@track

decorator explained

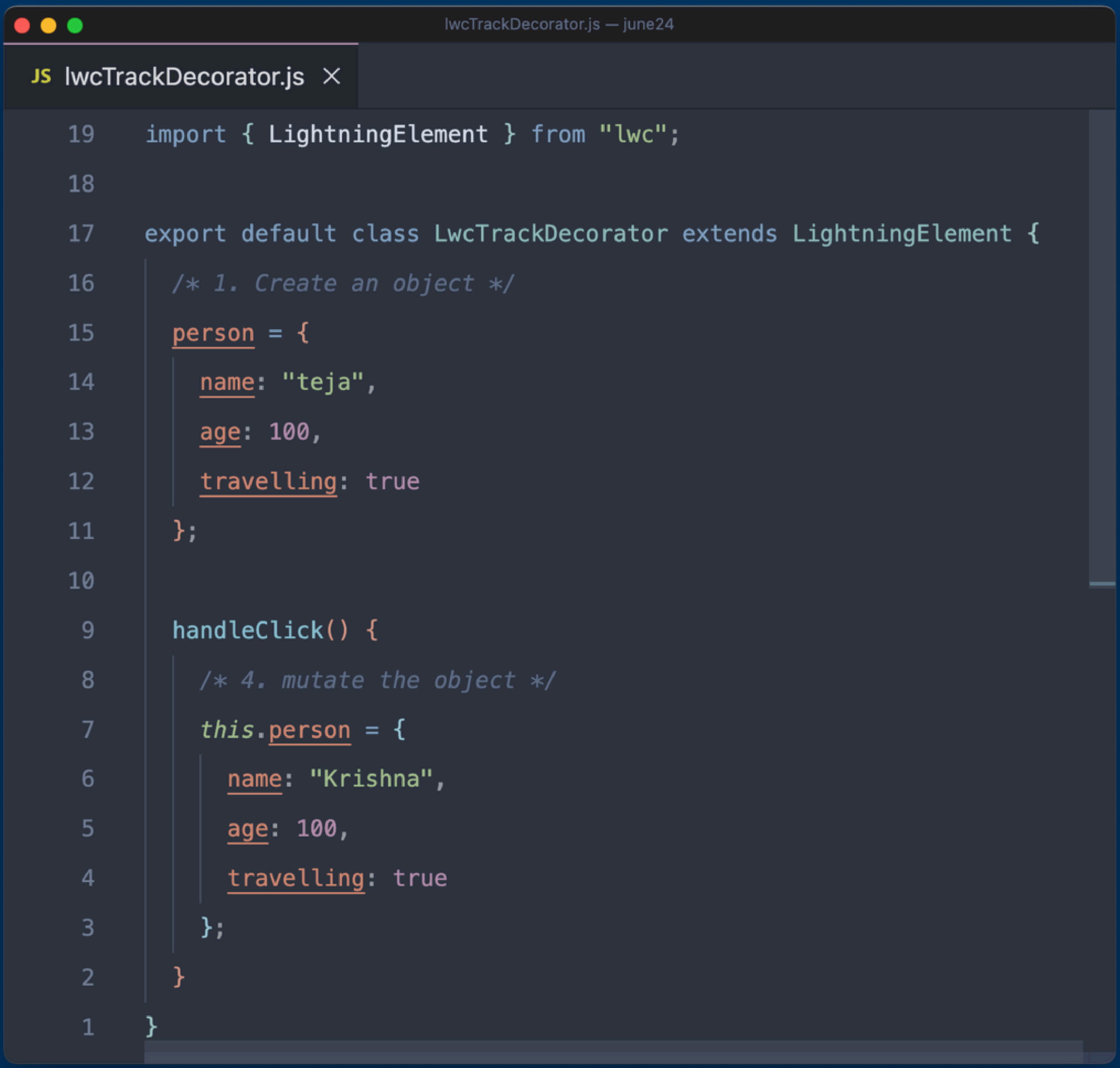
→

join 10K+ experts on SalsesforceCasts.com

Object props

By default all the props are reactive, but, for an object to be reactive we need to redeclare the entire object to the prop even if one key is modified.

If you notice #7 am redeclaring the entire object.

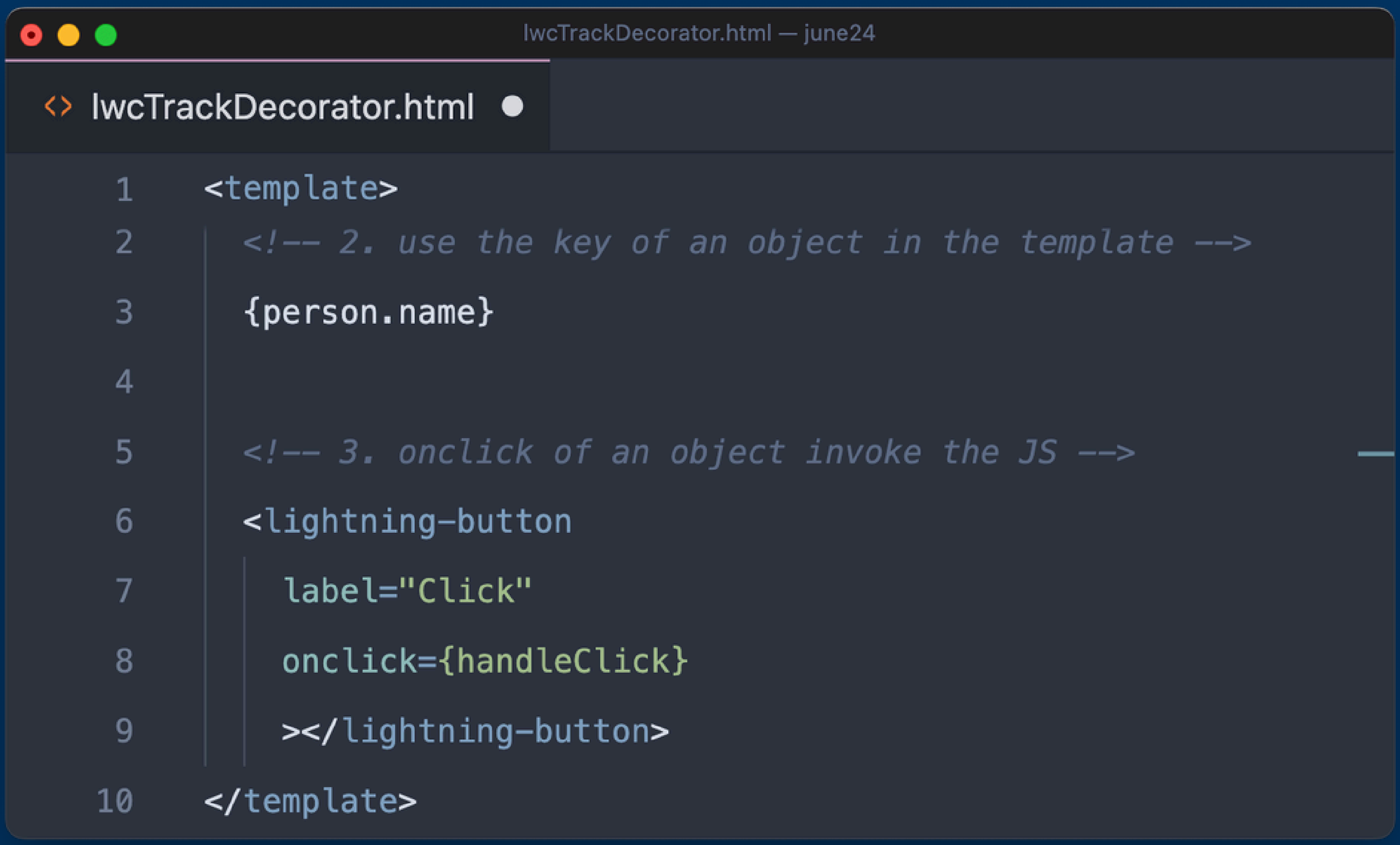


```javascript
19  import { LightningElement } from "lwc";
18
17  export default class LwcTrackDecorator extends LightningElement {
16    /* 1. Create an object */
15    person = {
14      name: "teja",
13      age: 100,
12      travelling: true
11    };
10
 9    handleClick() {
 8      /* 4. mutate the object */
 7      this.person = {
 6        name: "Krishna",
 5        age: 100,
 4        travelling: true
 3      };
 2    }
 1  }
```

template file

Here is the template file that will display the value of the key of the prop.

Here is a typical template file rendering the key of an object.



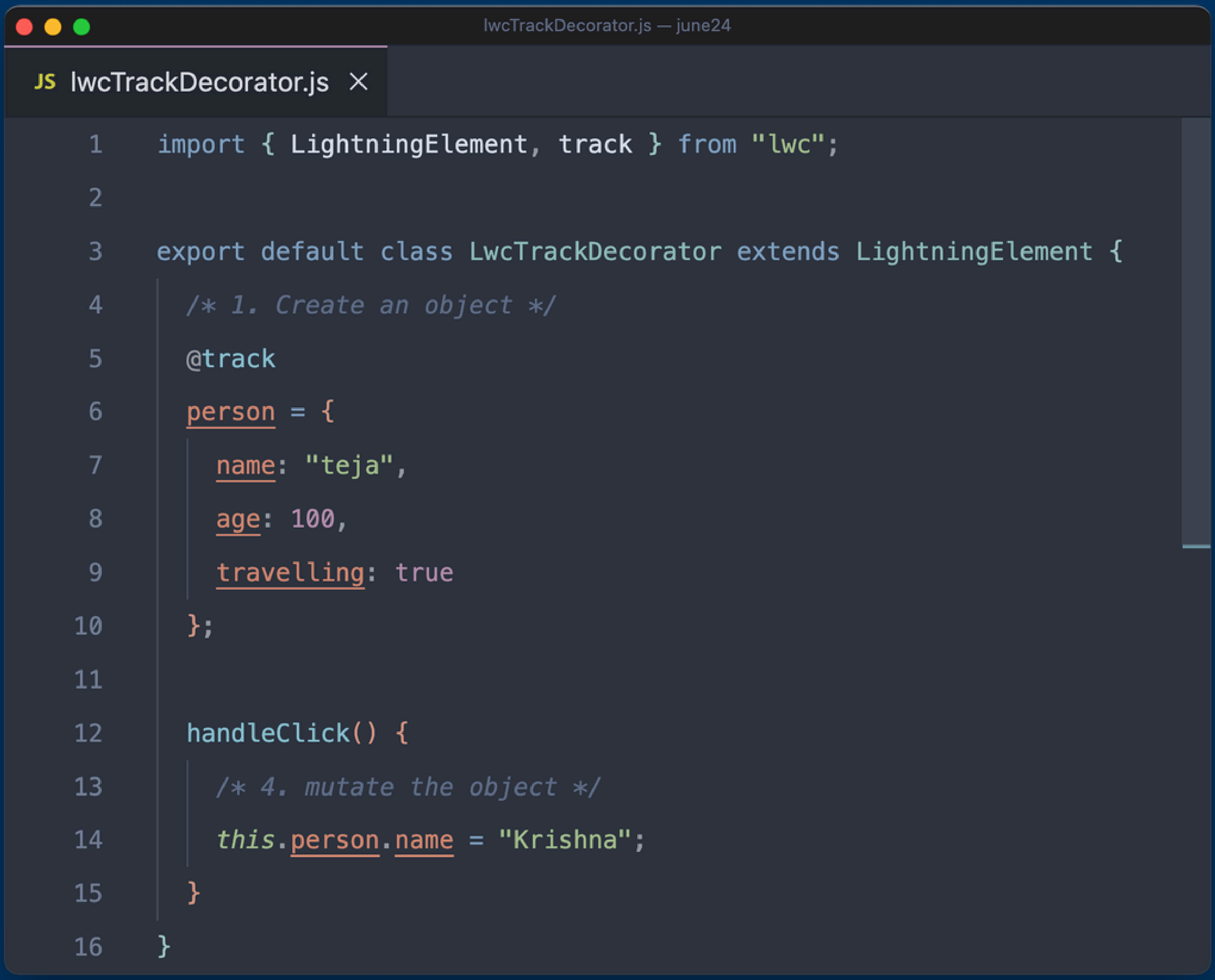
```html
<template>
  <!-- 2. use the key of an object in the template -->
  {person.name}

  <!-- 3. onclick of an object invoke the JS -->
  <lightning-button
    label="Click"
    onclick={handleClick}
    ></lightning-button>
</template>
```

@ track decorator

We can avoid redeclaring the entire object with the help of @track decorator.

Notice how am only updating a key of the object while using @track in # 5.



```
 1  import { LightningElement, track } from "lwc";
 2
 3  export default class LwcTrackDecorator extends LightningElement {
 4    /* 1. Create an object */
 5    @track
 6    person = {
 7      name: "teja",
 8      age: 100,
 9      travelling: true
10    };
11
12    handleClick() {
13      /* 4. mutate the object */
14      this.person.name = "Krishna";
15    }
16  }
```

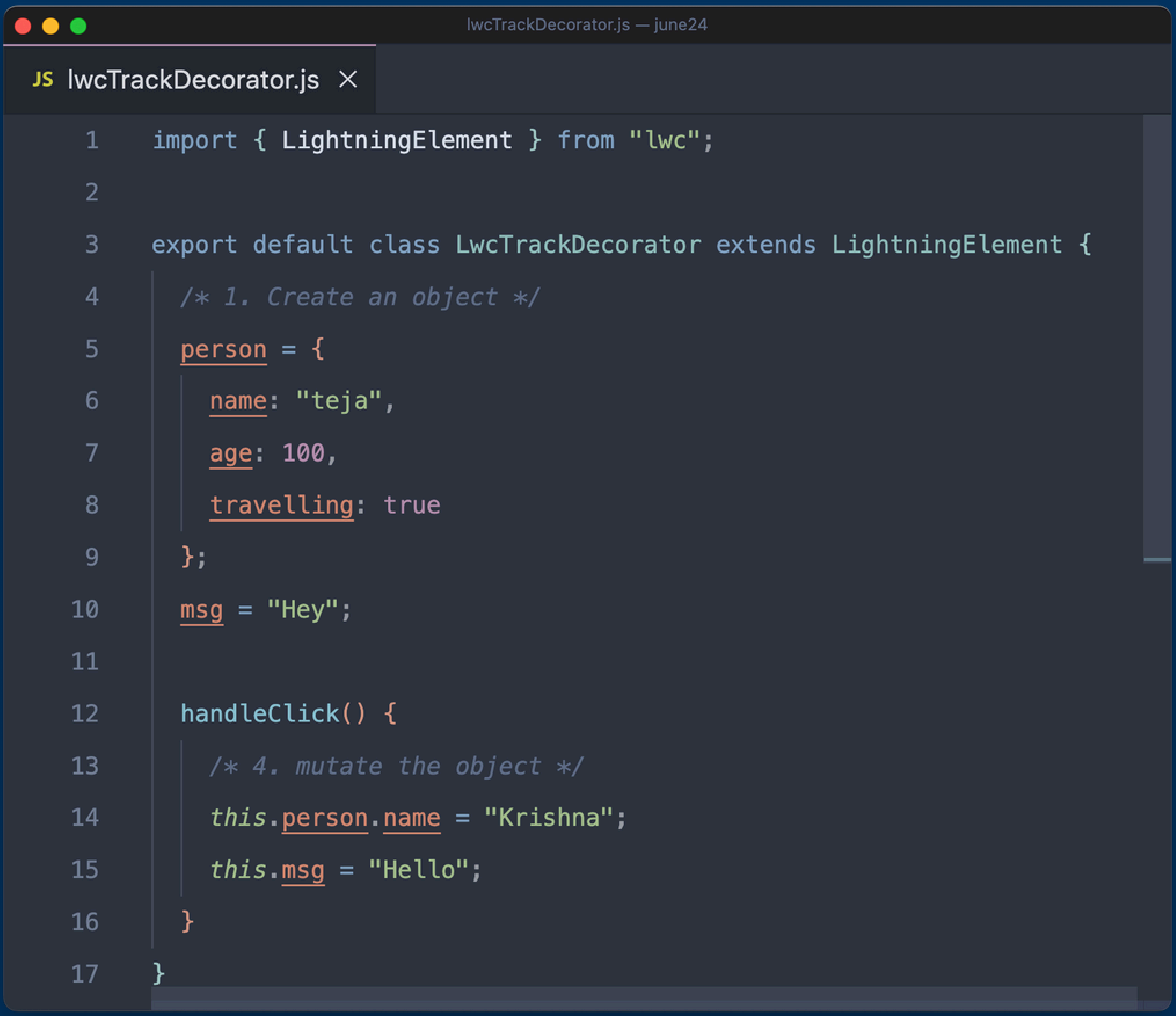
join 10K+ experts on SalsesforceCasts.com

Corner case

When am mutating more than one prop, we don't need @track explicitly.

But I strongly recommend using it, as it might help others who are reading your coding.

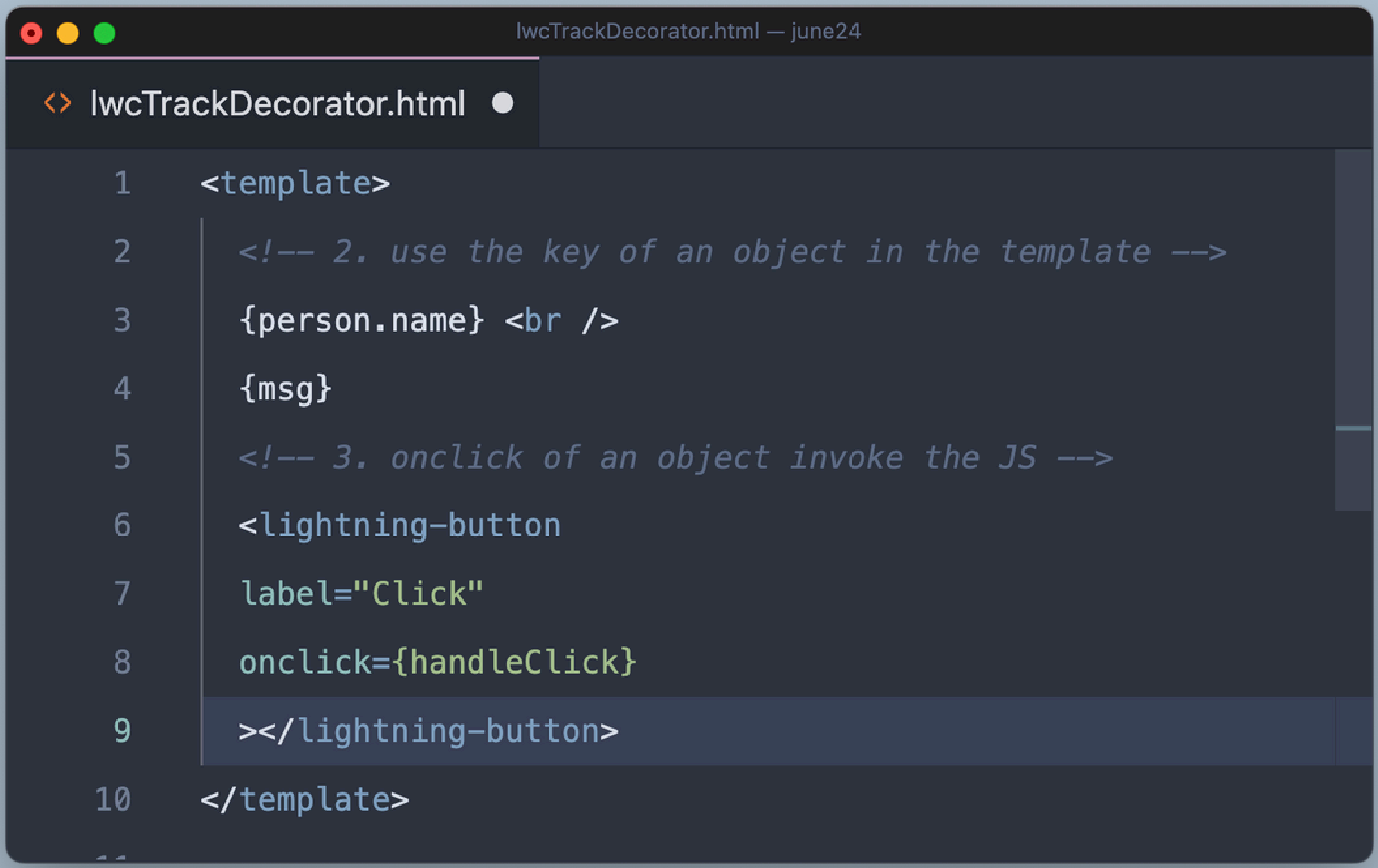
Check #15 and also notice am not using @track.



```
1   import { LightningElement } from "lwc";
2
3   export default class LwcTrackDecorator extends LightningElement {
4     /* 1. Create an object */
5     person = {
6       name: "teja",
7       age: 100,
8       travelling: true
9     };
10    msg = "Hey";
11
12    handleClick() {
13      /* 4. mutate the object */
14      this.person.name = "Krishna";
15      this.msg = "Hello";
16    }
17  }
```

join 10K+ experts on SalsesforceCasts.com

Do not forget to use both the props on the template file.



```html
<template>
  <!-- 2. use the key of an object in the template -->
  {person.name} <br />
  {msg}
  <!-- 3. onclick of an object invoke the JS -->
  <lightning-button
  label="Click"
  onclick={handleClick}
  ></lightning-button>
</template>
```

join 10K+ experts on SalsesforceCasts.com

Like it?

Leave a comment below

join 10K+ experts on SalsesforceCasts.com