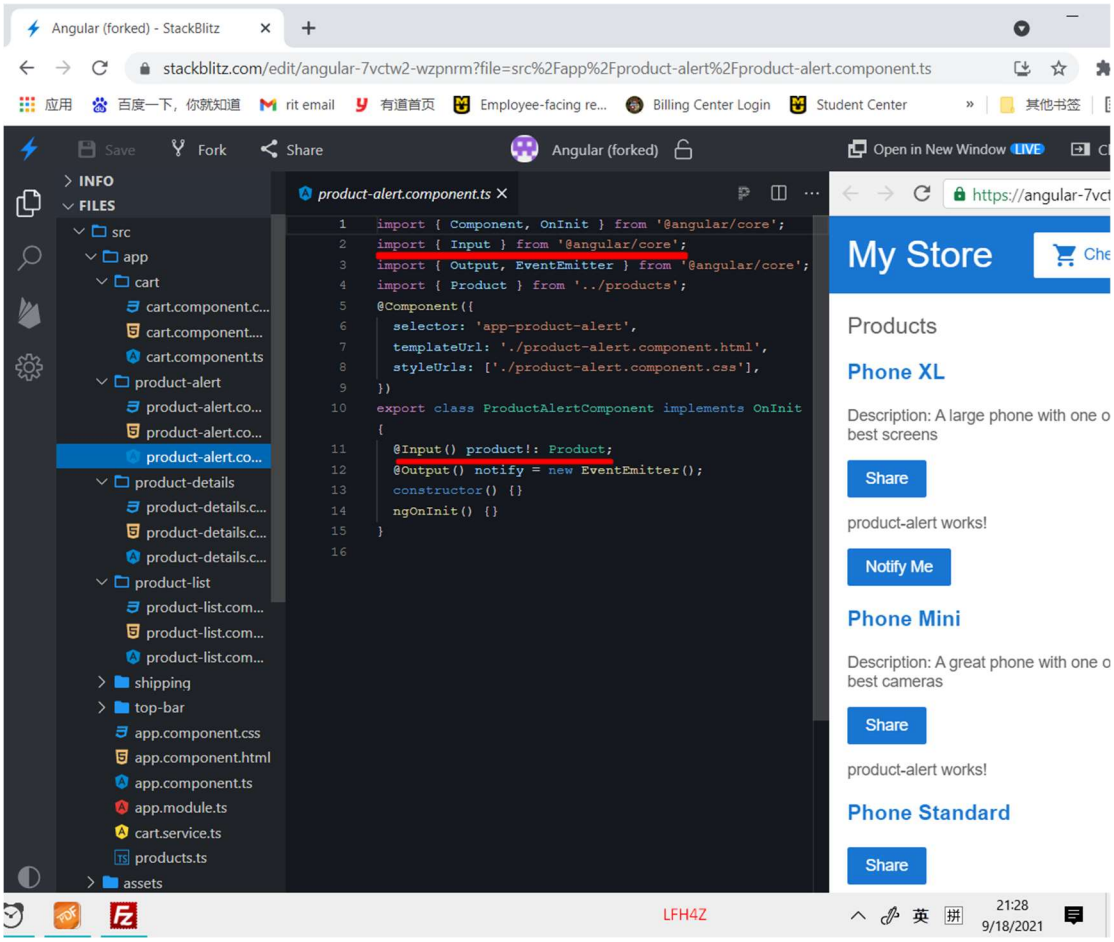


1.1 Make Use of Child Component

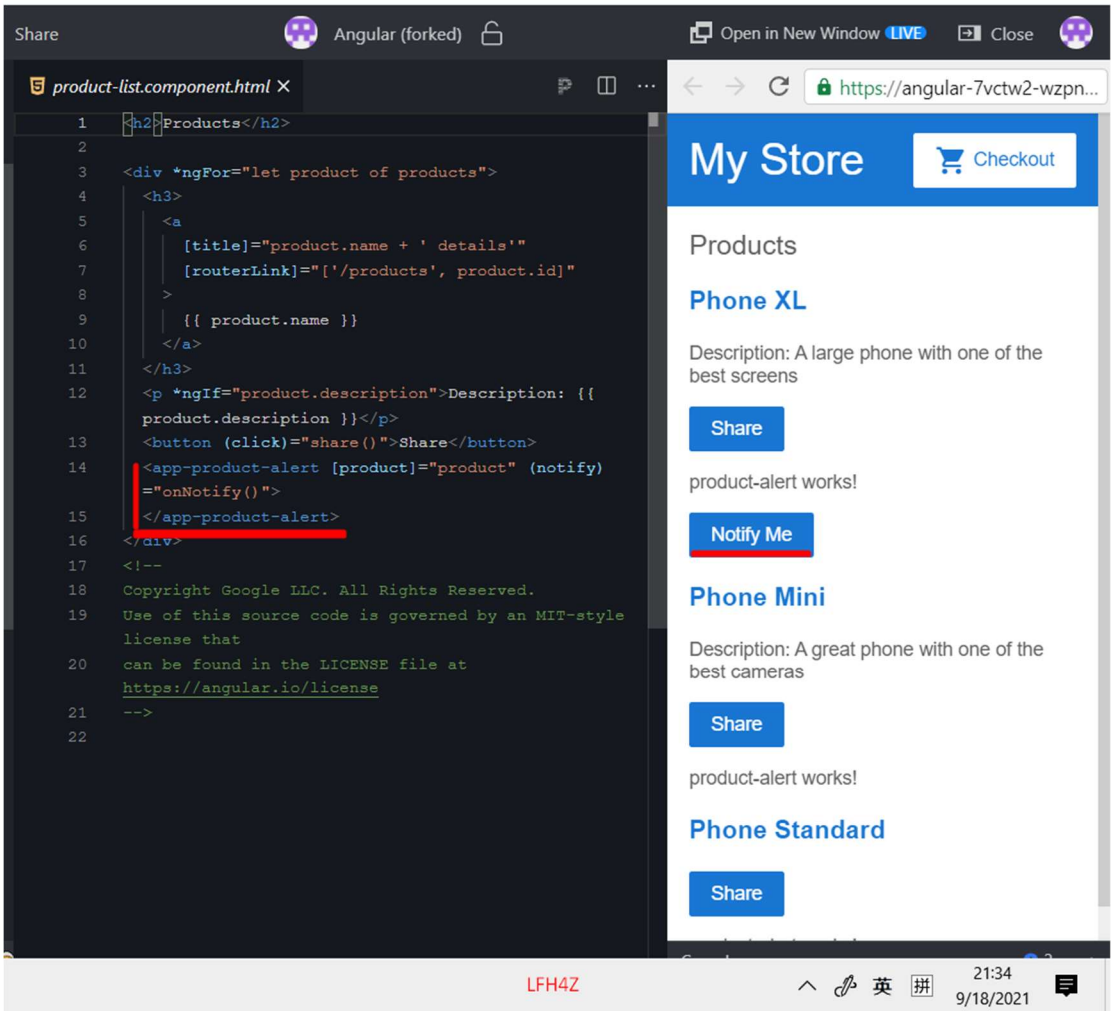
1)Create a product alert component and add the component to the app.module.ts file



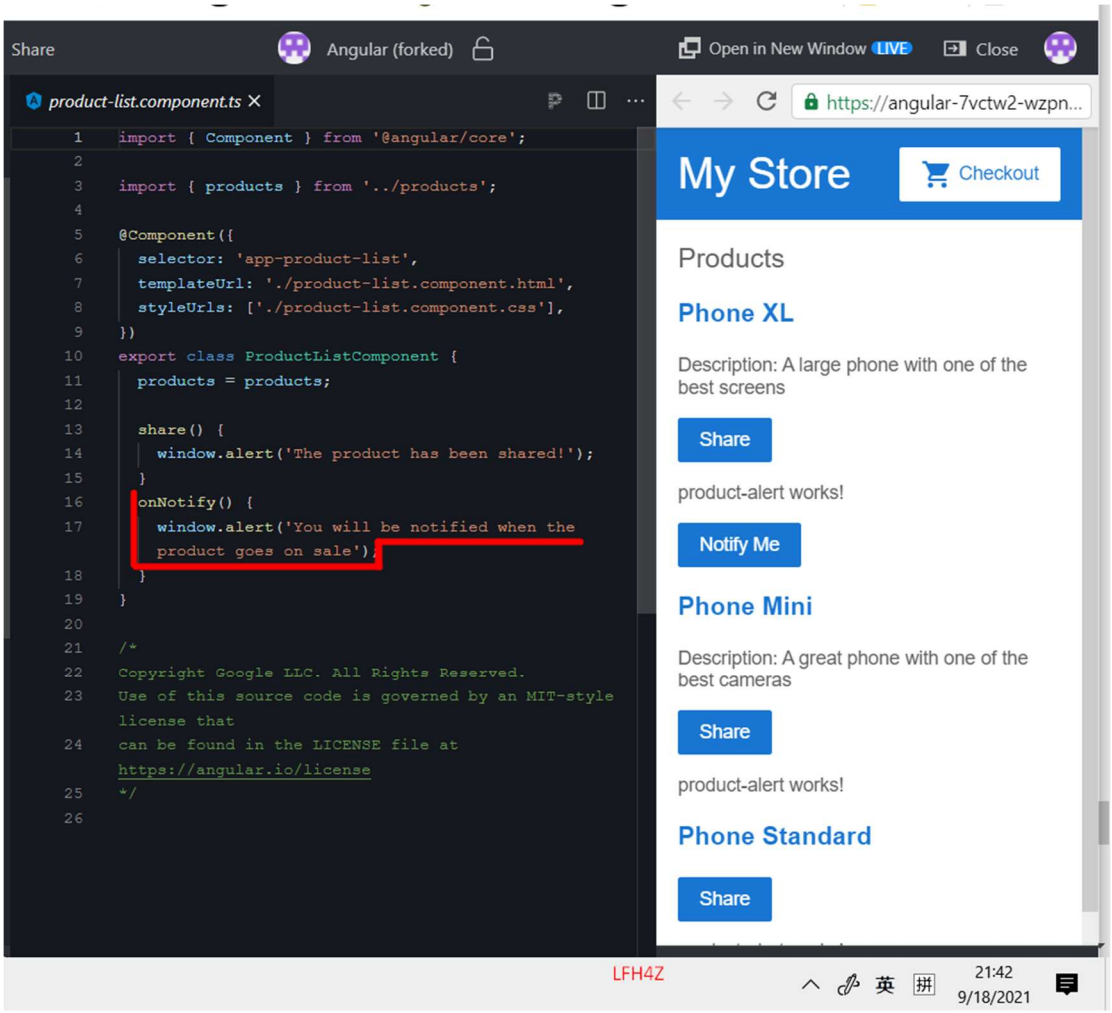
2) Add Input class from the @angular/core. This input object will receive the input object from the parent component which make use of this component.



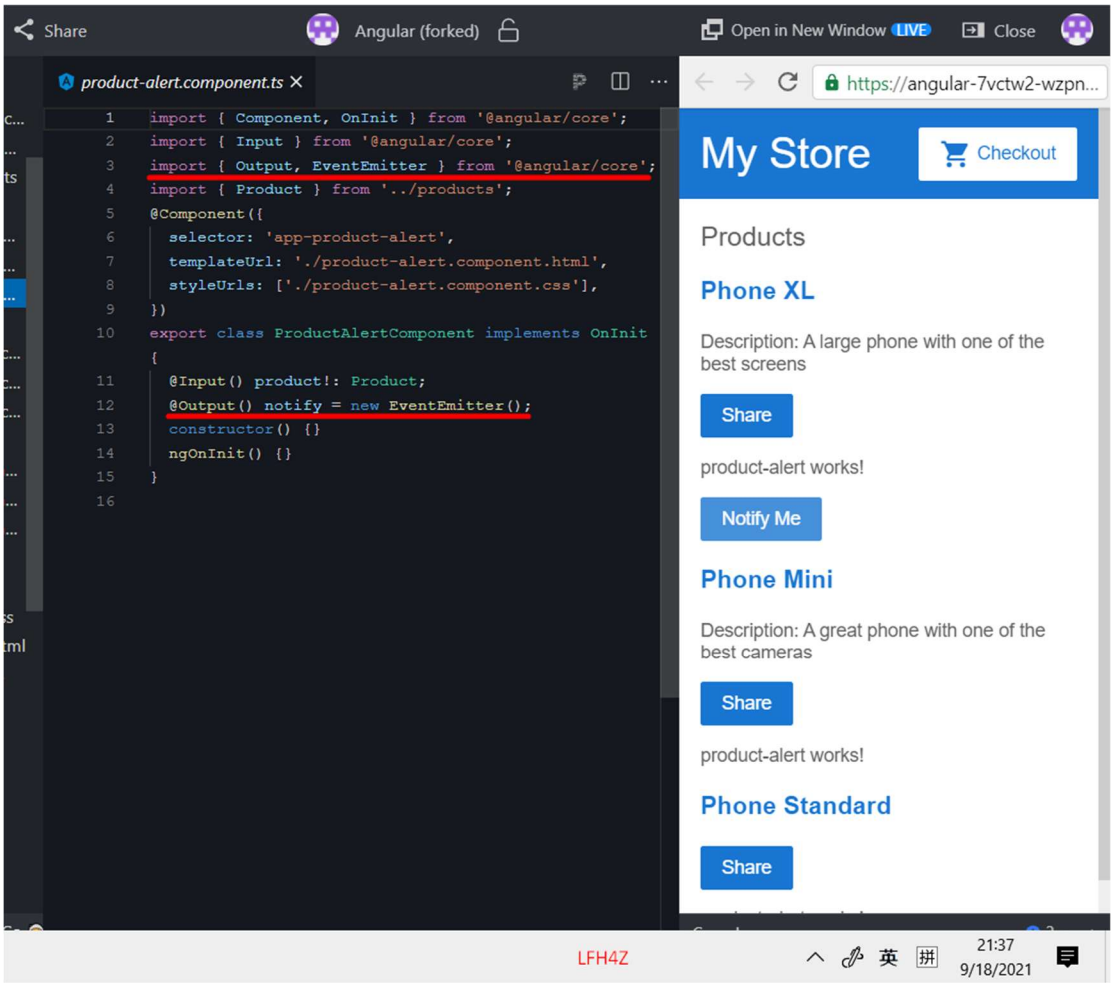
3)Add the product alert component to the product list web page. The [product]="product" defines the object to be sent to the product-alert component. The (notify)="onNotify()" means when the notify object sent from the child component is received, the parent component will call the onNotify() function.



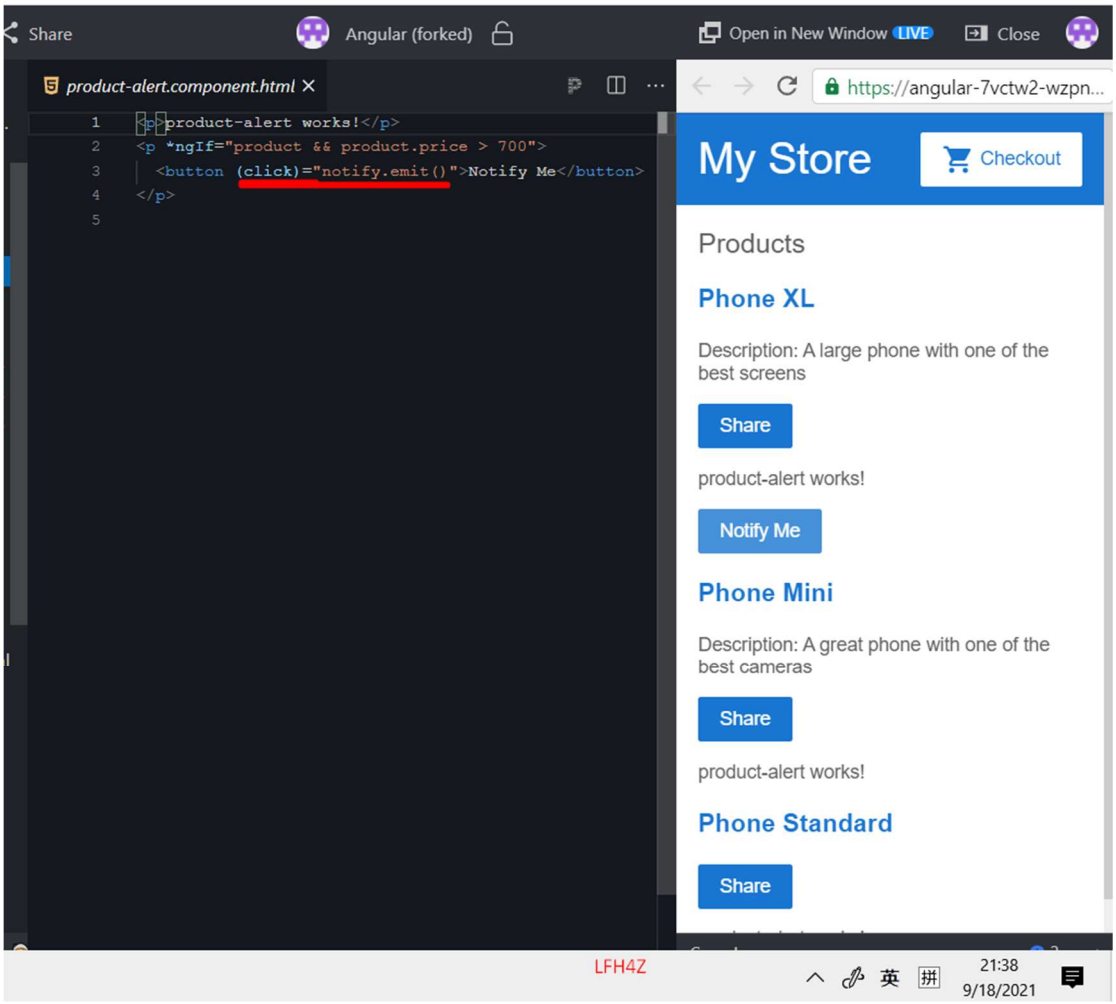
4) At the product-list.component.ts file, add the onNofity() function to verify the successful execution when the child component is called on.



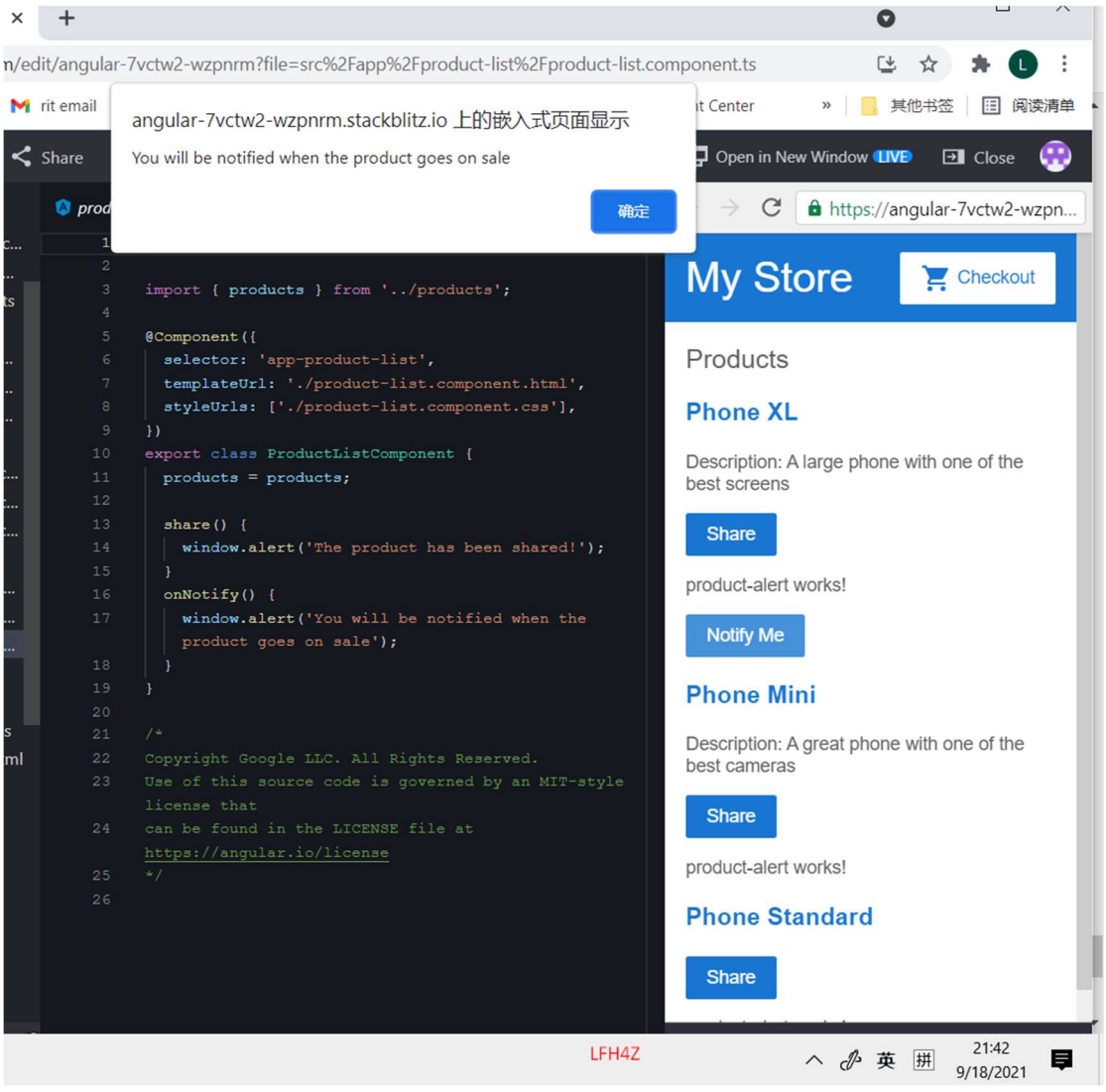
5) Add EventEmitter to the product-alert.ts file so that the component can use the object of the EventEmitter to notify the parent component



6) Add the content to the product alert html file. When the parent comopnent call the child component, it will show on the webpage of the parent component.

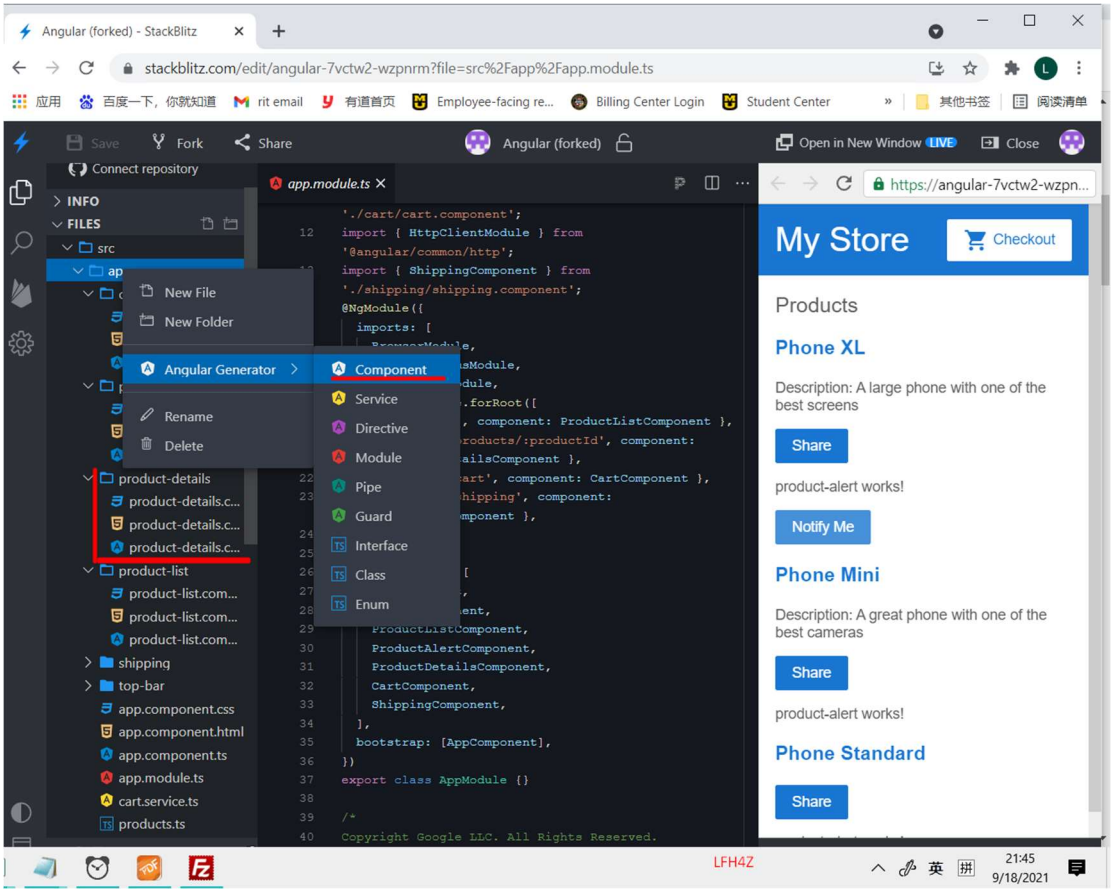


7) Verification

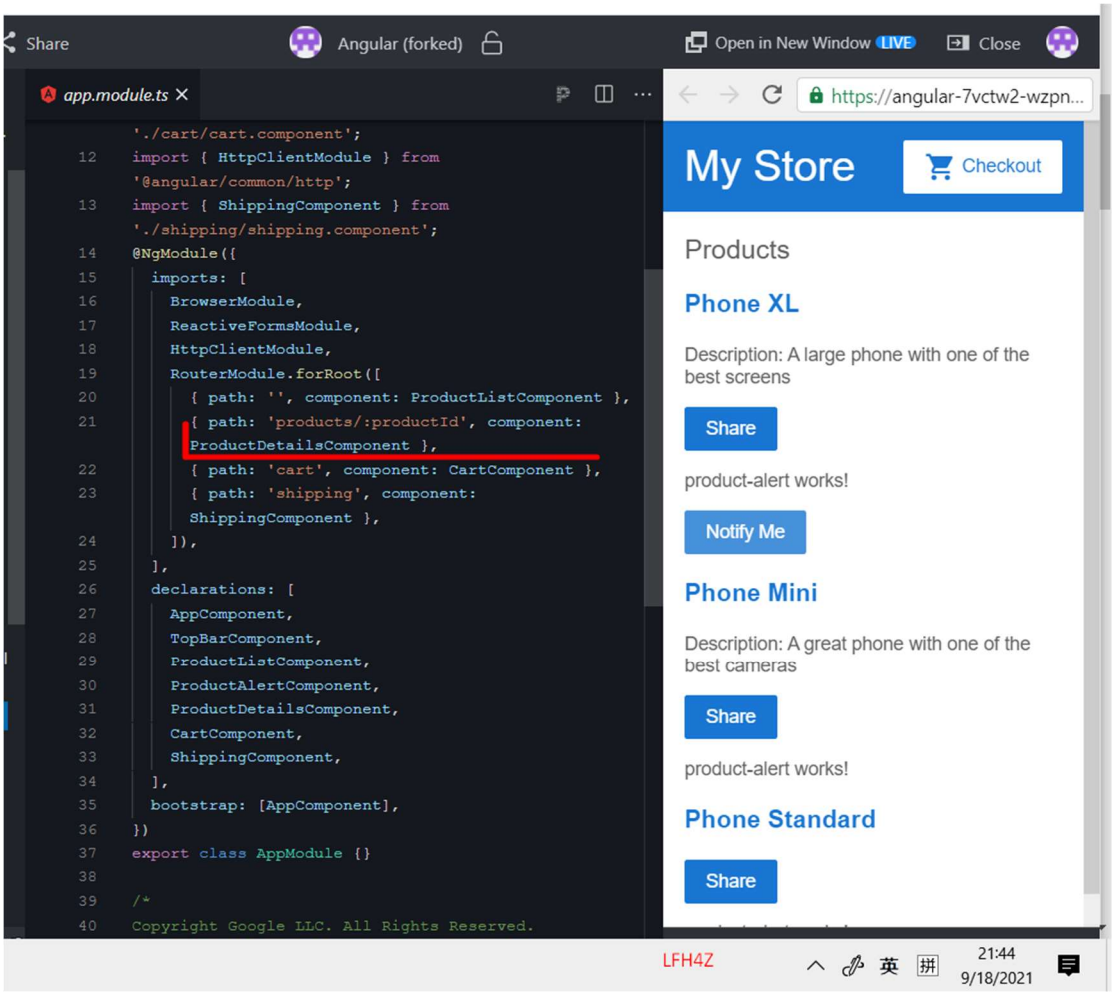


2 Navigate From One Page to Another

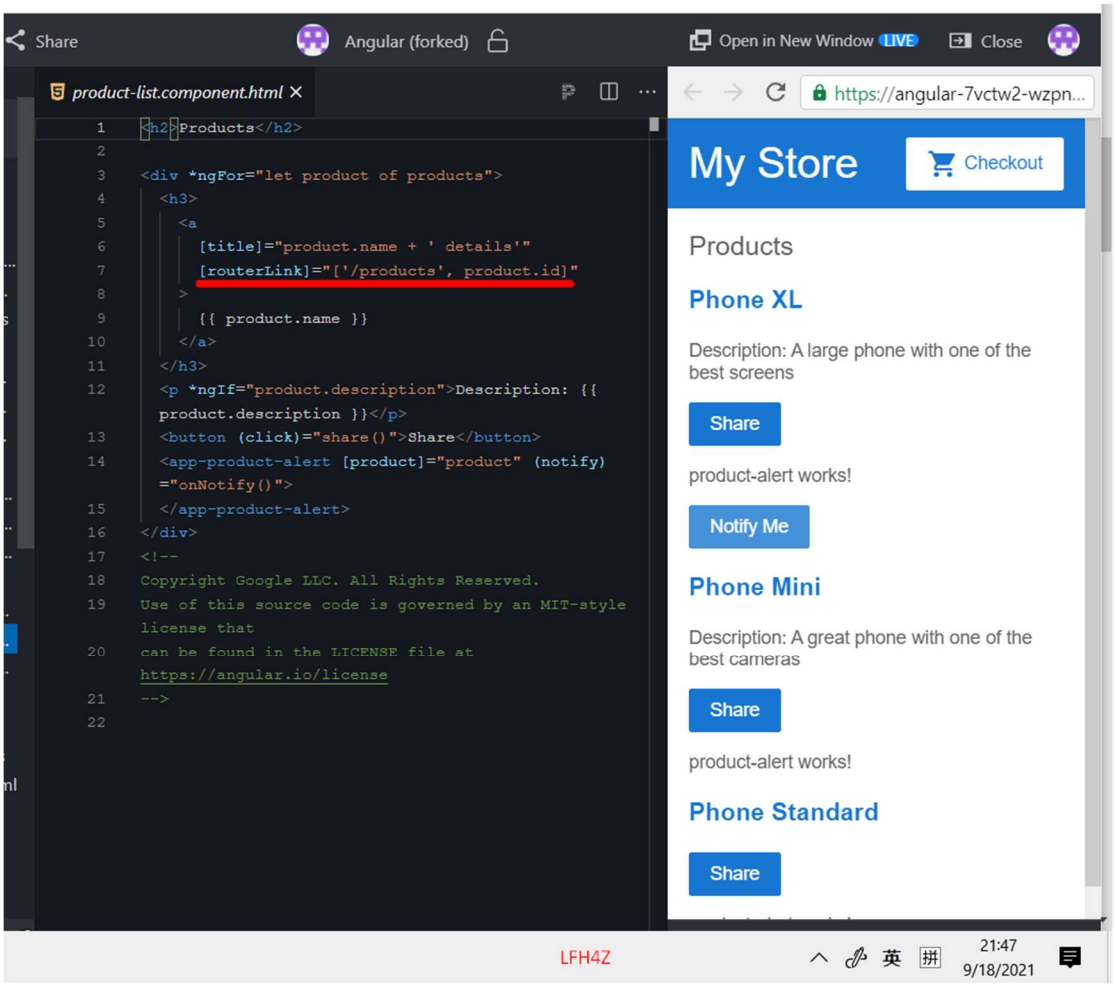
1) Create the product detail component to show the detail when clicking the items.



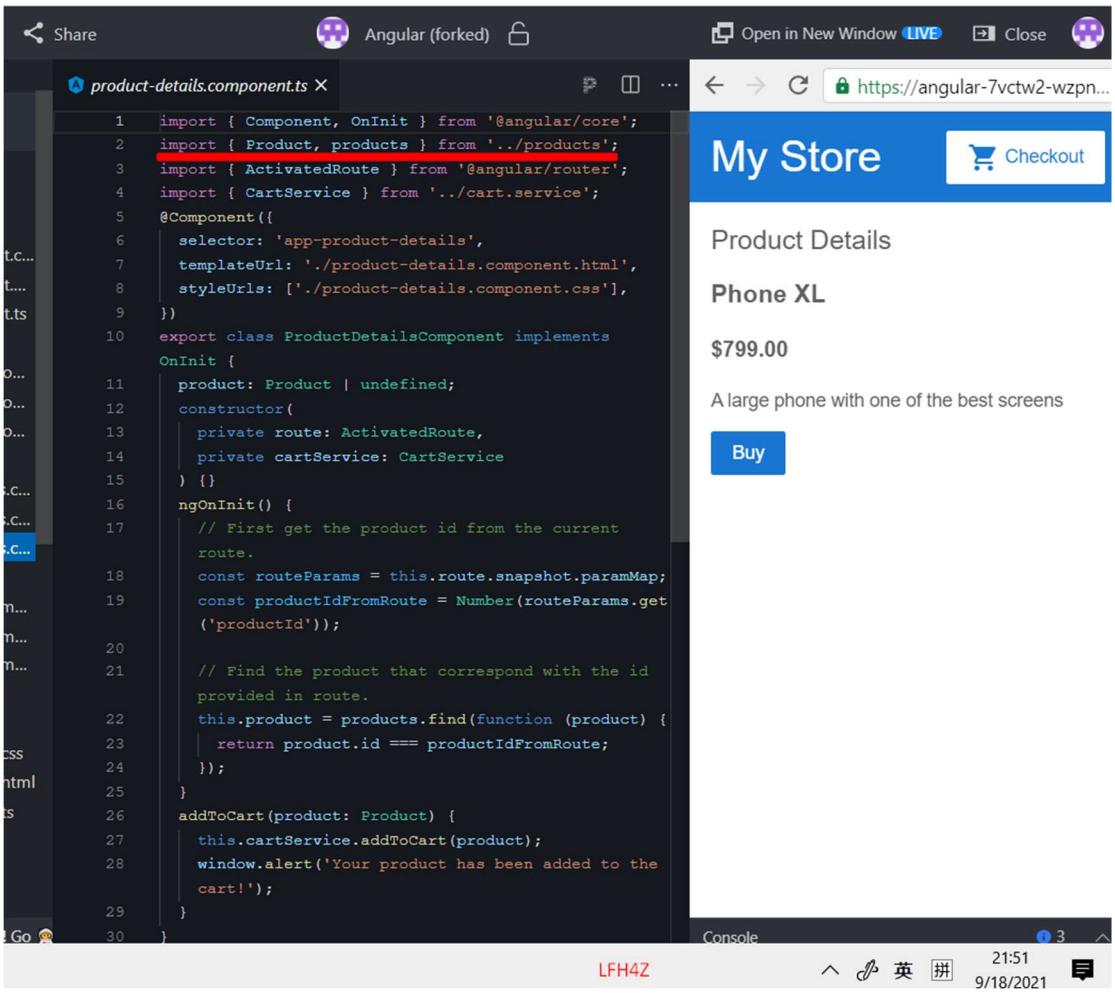
2) Add the component to the app.module.ts so that the component can be found and add the path to the app.module.ts file.



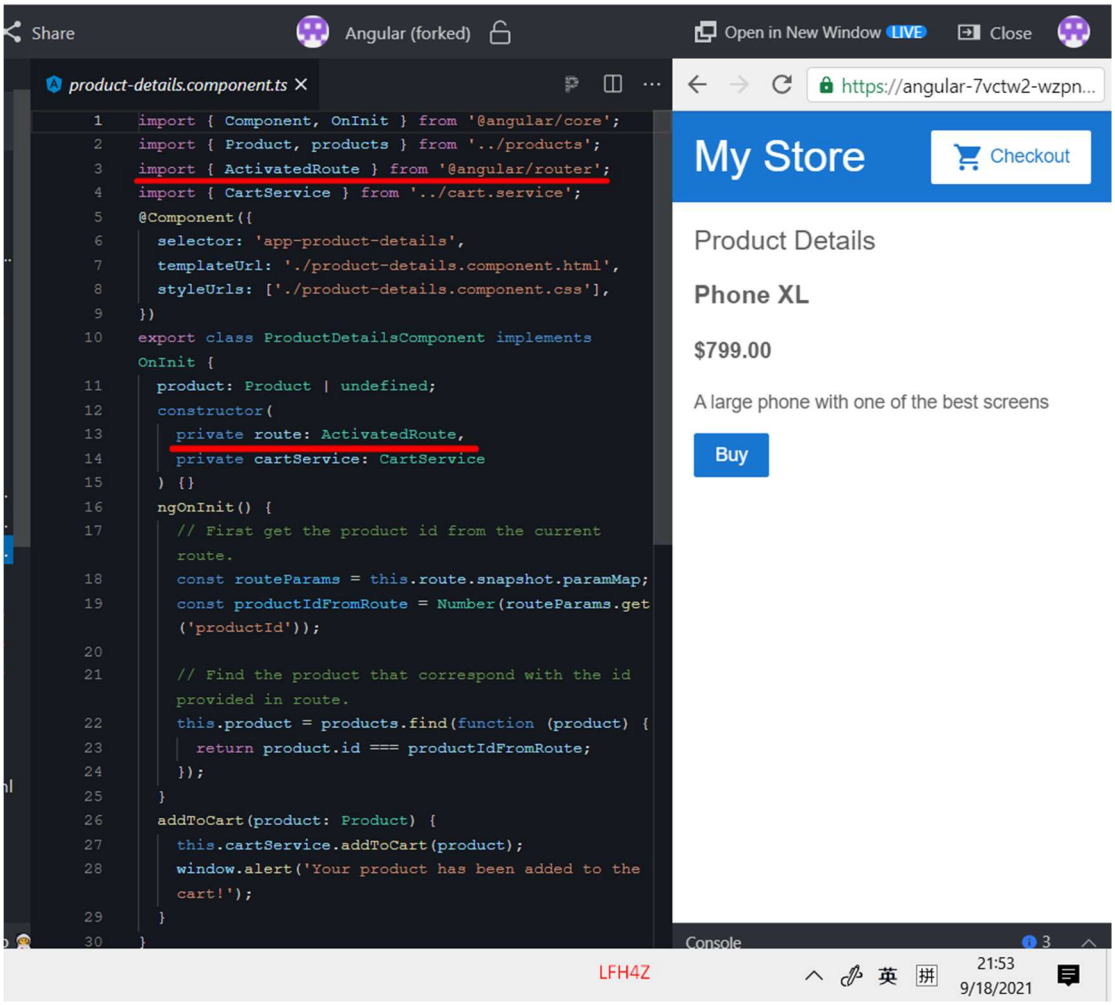
3) Add the link to the product-list page



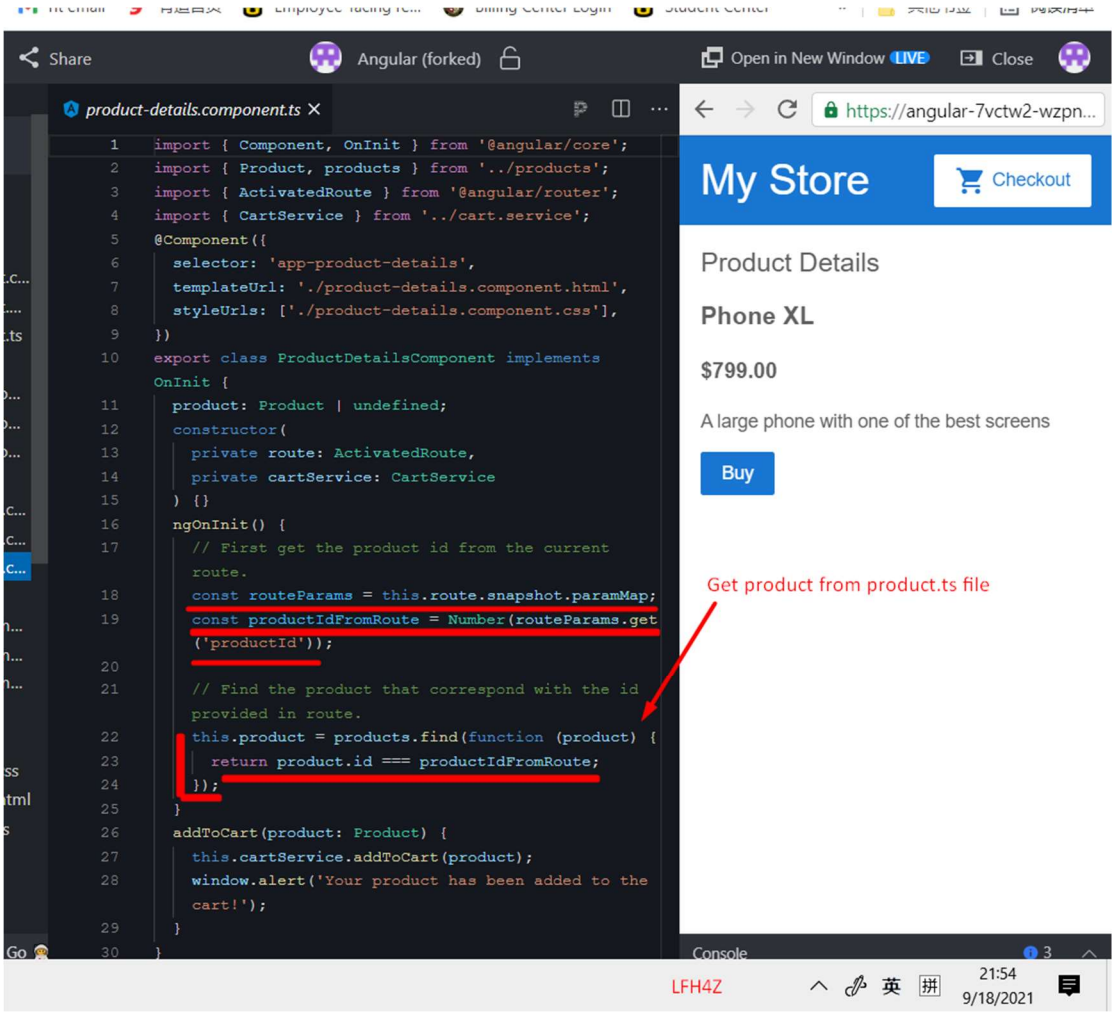
4)Import the Product class and product object to the product detail page



5) Import ActivatedRoute so that the link information will be gotten

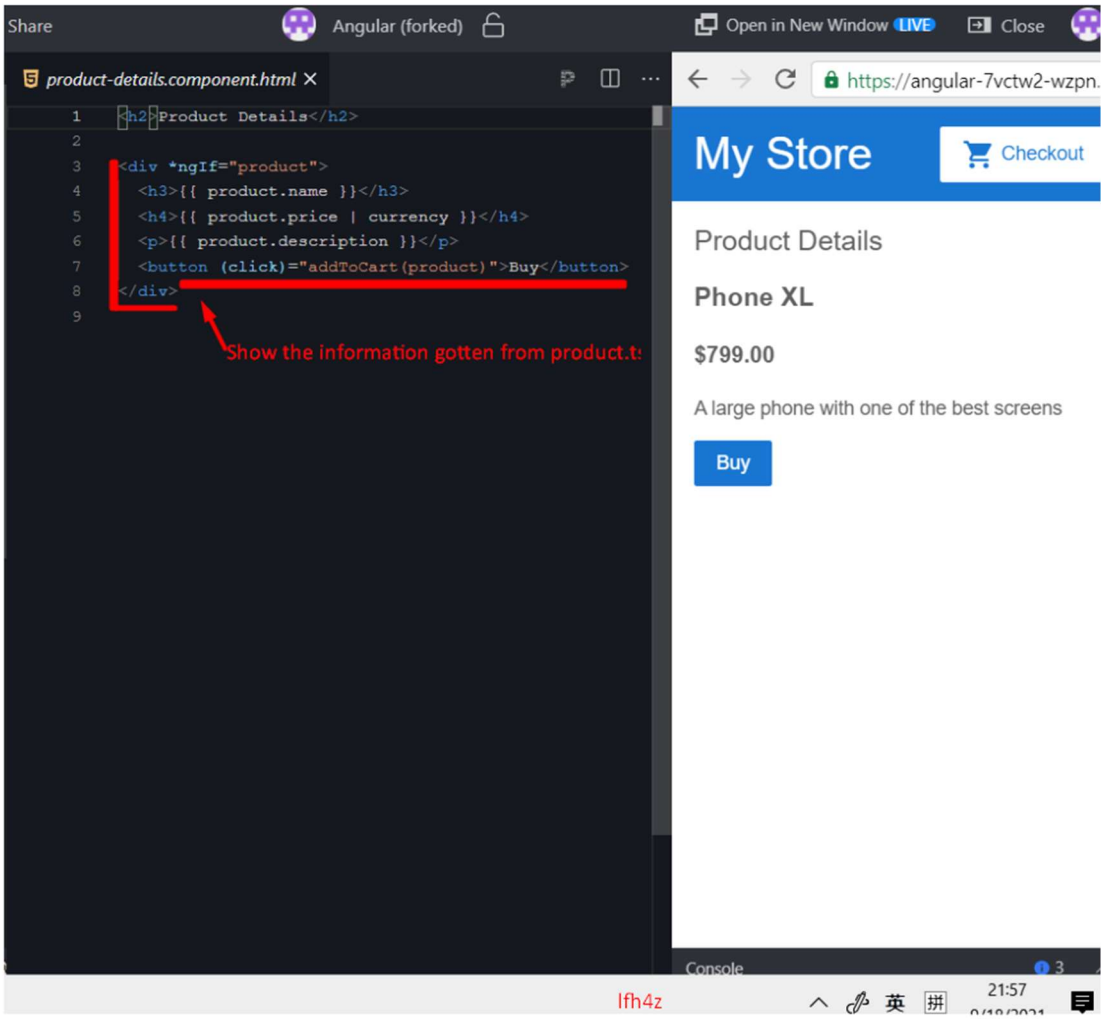


6)Use the this.route.snapshot.paramMap to get the link information and use routeParams.get{'productId'} to get the product id. Then use the gotten product id to get the product from the product.ts

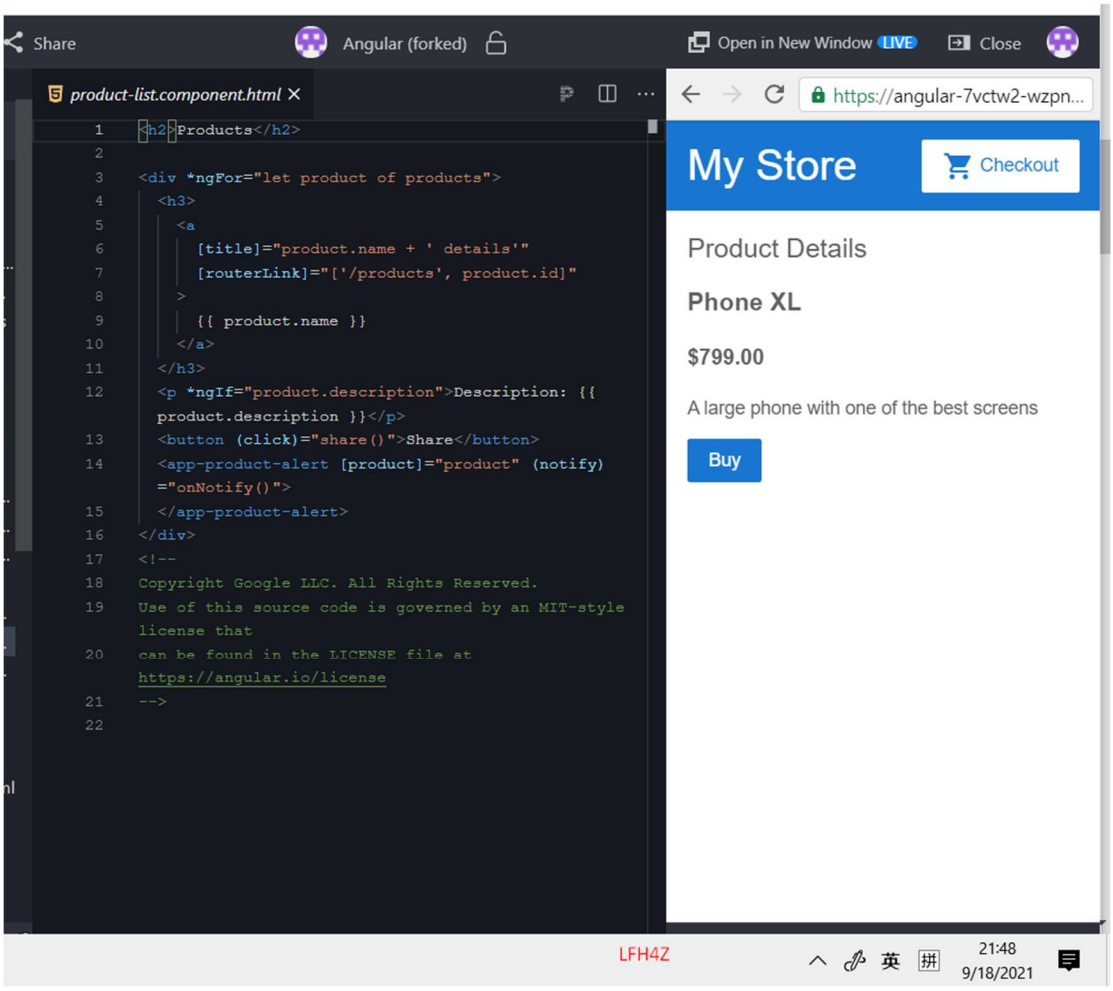


7) Set the html content in the product detail page and create a button "buy" which call on the

function of the service to add this item to cart.

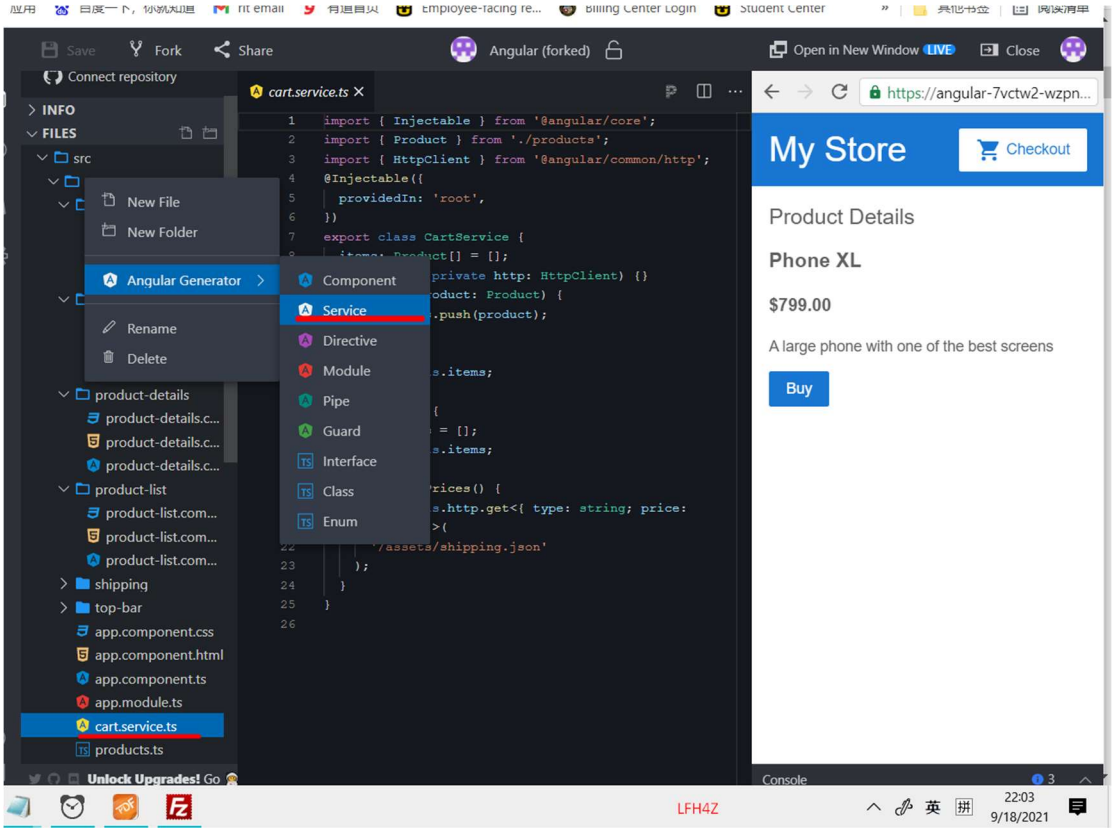


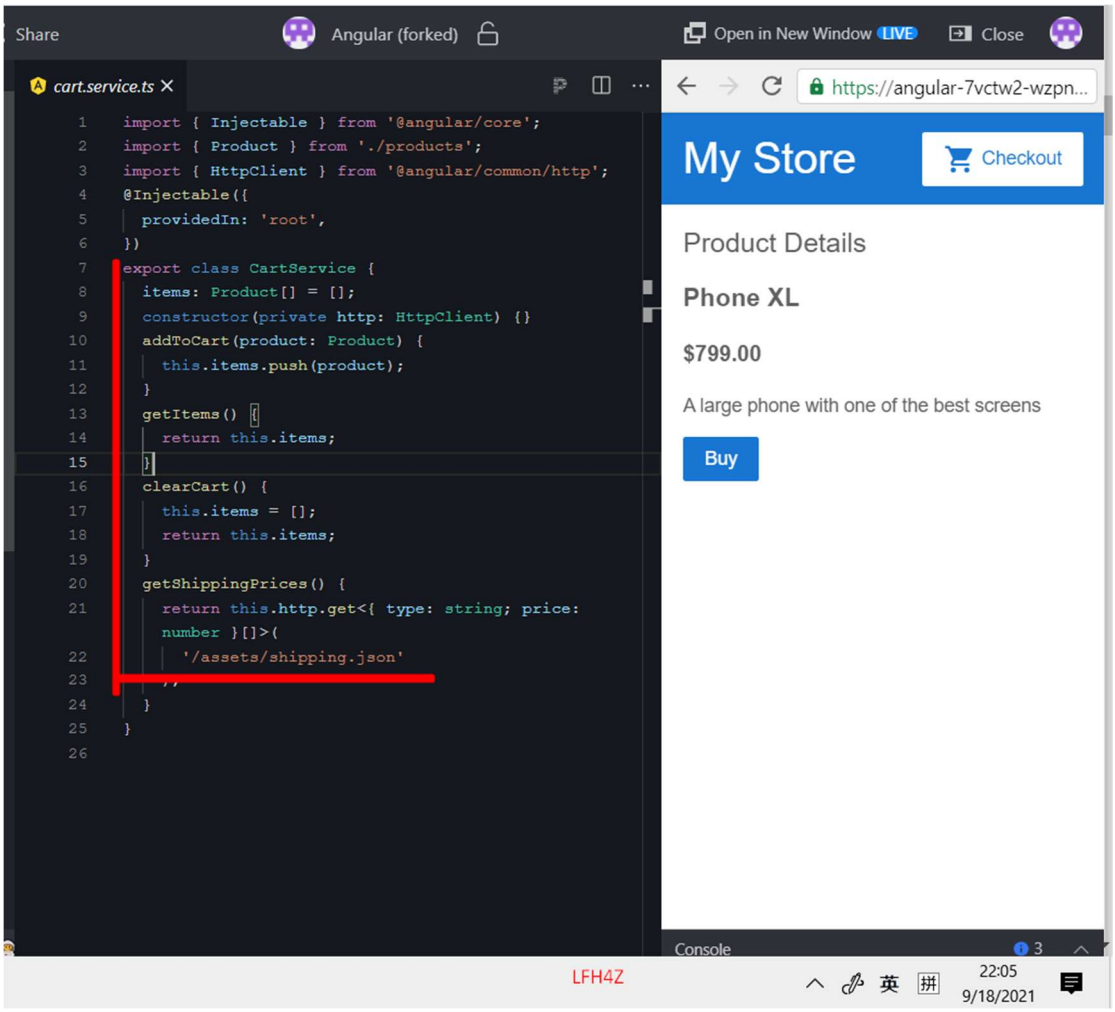
8)add the routerLink to the product.name so that when clicking the product.name, the link will be redirected to the detail page.

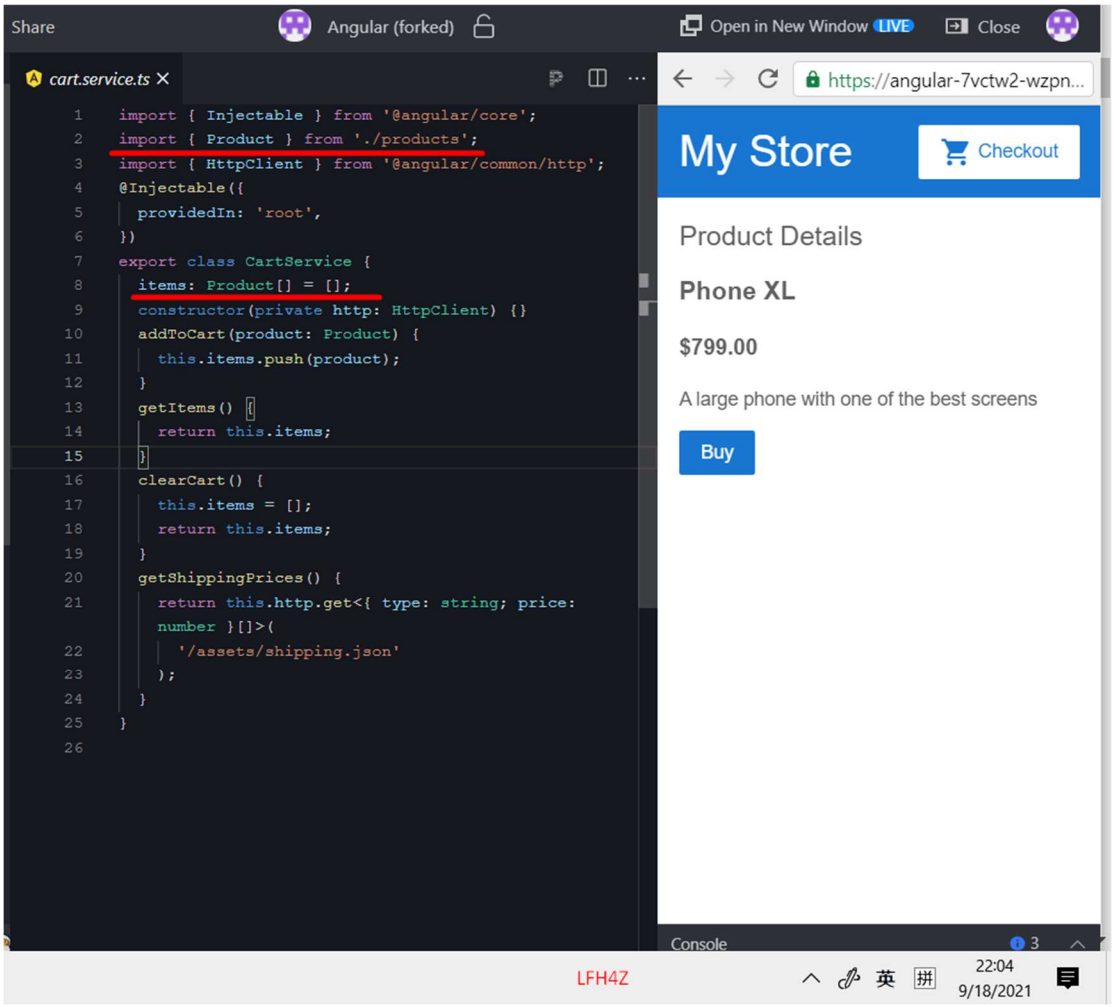


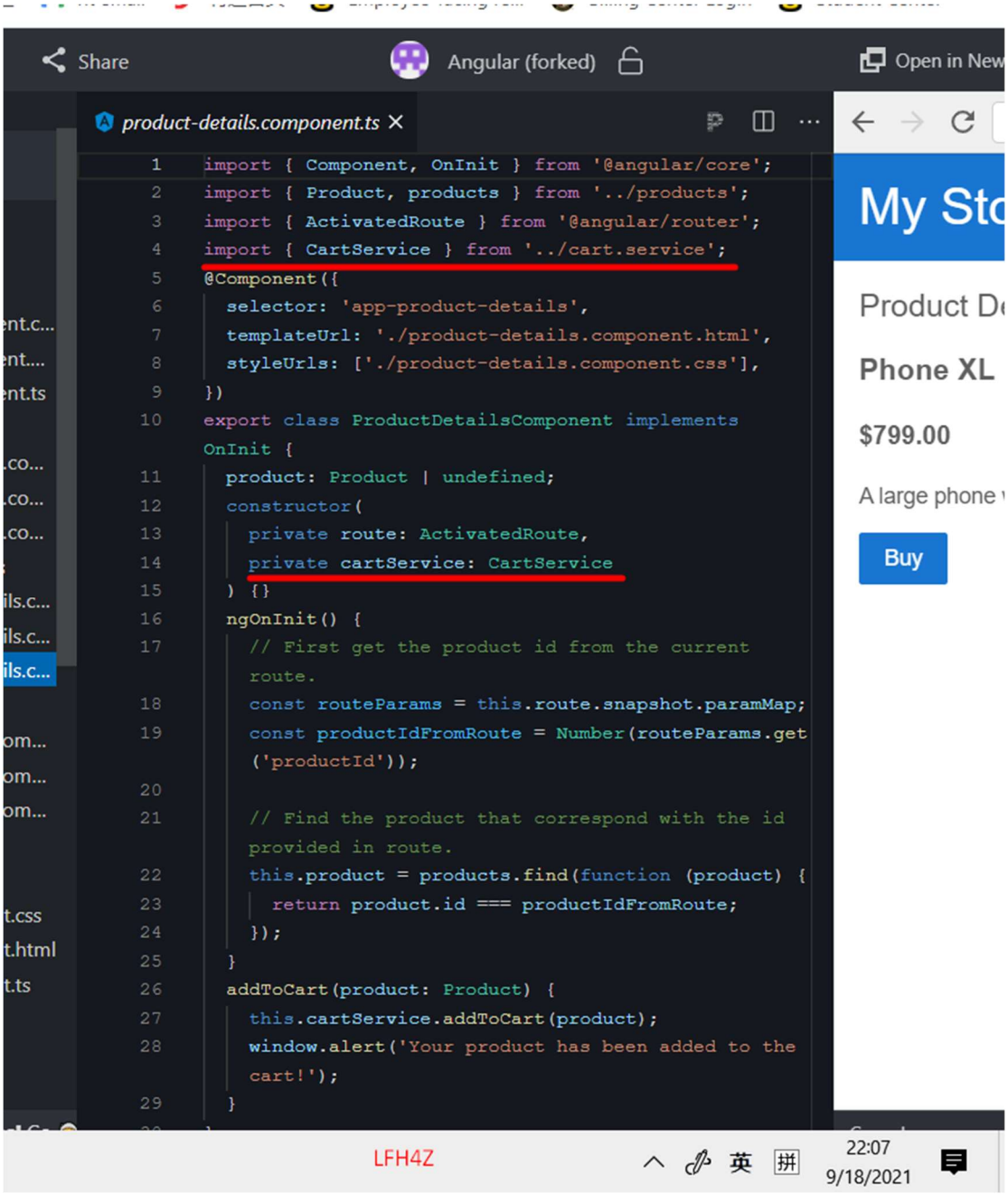
3. Using Service to Save Data

1) Create the cart service, import the object from product.ts, and create the function to be used within the export class. Then, import Cart Service to the product-detail component

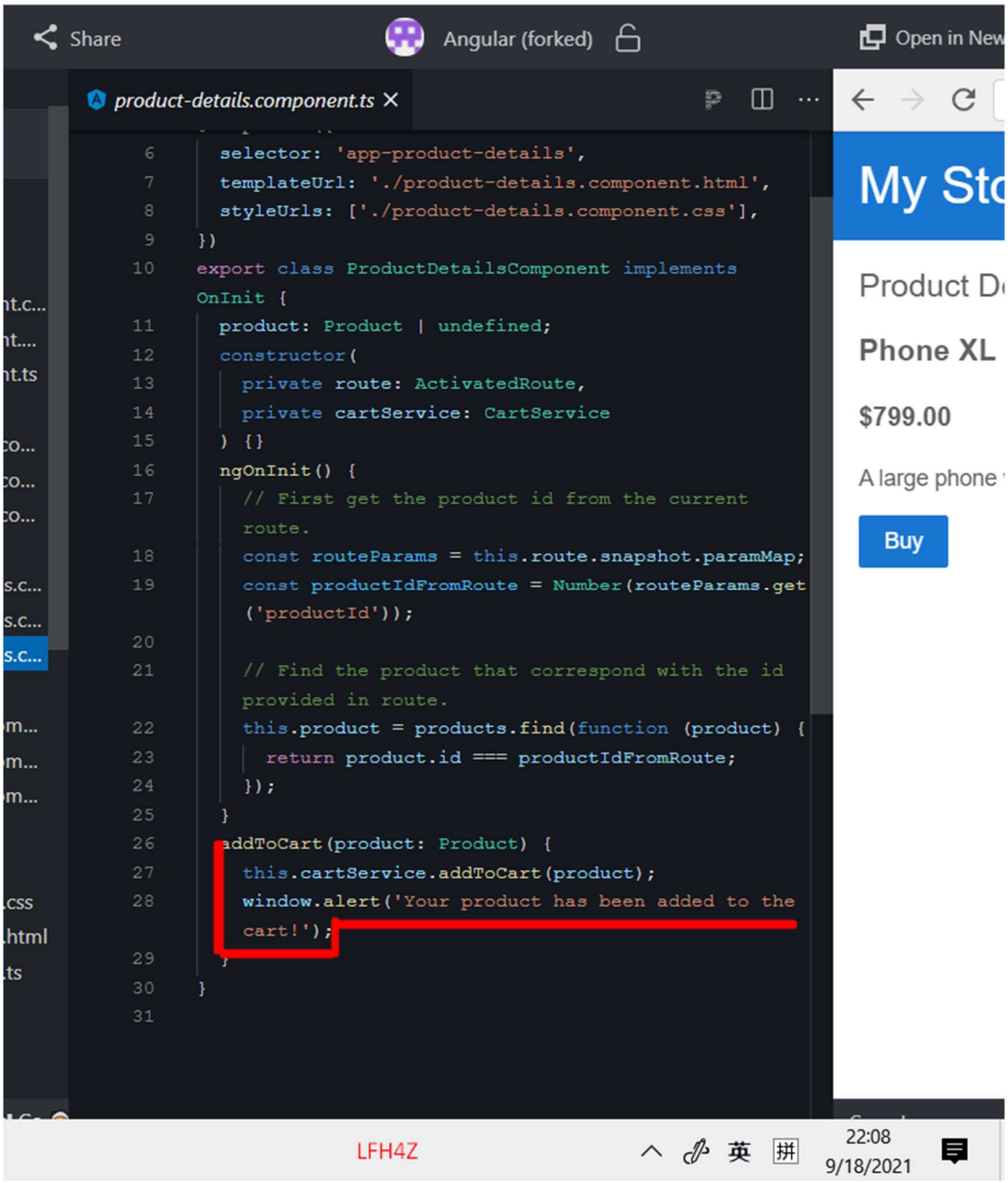




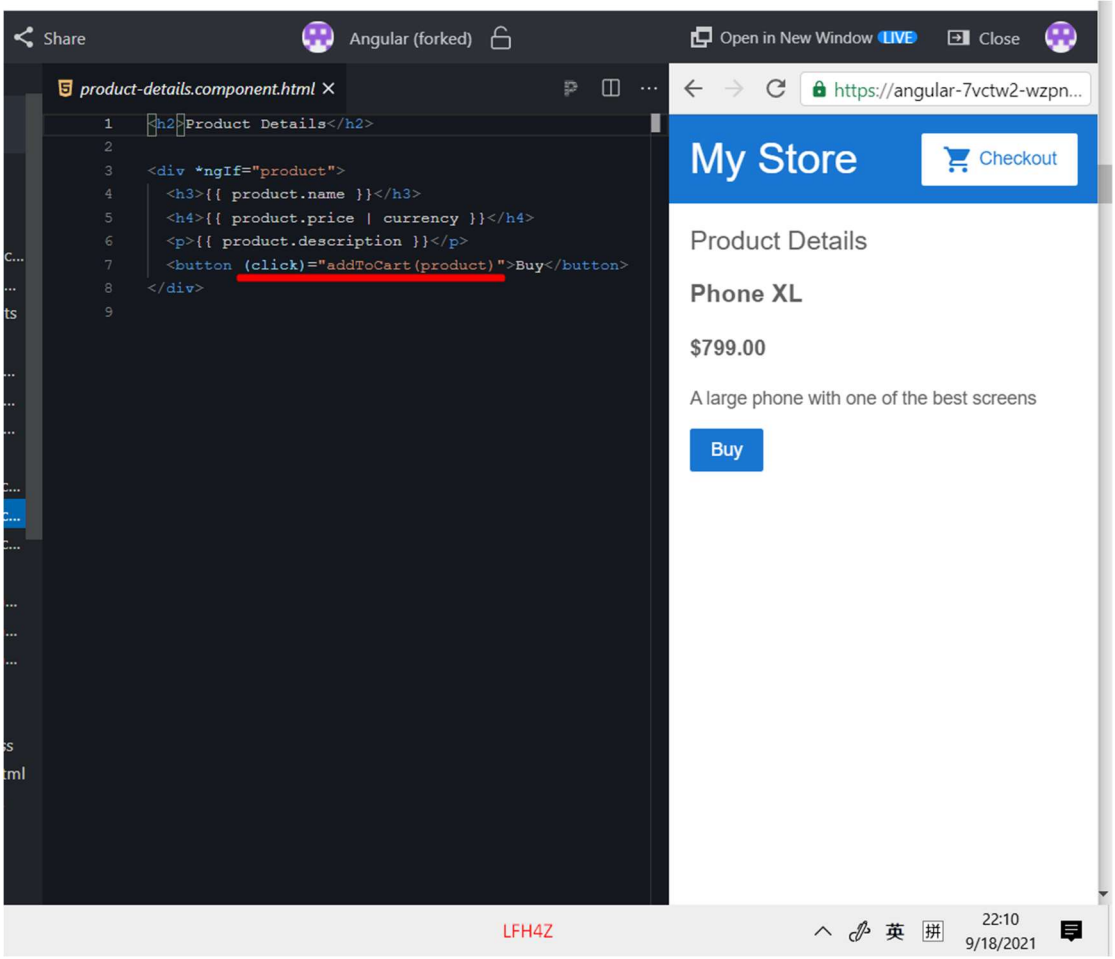

Share
Angular (forked)
Open in New Window
LIVE
Close
cart.service.ts
import { Injectable } from '@angular/core';
import { Product } from './products';
import { HttpClient } from '@angular/common/http';
@Injectable({
providedIn: 'root',
})
export class CartService {
items: Product[] = [];
constructor(private http: HttpClient) {}
addToCart(product: Product) {
this.items.push(product);
}
getItems() {
return this.items;
}
clearCart() {
this.items = [];
return this.items;
}
getShippingPrices() {
return this.http.get<{ type: string; price:
number }[]>(
'/assets/shipping.json'
);
}
}
https://angular-7vctw2-wzpn...
My Store
Checkout
Product Details
Phone XL
$799.00
A large phone with one of the best screens
Buy
Console
3
LFH4Z
英
拼
22:04
9/18/2021



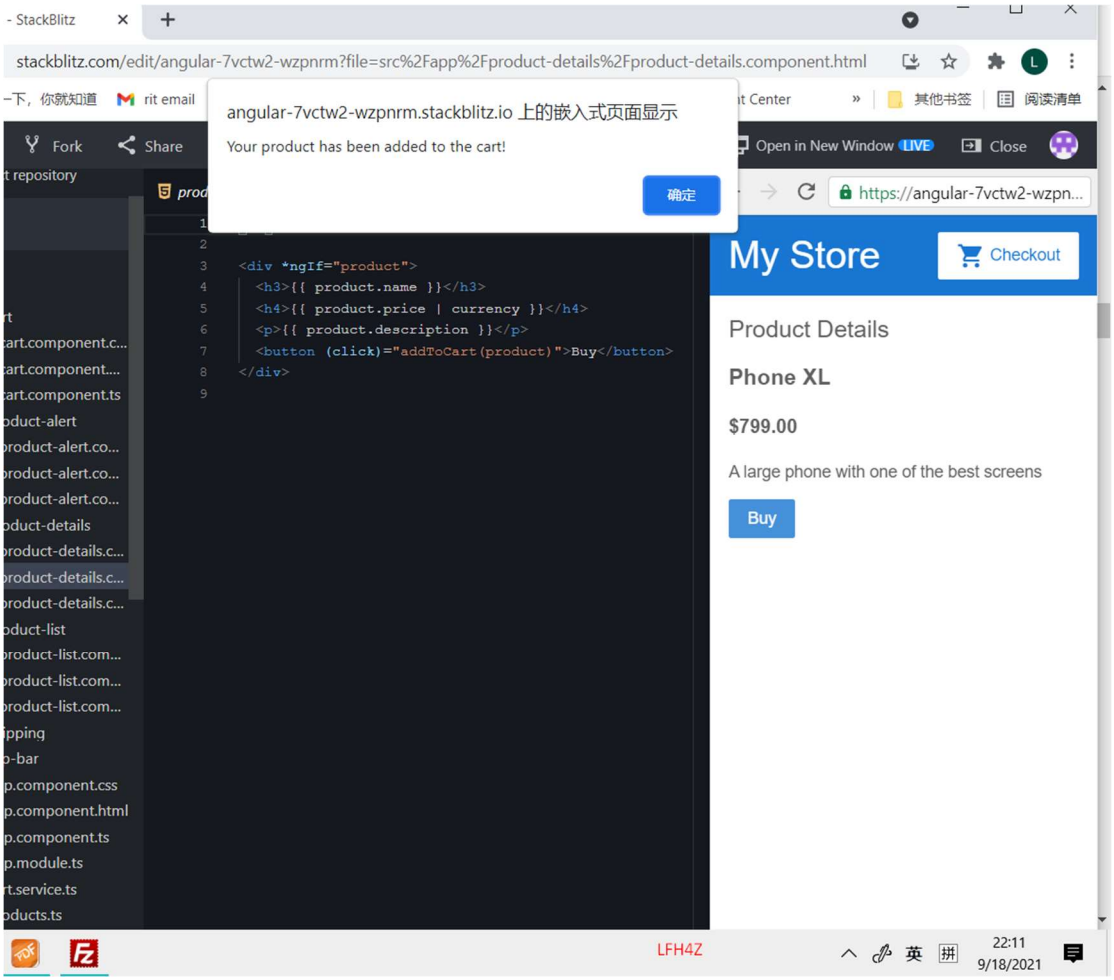
2)Add function in the product detail component to use the service



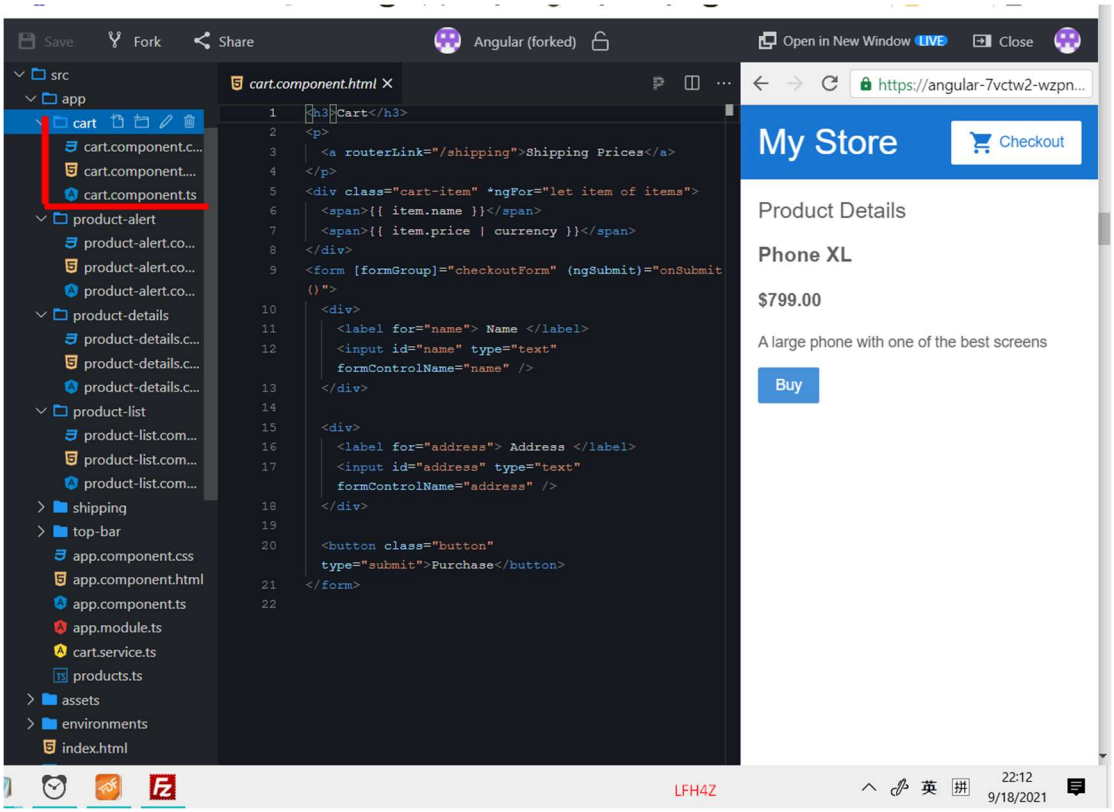
3)Create the button on the product detail page to use the function of the product detail component



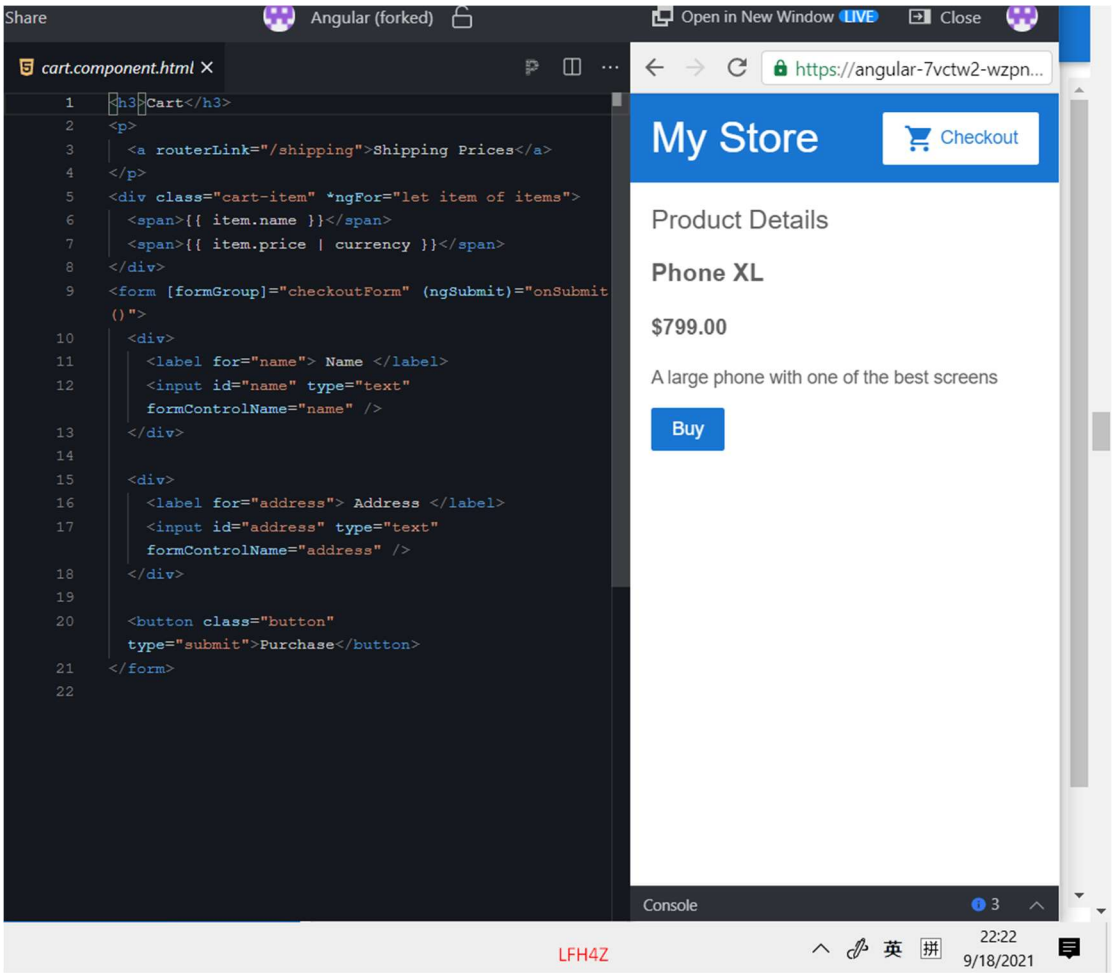
4)Verify the button



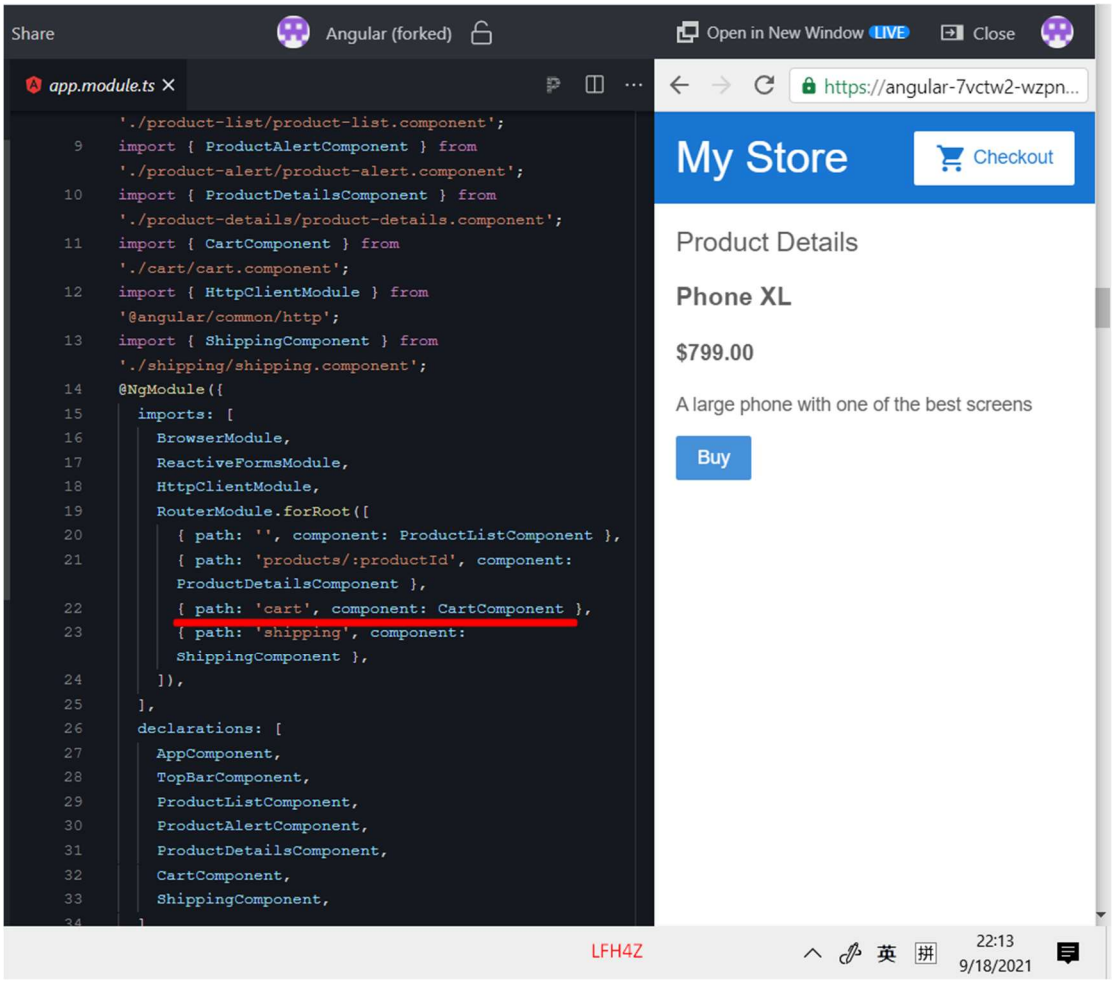
5)create the cart component to show the view of the cart



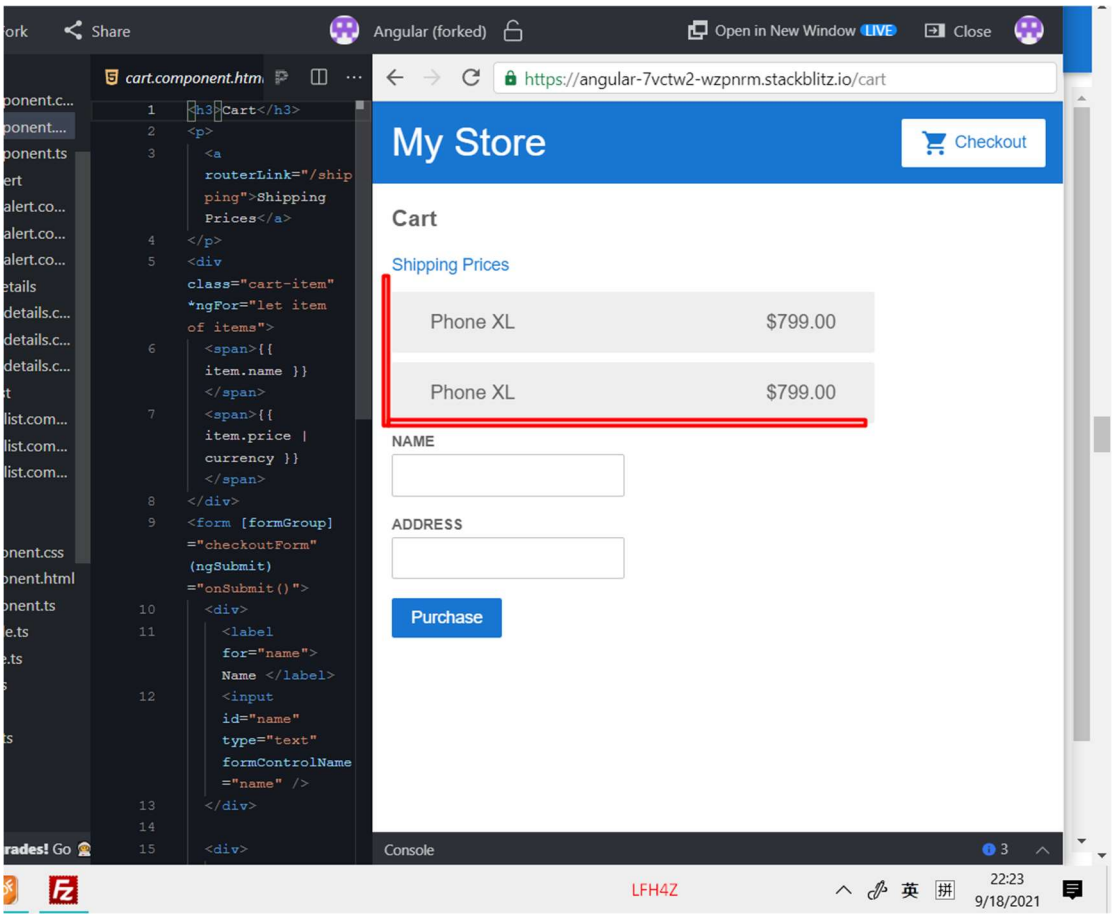
6) set the html content in the cart component



7) Set the URL for the cart page.

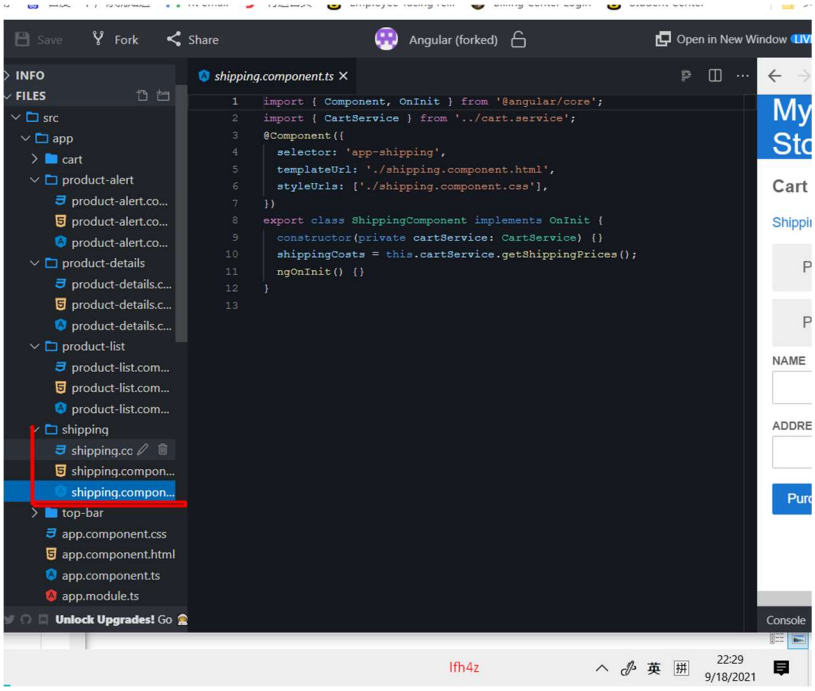


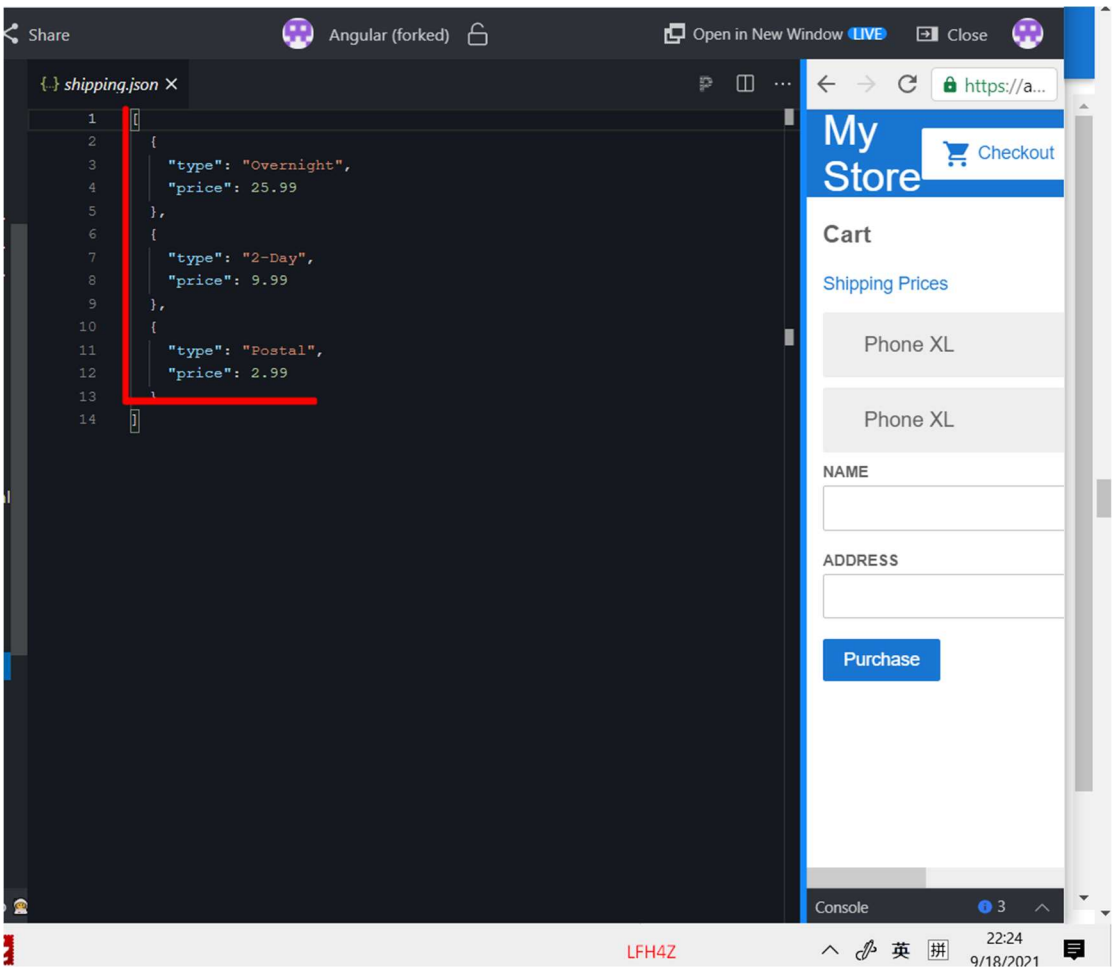
8)Verify the progress by clicking the checkout page which contains the items added



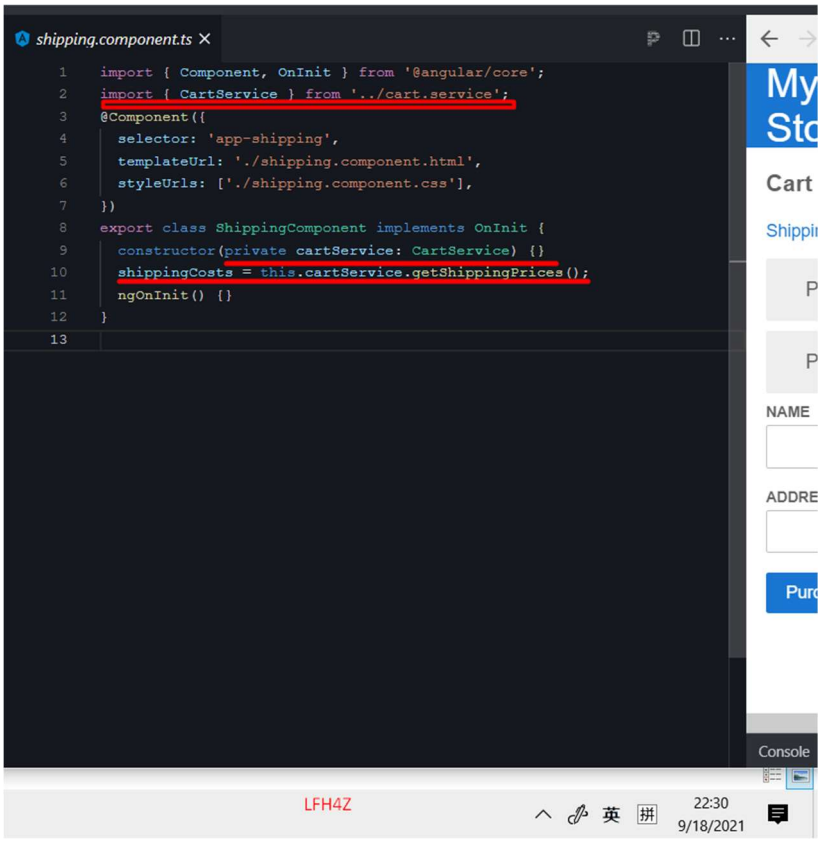
4. Using Http to Get Data

1)Create shipping component and create the shipping.json file

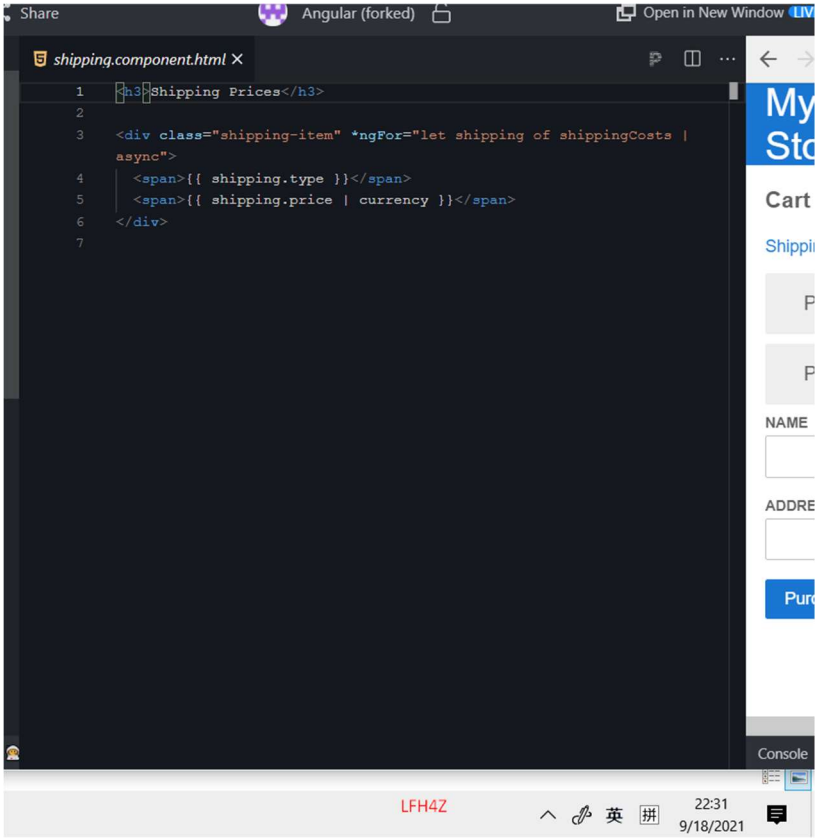




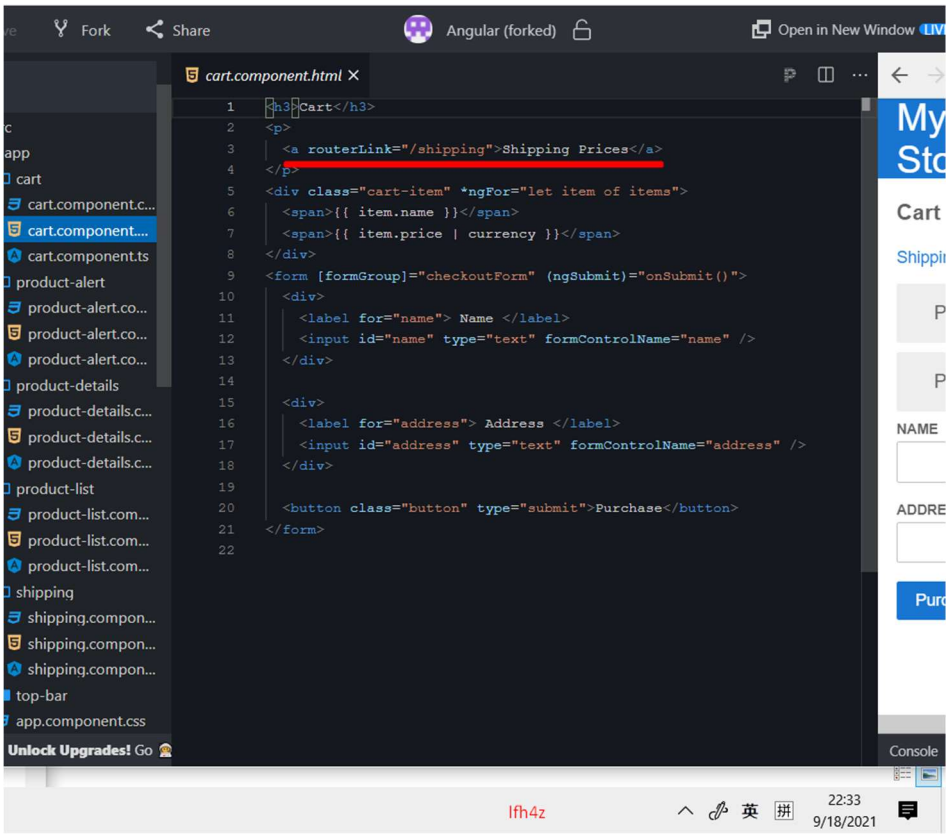
2) Import CartService and create an object in the constructor(). Then use this service object to get shipping price.



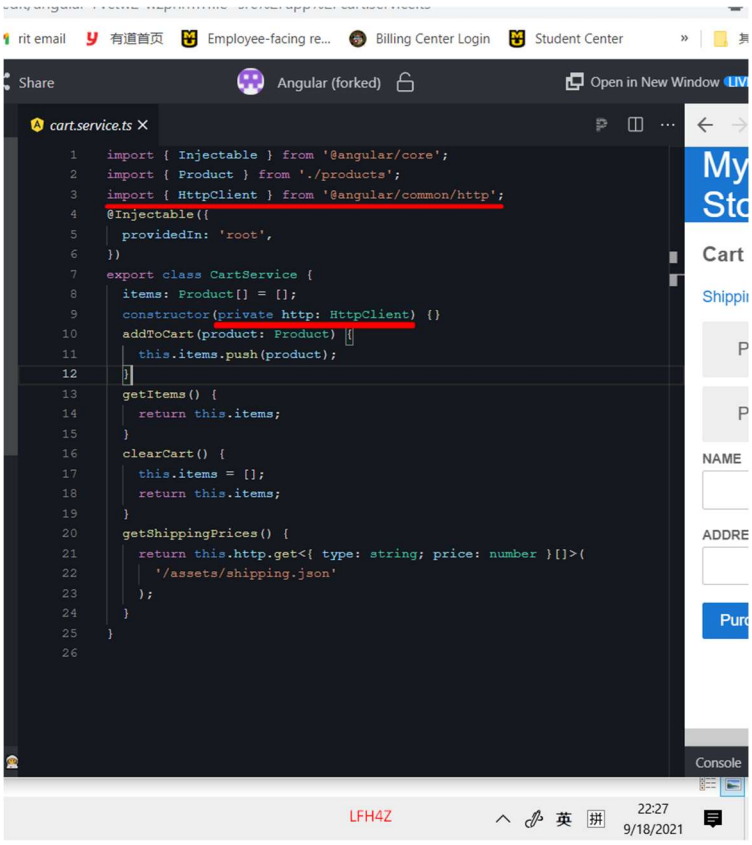
3) Set up the html file to show the content of shipping on the shipping page



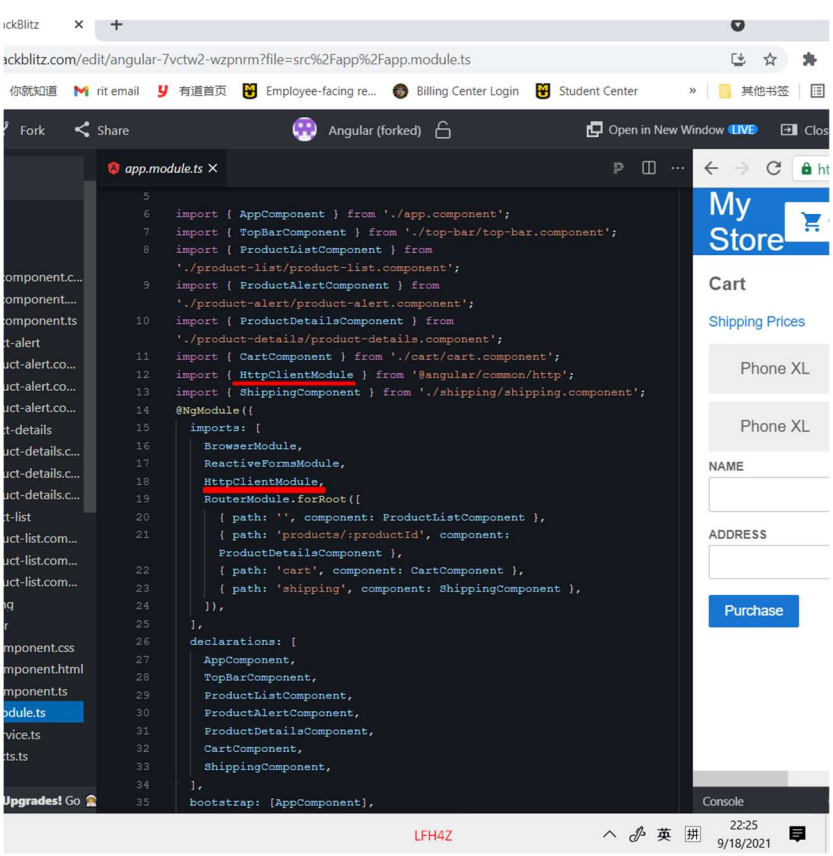
4)Set up a link on the cart page which links the shipping page



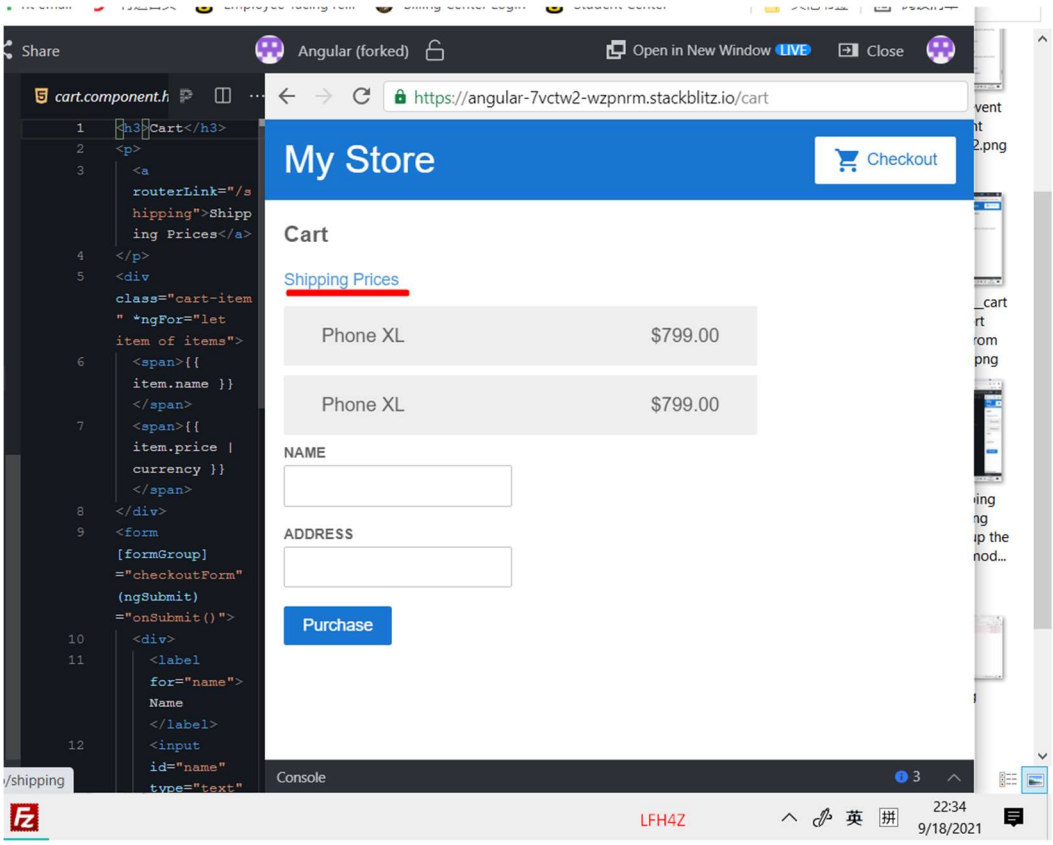
5) Import the HttpClient and create an object in the constructor()

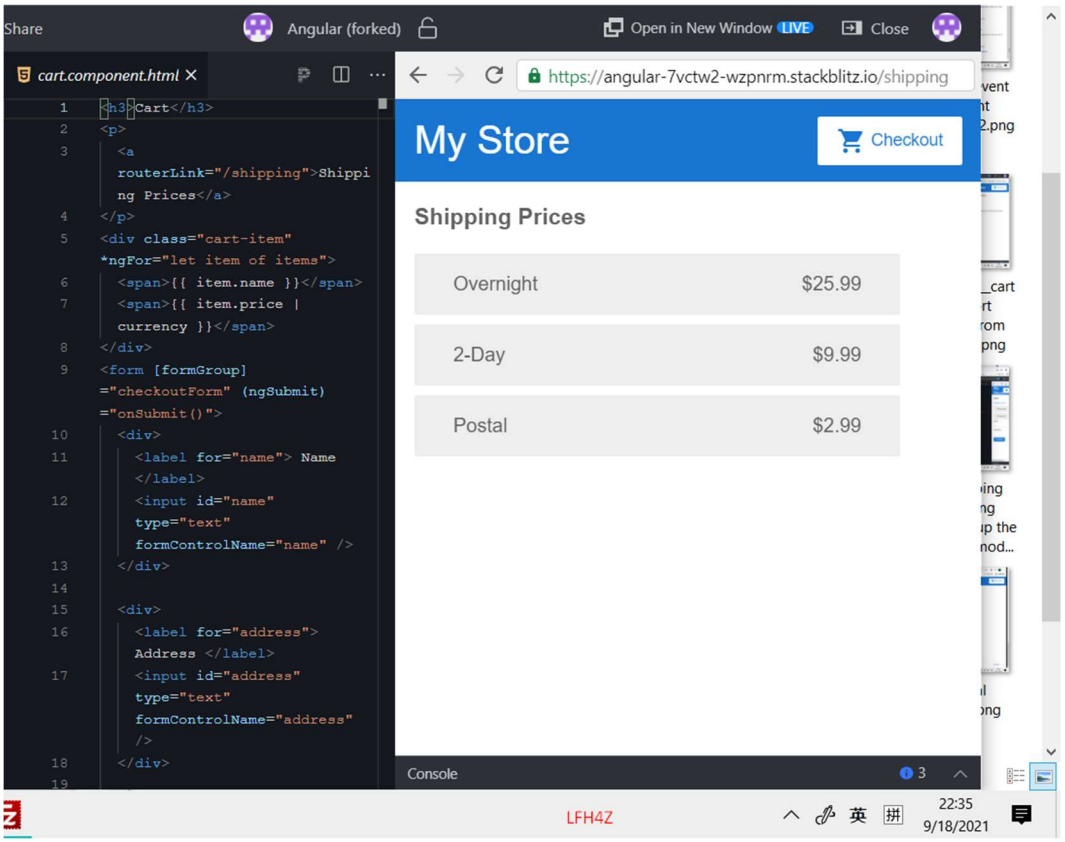


6) Add the HttpClientModule to the modul.ts file



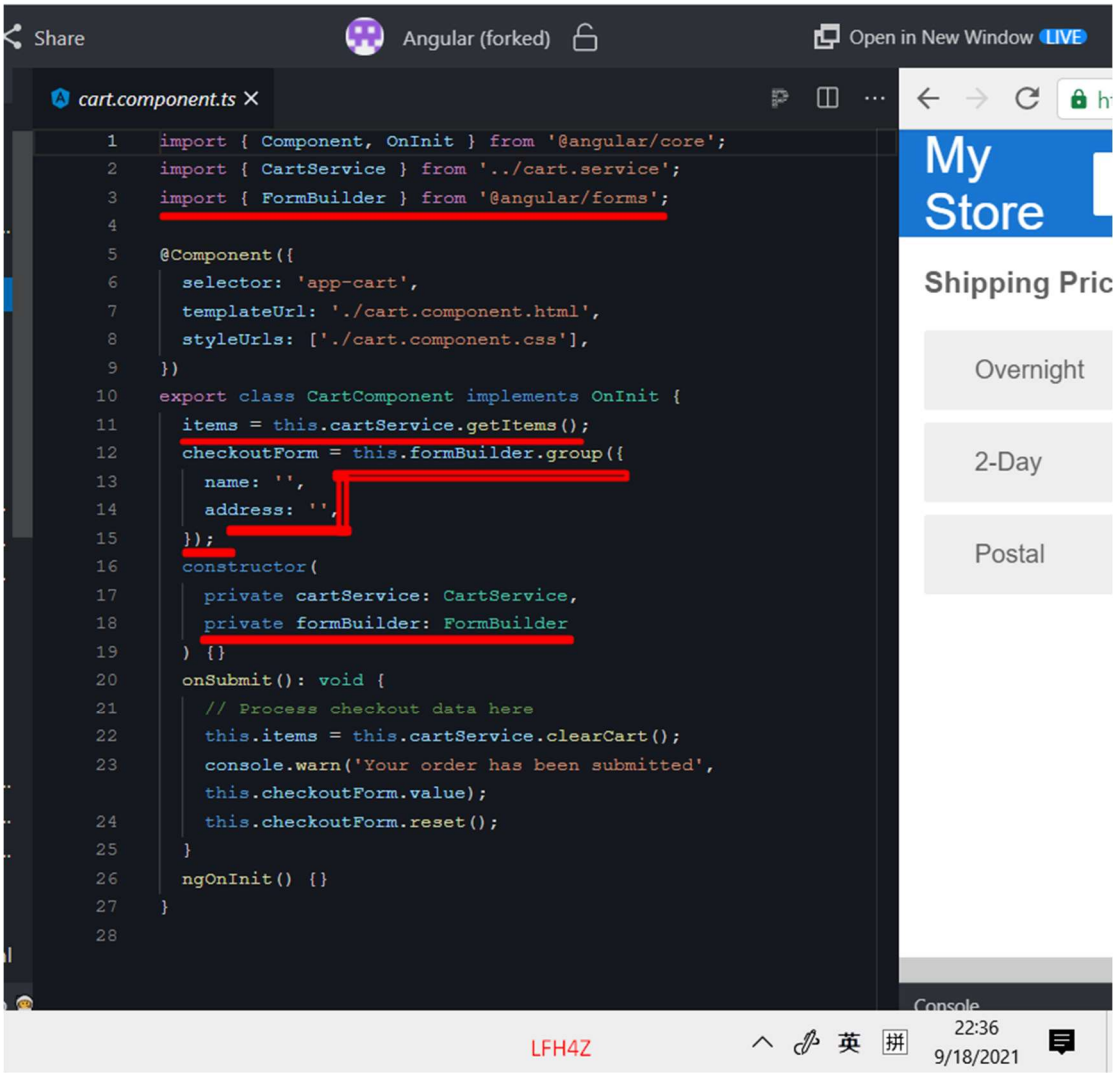
7)Verify the shipping page by clicking the link



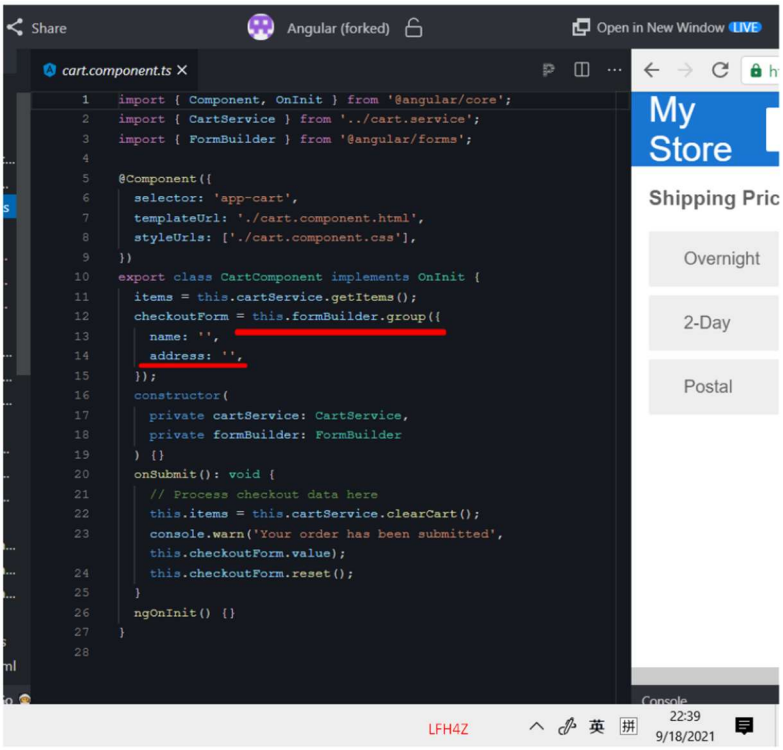


5. Using form

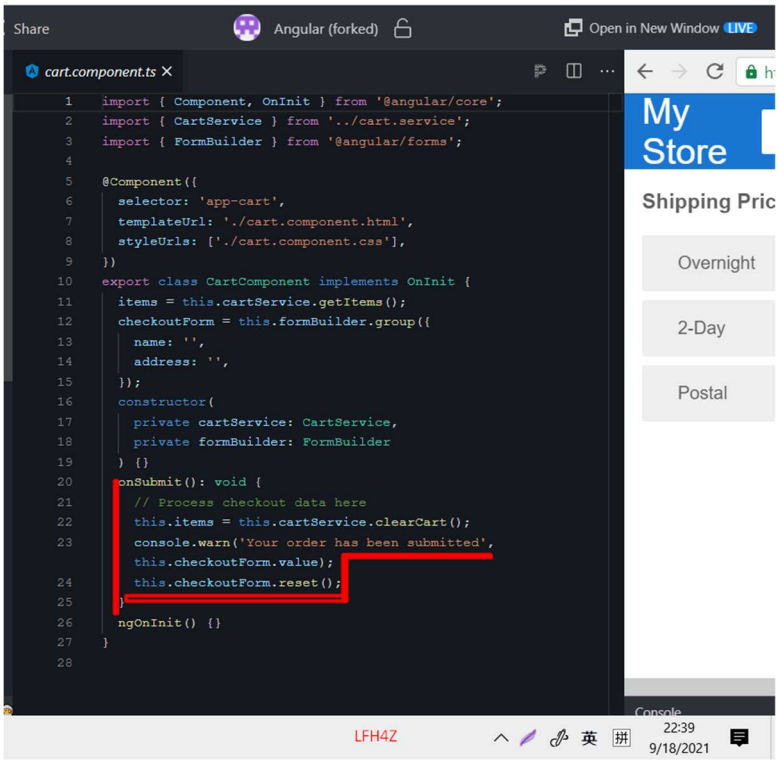
1) Import the “FomBuilder” component and create a object in the constructor()



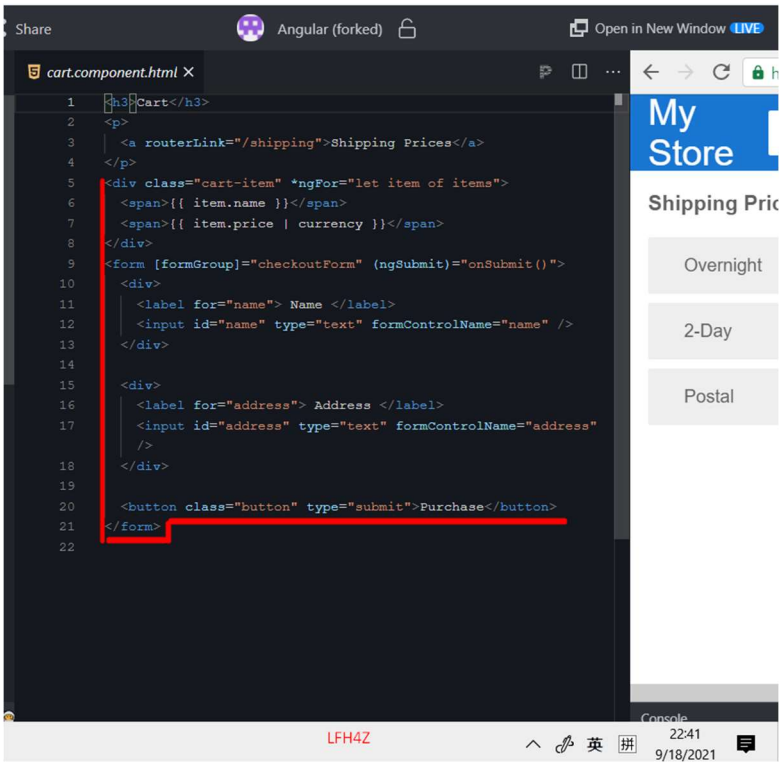
2)Create an object to set up the properties of the form.



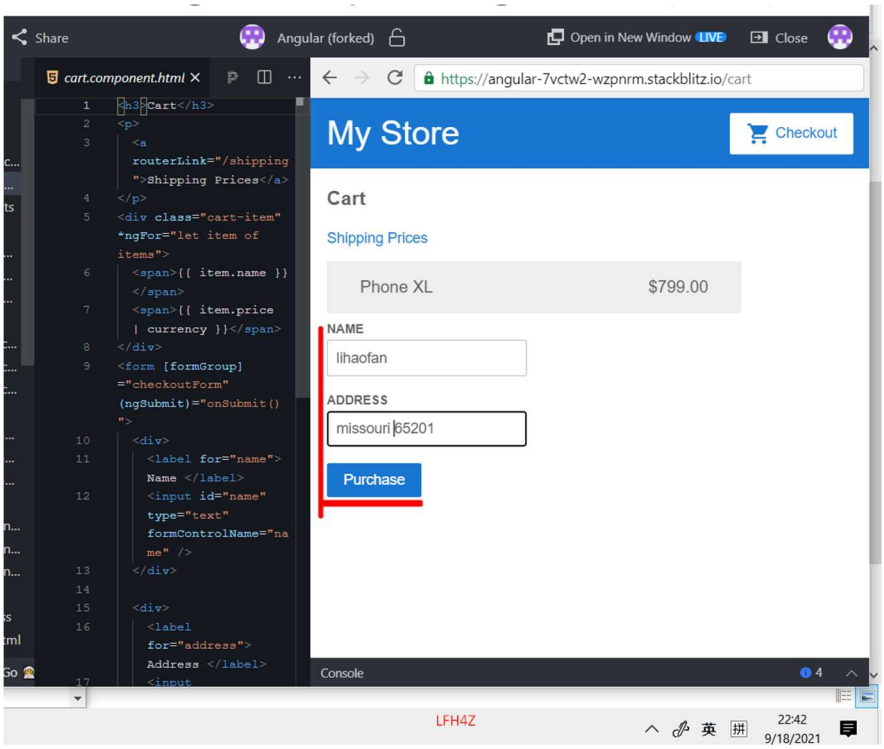
3)Create onSubmit() function to flush the content of the cart when the form is submitted.



4)Set up the html content in the cart component

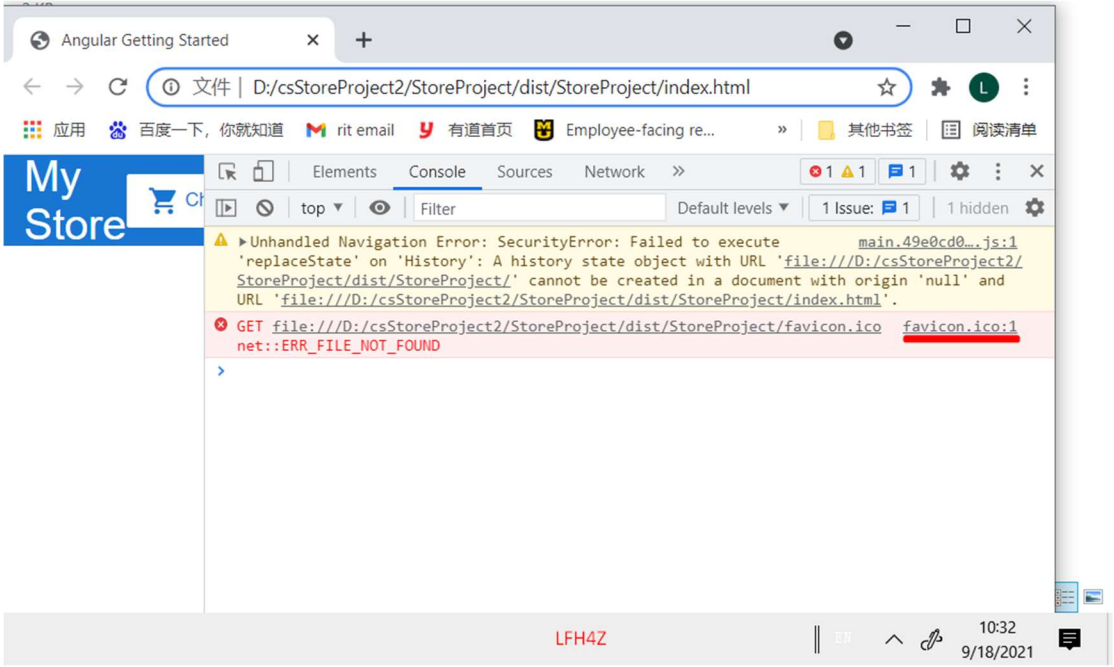


5)Verify the form



6.Deploy Error

1) I tried to use ng build to transform the project to static website. But I encountered the following favicon.ico error. I fixed it by adding an image, named “favicon.ico”.



2)When I tried to run the static website on my web server. It occurs the following error. This is because <base href="/"> can't guide the webpage to correct resources. I fixed it by deleting it and adding <base href="http://www.lihaofan.me/exploration1/StoreProject/">.