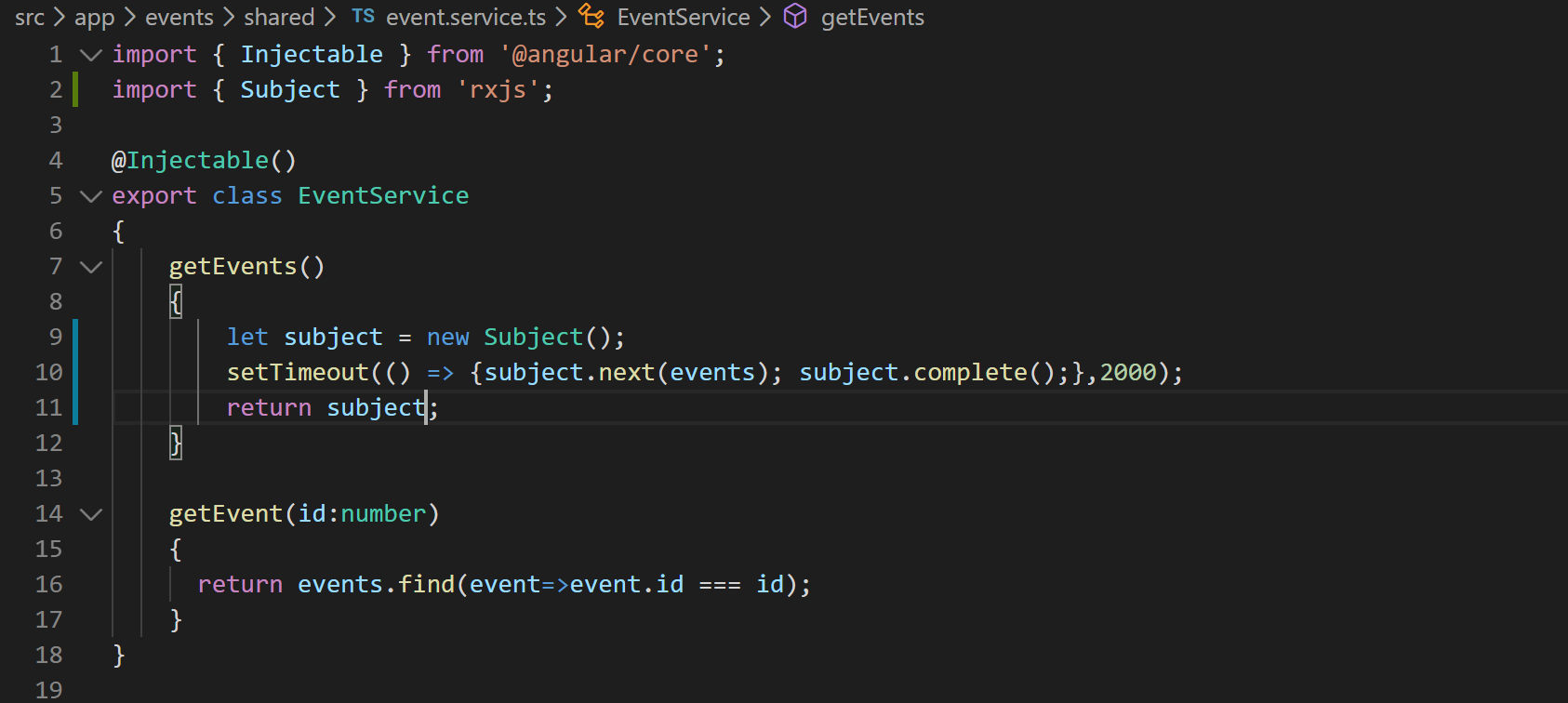
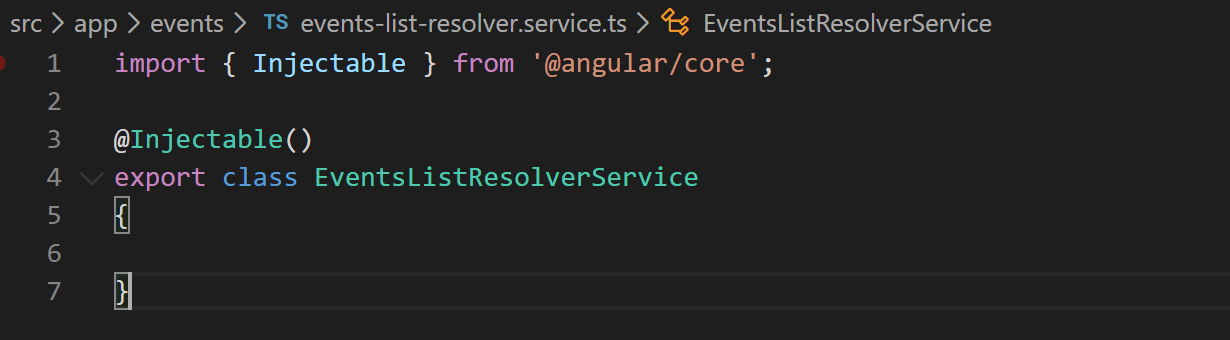
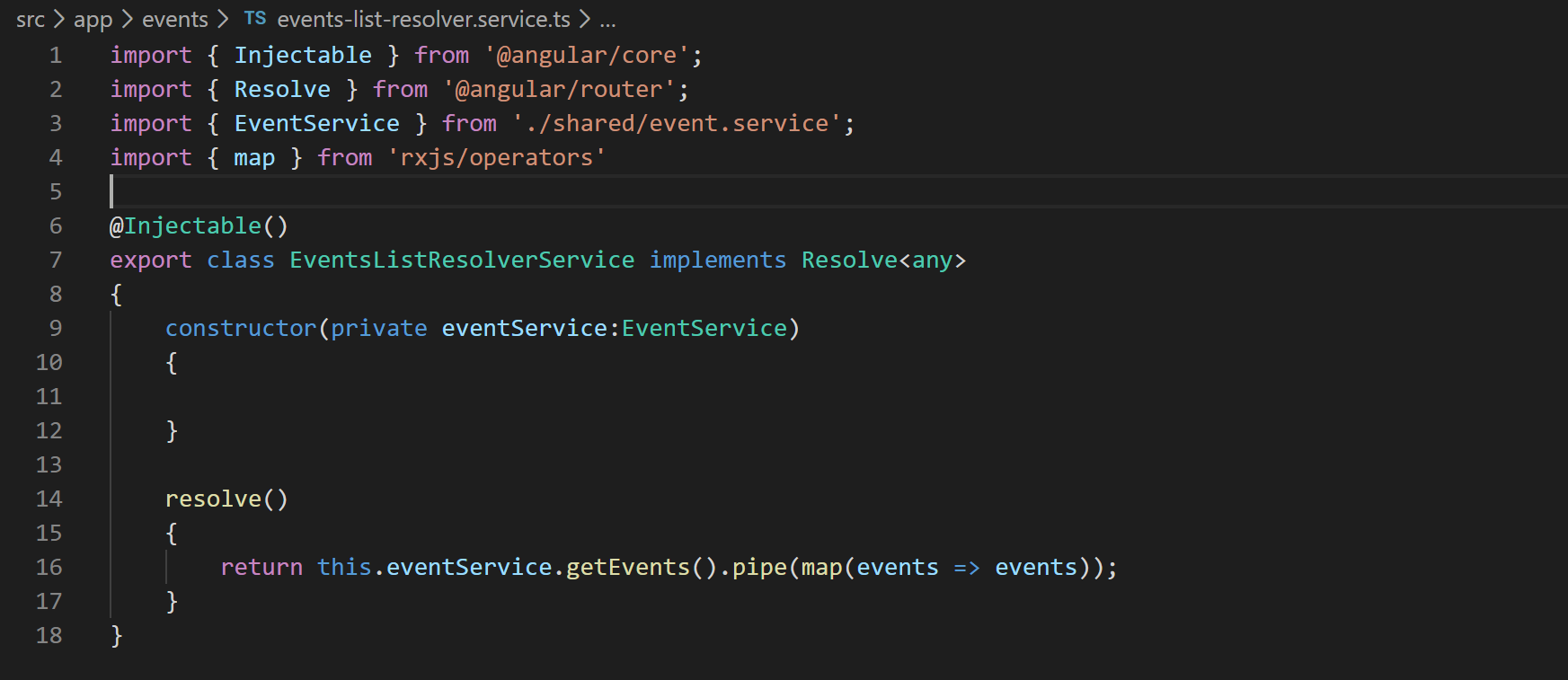
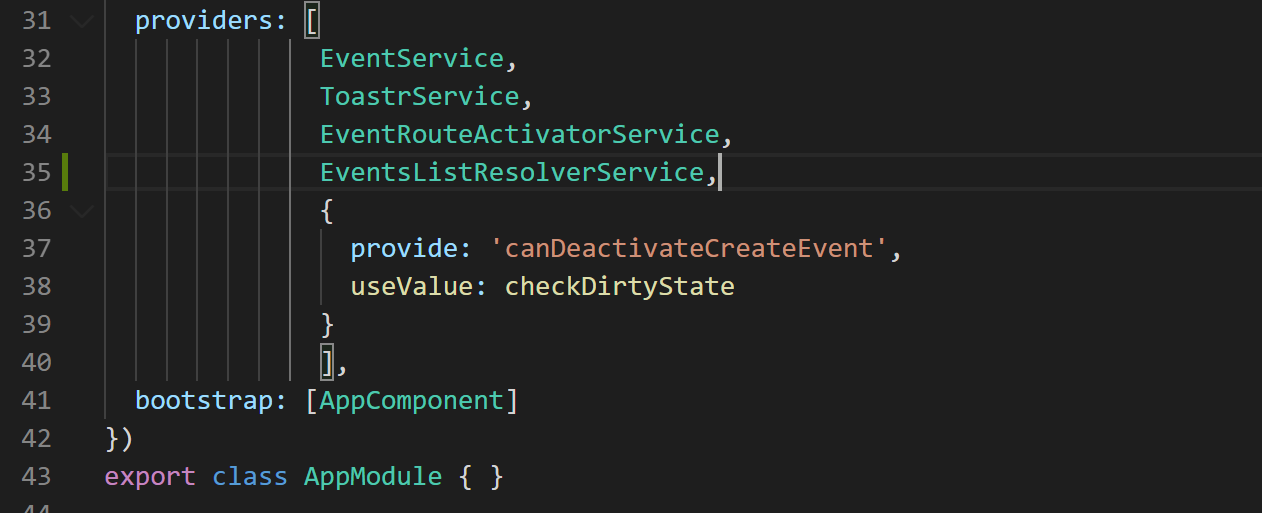
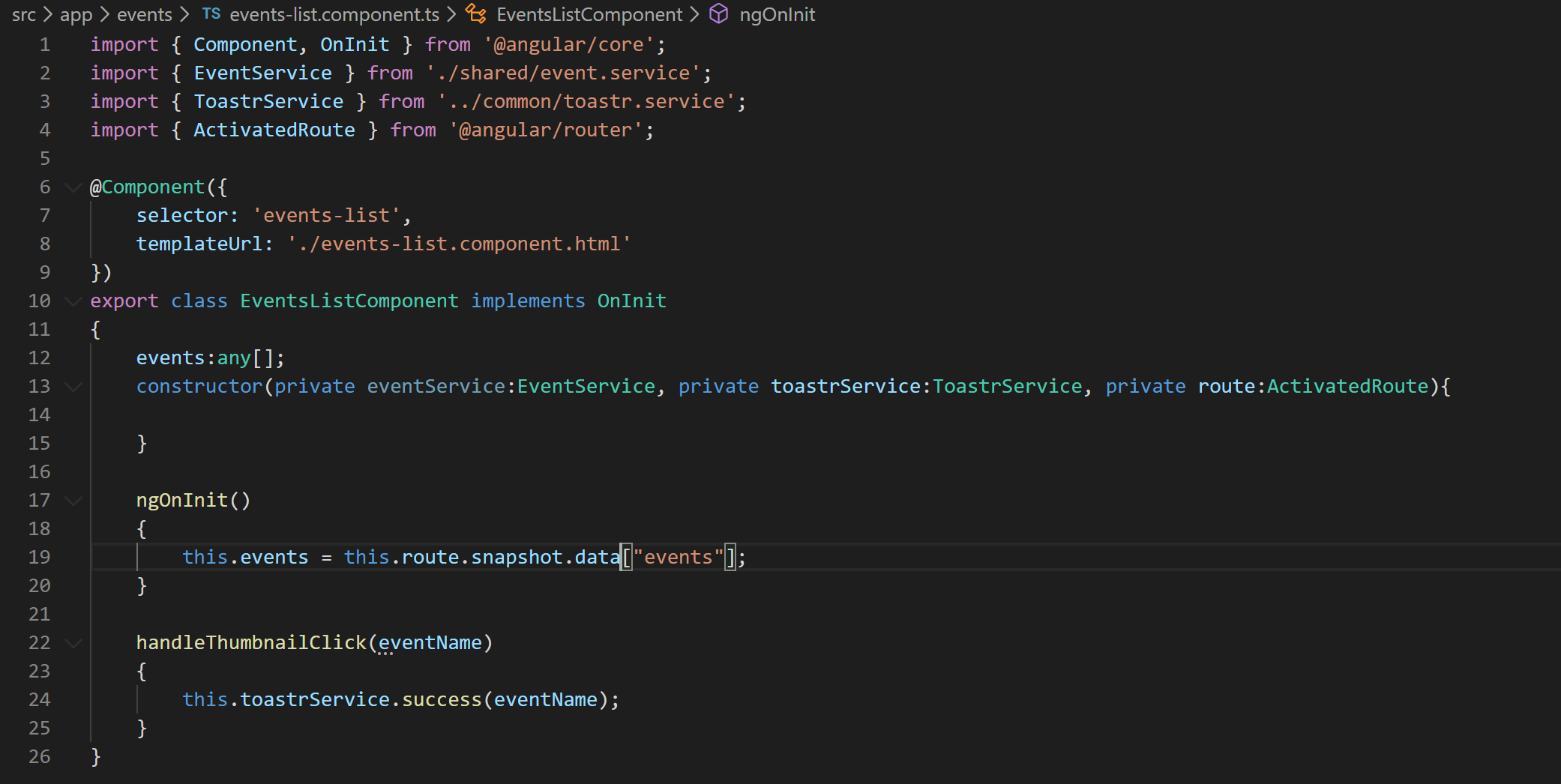
1. We can preload route data using “resolve” route guard prior to loading the component.
2. In real world when we make AJAX calls to service data sometimes takes time to come back and till that time we don’t want to render the component partially. So to prevent that from happening we can use the ‘resolve’ route guard.
3. Lets change our event service a little bit so it returns data after some artificial delay. We will use ‘rxjs’ ‘Subject’ observable and “setTimeOut” function for it like below:
4. 
5. Now if you notice VS Code terminal is reporting an error which is indicating that the “events-list” component is not able to consume the observable. To consume an observable you have to subscribe to it. So lets subscribe to it like below:
6. Now if you go to “Create New Event” page and come back to “Events List” page you will notice that the data loads with a little delay. But the component is still rendering partially. To prevent the component render partially now lets us create the resolver service.
7. Right click on “events” folder and create a file called “events-list-resolver.service.ts”.
8. Lets add the basic shell like below:
9. 
10. Now lets implement ‘Resolve’ interface like below:
11. 
12. Now lets add this service in providers array in app.module.ts like below:
13. 
14. Now lets consume this service in the resolve route guard in app.routing.ts file like below:
15. 
16. With this what we are telling Angular is that someone hits this route call the “EventsListResolverService” to get data and put data in a property in route parameters named as “events”.
17. Now lets consume this route property in our events list component. So go to file “events-list.component.ts” and change code like below:
18. Remove the service call from ngOnInit() method and change the code like below:
19. 
20. Now if you notice “events” page loads after 2 seconds of delay.