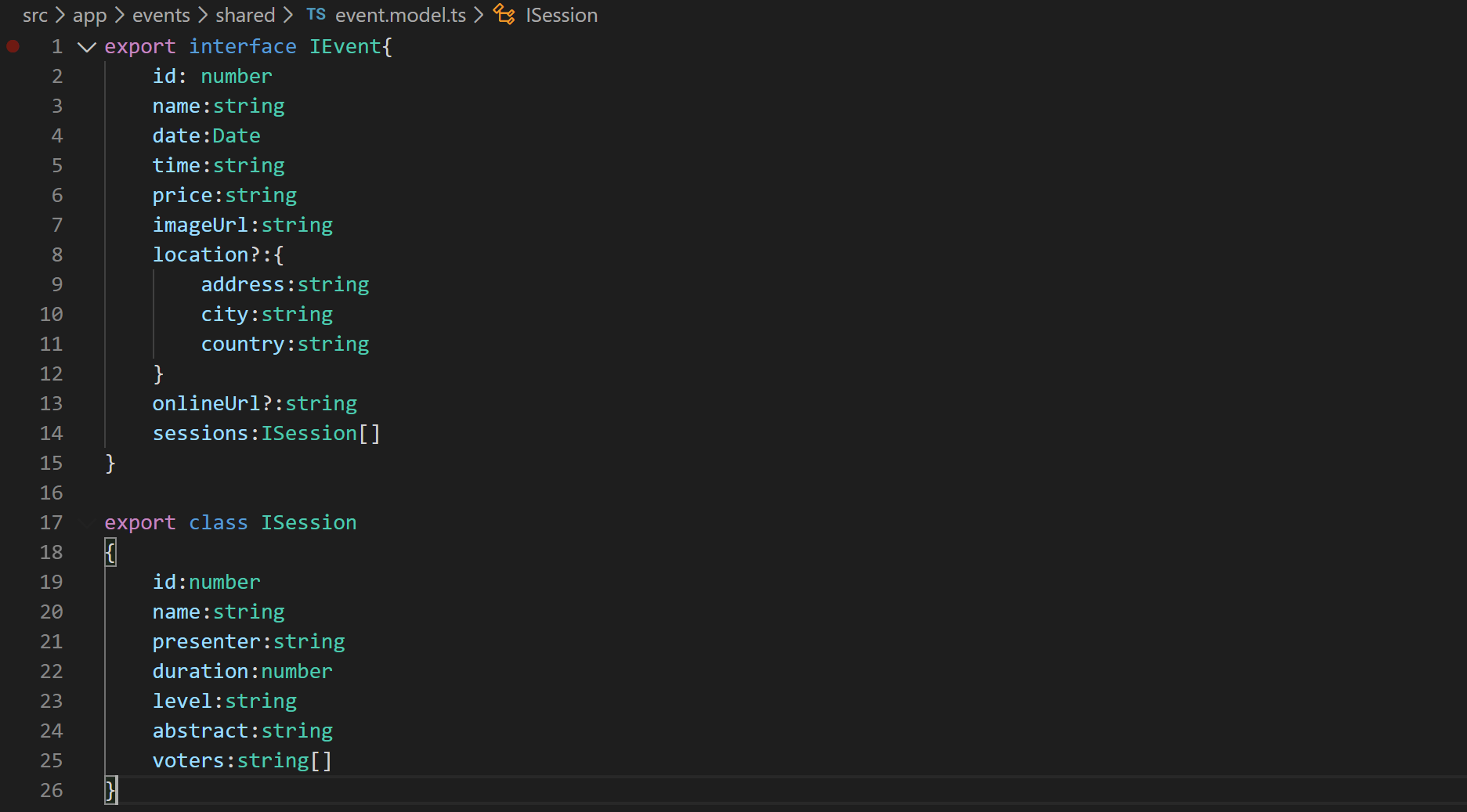
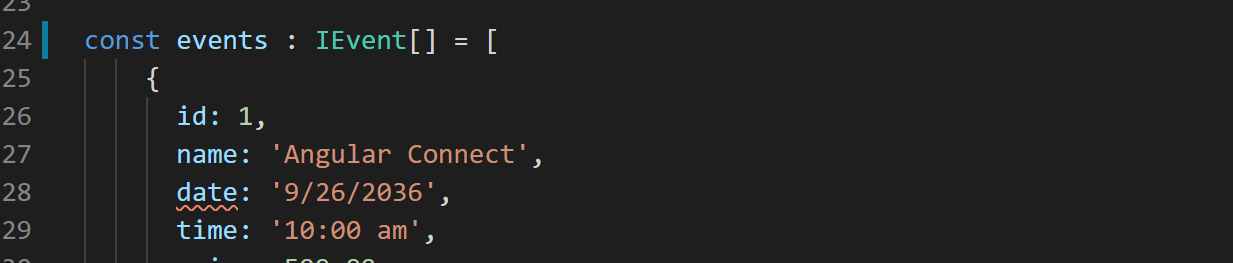
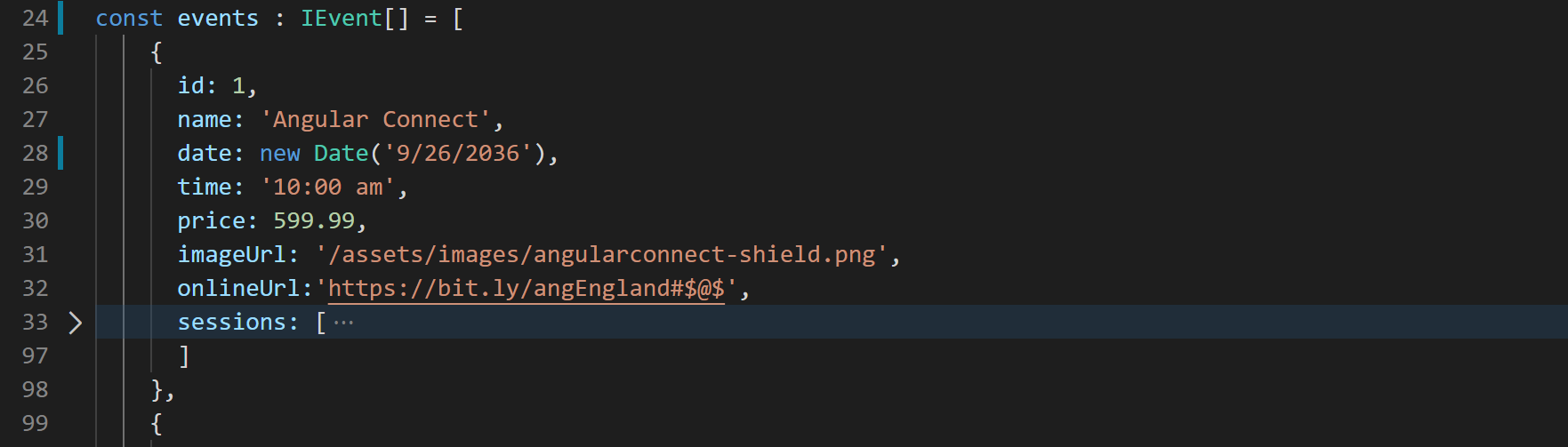
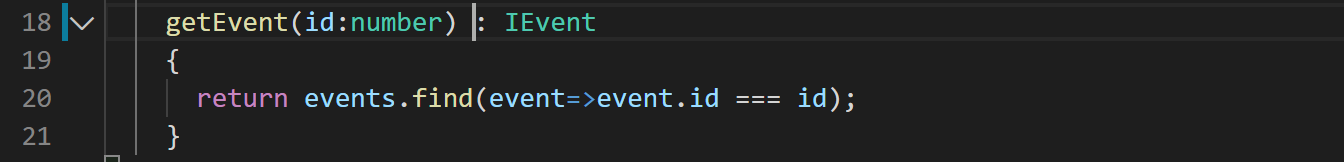
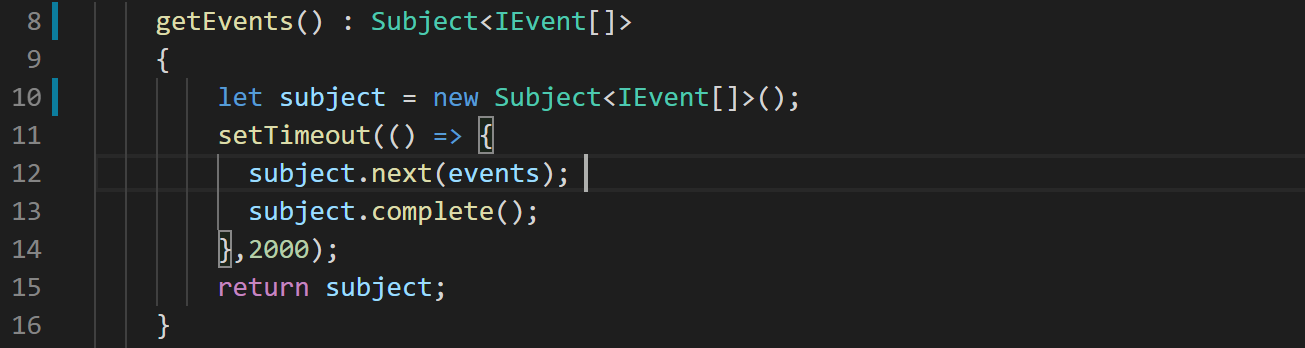
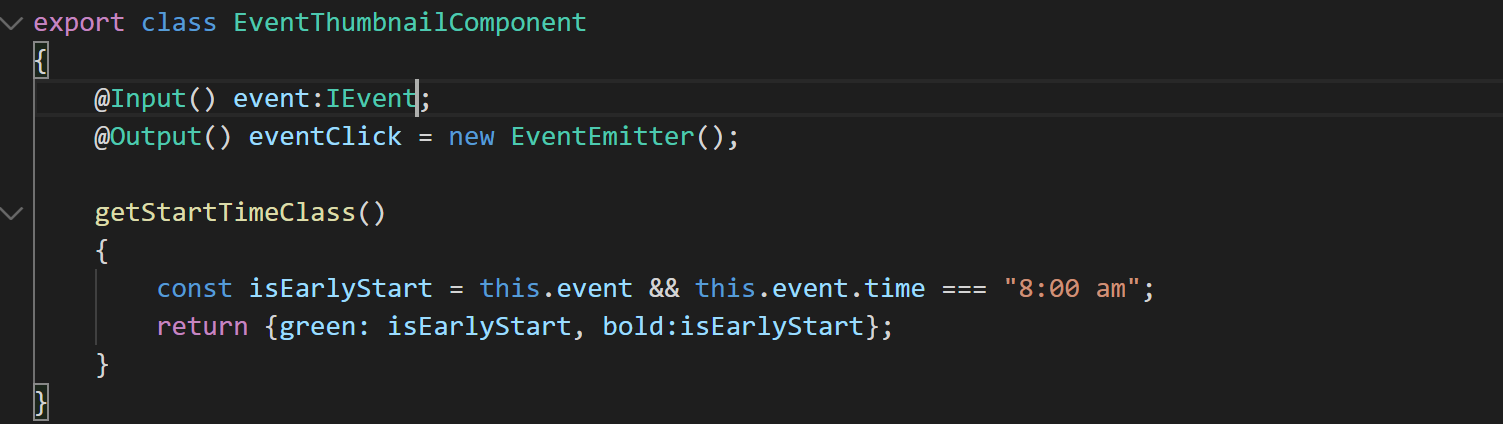
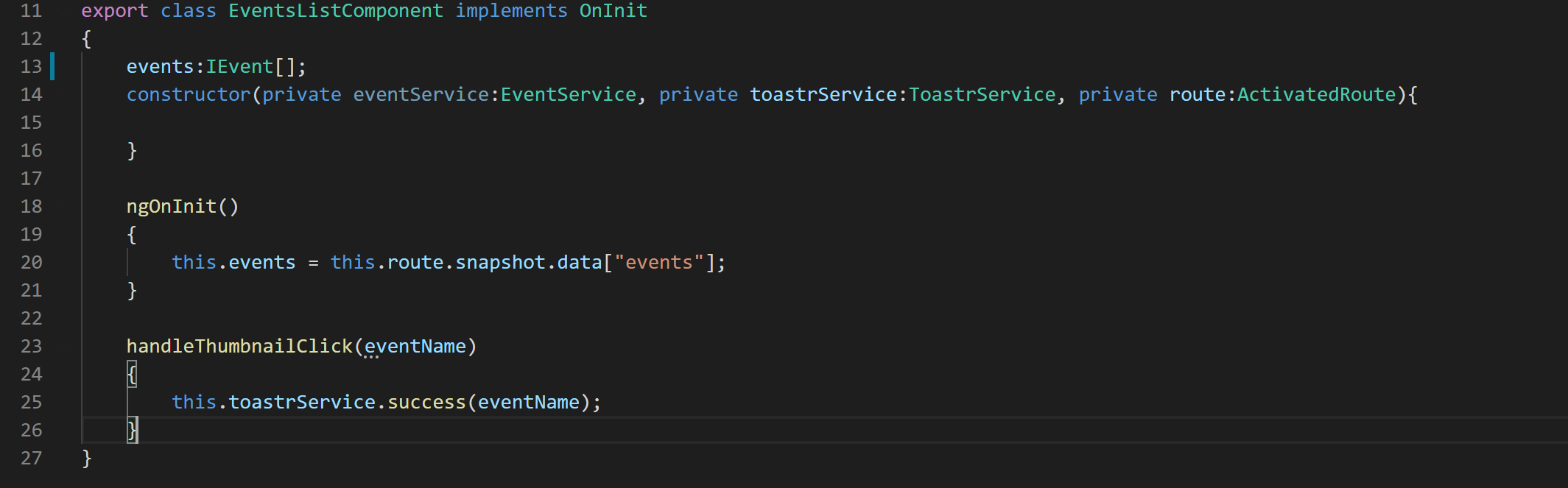
1. Till now we have been using the ‘any’ type extensively. Now that we are about to create forms to store data in our database lets start defining models which will provide us some type safety.
2. So we will first create the “Event” model. Right the “shared” folder in “events” folder and create a file called “event.model.ts”.
3. To create the model we can either use interface or a class it does not matter which one we use.
4. So lets define the model using interface like below:
5. 
6. Now lets start using these models to get the benefit of type safety. Open the file “event.service.ts” and attach the “IEvent” model to the “events” array constant like below:
7. 
8. Now if you notice the VS code terminal, you will see that it is complaining about string being assigned to a Date type. So that is the benefit we get with TypeScript.
9. So let us fix the issue by using Date() constructor like below:
10. 
11. Now let us fix the return type of our methods in this service.
12. Change the return type of “getEvent” method like below:
13. 
14. Change the return type of “getEvetns” method like below:
15. 
16. Now lets use the “IEvent” model in few more places.
17. Open file “event-thumbnail.component.ts” and lets define the type for our input property like below:
18. 
19. Open file “events-list.component.ts” and define the type for the “events” array like below:
20. 
21. Open the file “event-details.component.ts” and define the type for “event” property like below:
22. 