

Event Binding Lab

Shipping

1. Edit your shipping.component.ts file. Add a new method called `GetBestLocation(orderLine)`. Note that it will receive an `orderLine` object. For now it should just set `orderLine.locationID` to "01A1A" and then `console.log` it.
2. Edit your shipping.component.html and find the 'Get best location' button. Add an angular click event handler to it. When fired, it should call the "`GetBestLocation()`" method you just wrote. Don't forget to pass in the current `orderLine`.
3. Run and test. When you click your button, the button should be replaced by a location.
4. Wire up the "Mark as shipped" button to run a method called `MarkAsShipped(order)` which receives in an order object. This method should change the order's status from 0 to 1.
5. Wire up the "Problem" button. It should run a `MarkWithProblem()` method which also receives an order object. It should set the order's status to 2. And it should change the instructions to say "Don't pick this order. It needs to be looked at by a supervisor."
6. Run and test. Look at the orders in the console and make sure that the status is indeed changing.

Receive inventory

At this point we have worked on several components but one has been neglected, the receiving component. Let's fill that in now.

7. First, notice that in `setup/assets/html`, you have a starter for this component. Copy the html that is in there.
8. Open `receive-product-component.html` and paste the html from that other file.
9. Run and test. You should see a fine-looking component but it doesn't really do anything. It is just static HTML.

Our client has decided that the users should enter the package tracking number for every order we receive. We've decided that we will hide everything else on the page until the tracking number is entered.

10. Create a boolean class variable called `showForm`.
11. Make everything except the tracking number section be hidden until this variable is true.
12. Create an event handler for the button click. It should pass the tracking number into that method and set `showForm` to true.
13. Run and test. Did you make the rest of the component show when you click the button?
14. Notice that there's a button labeled "Receive Product". When the user clicks it, call a method called `receiveProduct()`. For now it can just `console.log()`. We'll make it do something meaningful later.
15. Make sure that when you click the button something is logged to the console.