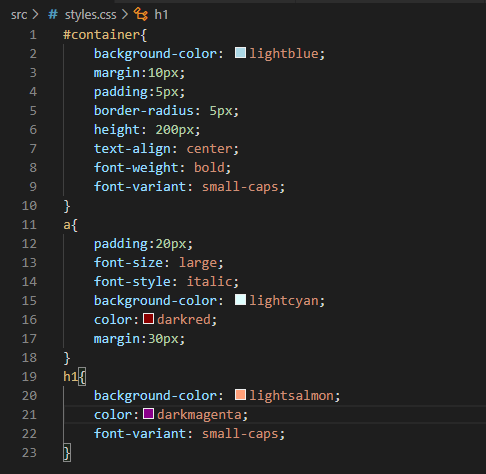
Angular Routing

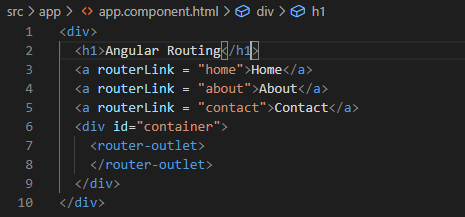
* The “Routing” concept is used to create page navigation in Angular Applications(angular 2+)
* Routing includes the process of mapping between the “route(url)” and corresponding component
  + http:/localhost:4200/home → Home Component
  + http:/localhost:4200/about → About Component
* The “@angular/router” package provides the essential API to create the routing
* <a routerLink=”/path”></a>
* <router-outlet></router-outlet>

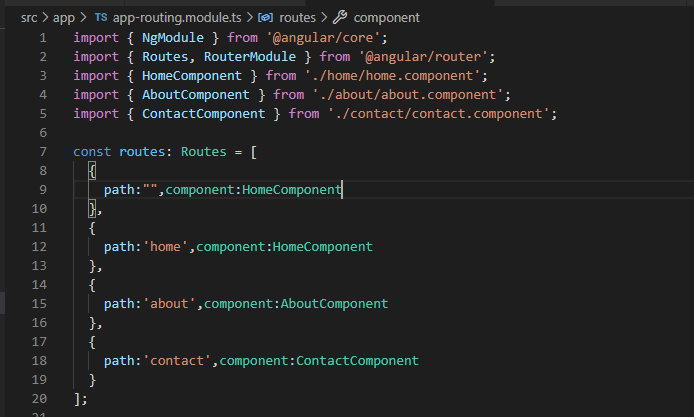
## Basic Routing Example:

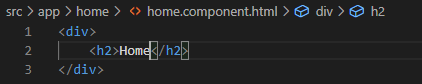
Step 1: Create a new Angular Application

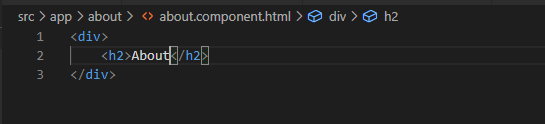
Step 2: Open the Project in Editor and create couple of components(About,Contact)

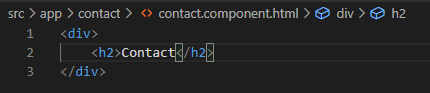






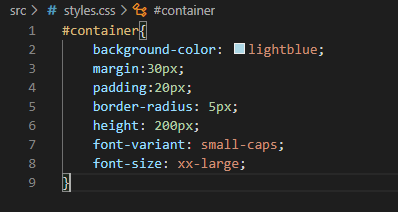


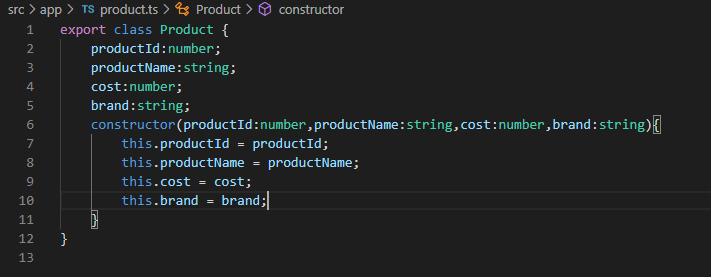




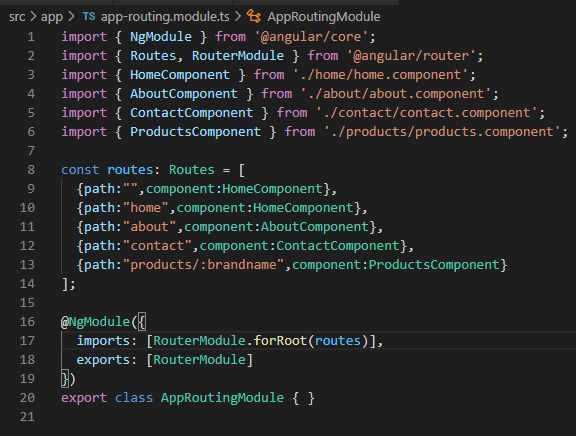
## Route Parameters:

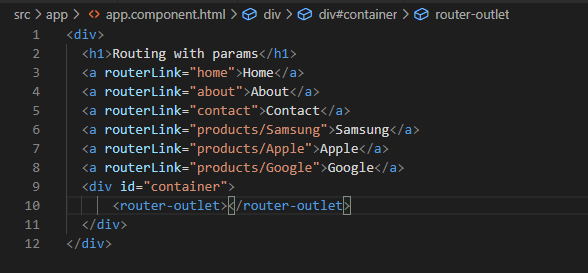
* You can pass parameters to the route
* Route Parameters is represented as “:parametername” syntax
* You can get the value of parameter in the component using “ActivedRoute” service.

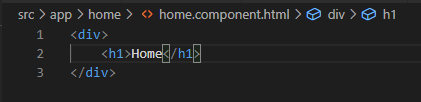


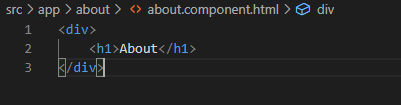


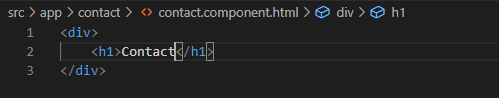


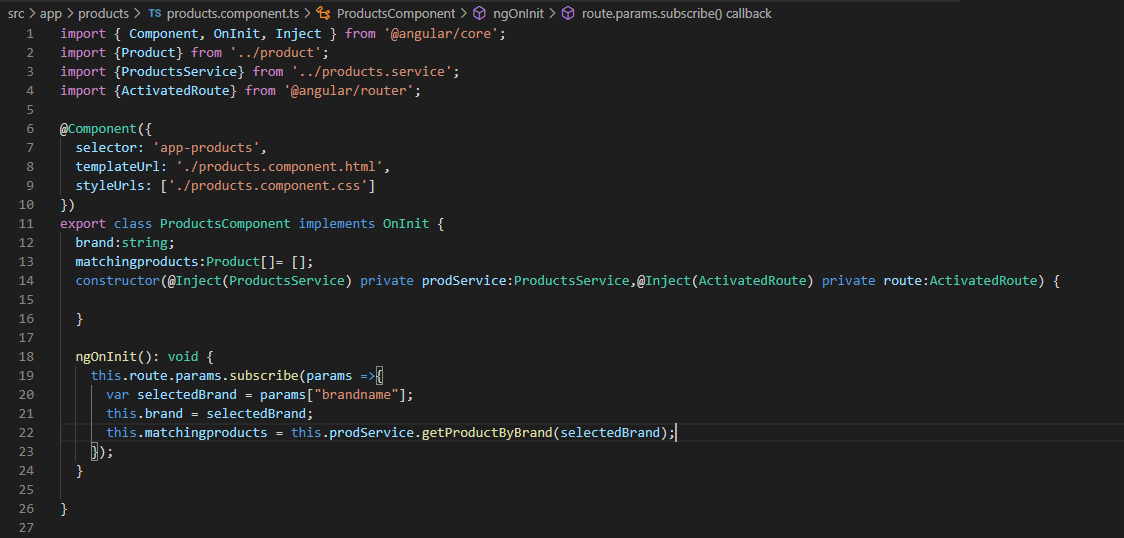


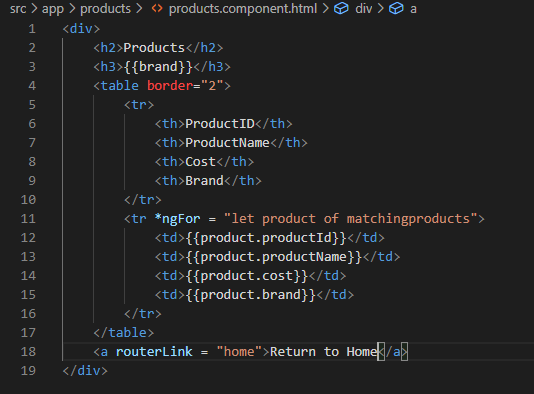






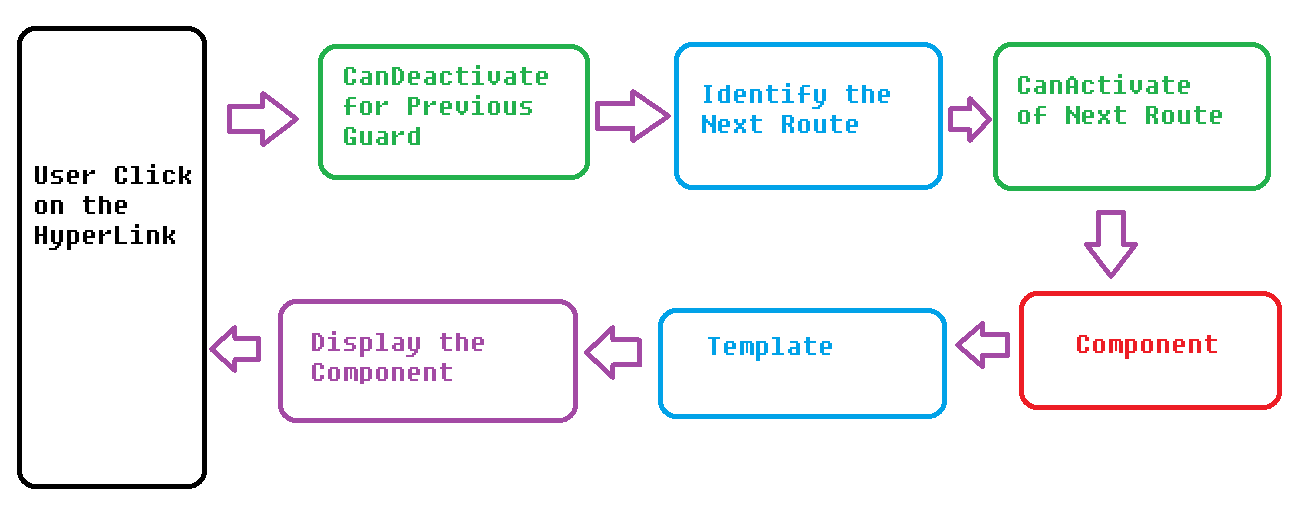






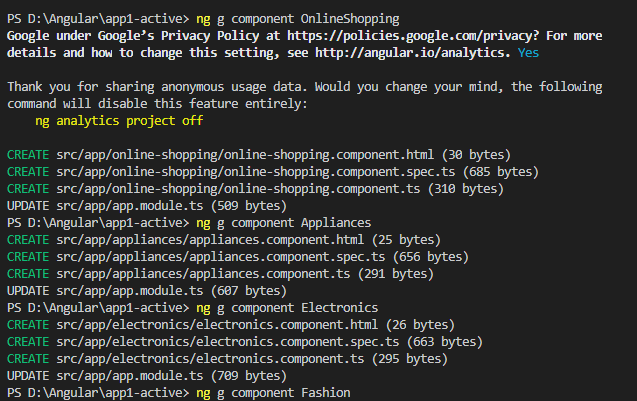
## Guards:

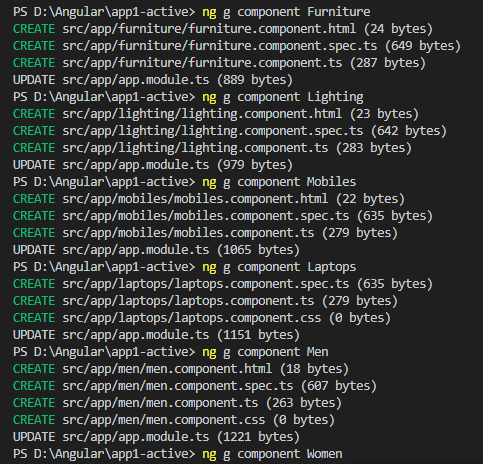
* The Guard service that executes at the specified situation while angular is navigating from one route to another route.
* Angular mainly supports two types of guards:
  + CanActivate : Executes before entering into a route
  + CanDeactivate: Executes before leaving a route

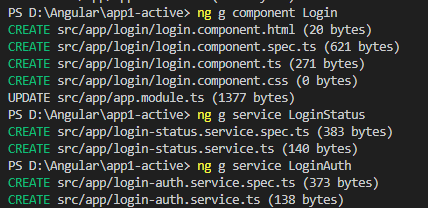


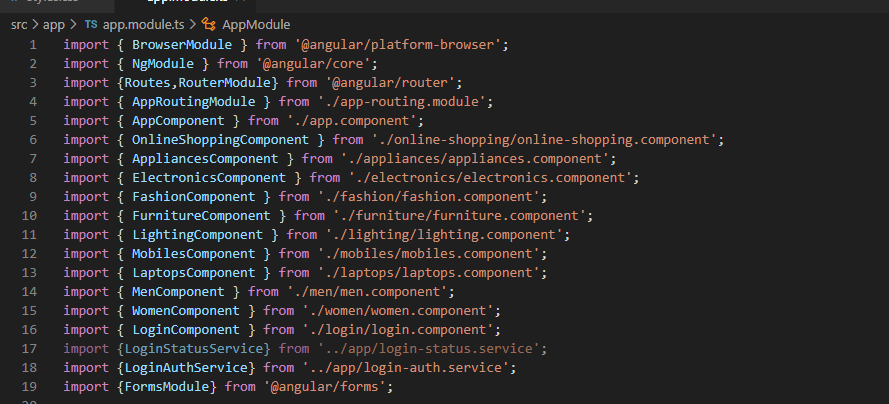
## CanActivate:

* The “CanActivate” guard executes before entering into a route
* Process→ User Clicks on the hyperlink → Identify the route → CanActivate Guard → Navigate to the Route → Corresponding Component
* The Guard can be created by implementing “CanActivate” interface
* The “CanActivate” interface has a method called “canActivate”. This method must return a boolean value
* It can receive an argument of “ ActivatedRouteSnapshot” type, which represents the current state of the route

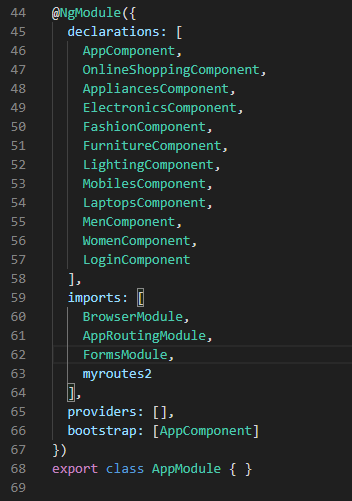


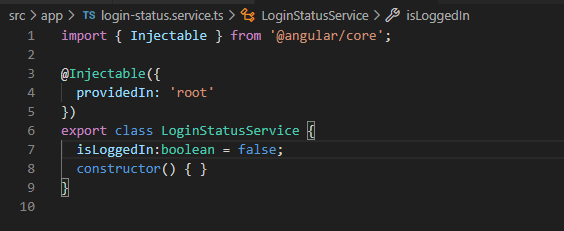


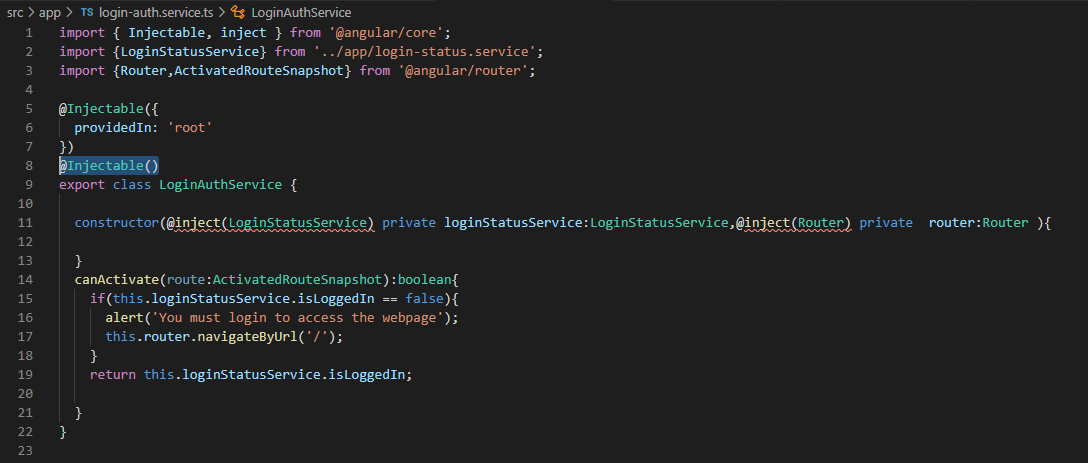






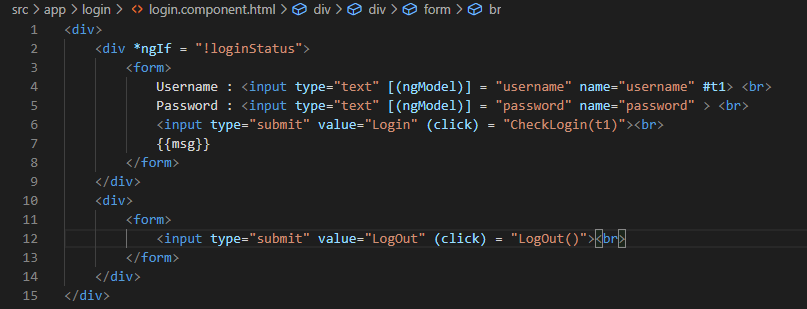


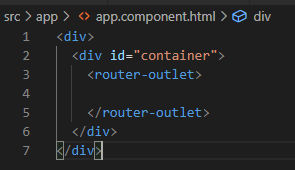


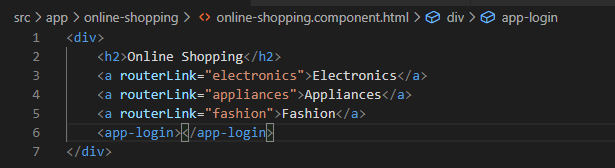


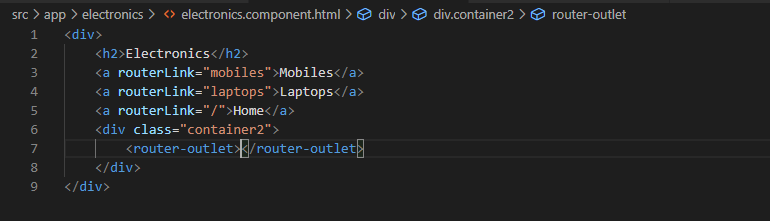


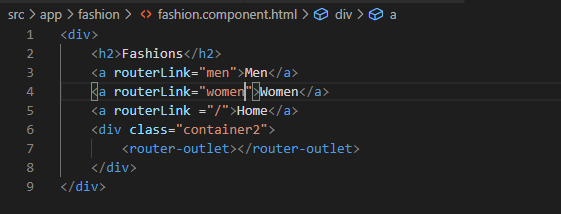


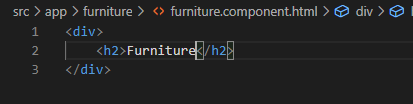












Do the same thing for the other child components