

## Step 20 ♥

Now you need to start displaying the team's information on the screen.

Below your destructuring assignments, assign the sport variable to typeOfSport.textContent.

Once you complete that task, you should see the result in the preview window.

178 typeOfSport.textContent = sport;

179

Submit and go to next challenge

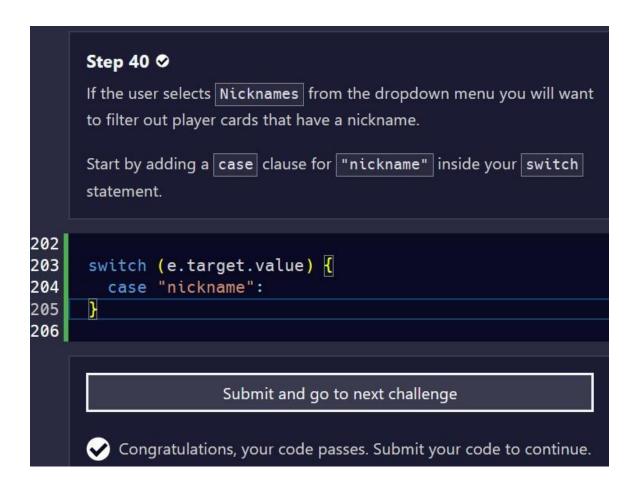
Congratulations, your code passes. Submit your code to continue.

# Step 30 ♥ Below the h2 element, add a paragraph element with the text Position: and an embedded expression that contains the position parameter.

```
185
186
      playerCards.innerHTML += arr.map(
187
        ({ name, position, number, isCaptain, nickname }) =>
188
           <div class="player-card">
189
190
              <h2>${isCaptain ? "(Captain)" : ""} ${name}</|
              Position: ${position}
191
192
            </div>
193
194
195
      );
196
```

Submit and go to next challenge

Congratulations, your code passes. Submit your code to continue.



# Step 47 ♥

The final step is to add a default clause if none of the other case clauses match the user selection.

For the default clause, call the setPlayerCards function without any arguments passed in.

Test out your dropdown menu, and you should see the player cards be filtered out by position or nickname.

Congratulations on completing the football team cards project!

### 225 default:

setPlayerCards(); 226

227

Submit and go to next challenge



Congratulations, your code passes. Submit your code to continue.

# Learn Modern JavaScript Methods by Building Football Team Cards

One common aspect of building web applications: processing datasets, and then outputting information to the screen. In this sports team cards project, you'll learn how to work with DOM manipulation, object destructuring, event handling, and data filtering.

This project will cover concepts like switch statements, default parameters, Object.freeze(), the map() method, and more.

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47			