

# Final Project: Part two

## Ajax

At this point you have a page that displays a list of persons. But this list is being typed in by hand from the form. Let's read those persons from an API by making an Ajax call.

### Examining the API

11. First, open a browser and navigate to <https://randomuser.me/api/?results=5>
12. Take a look at what is being returned. It may be helpful to open the browser's developer tools and study the network tab. If you notice, the value coming back is pure JSON. Remember that URL because we'll come back and use it in just a minute.

### Crafting a call to the API

13. Open your index.js file and create a new method called `getPersons()`.
14. Call `getPersons()` on an event. You can either:
  - a. Add a "Get Persons" button and call `getPersons()` on button click or
  - b. Call `getPersons()` on the `DOMContentLoaded` event.
15. Run and test, making sure that `getPersons()` is indeed being called.
16. In `getPersons()`, do a fetch from the URL we visited above.

### Processing the API response

17. Remember from the lecture that fetch returns a promise. So, after your fetch call, do this:

```
.then(res => res.json())  
.then(console.log);
```

That will convert the request string to a true JSON object and print it to the browser's console.

18. Run and test again.
19. Take a look in the developer tools again and examine the data being logged. You should have an object that has our list of people inside it. What is the name of that key? \_\_\_\_\_
20. Chain another `.then()` function that will take the response object you logged in the step above and return only the property you discovered above. (Hint: you'll add one line of code and it will be another `.then()`)
21. Run and test. If you've done it right, every time you refresh, you'll see a new list of random people in the developer tools console.

### Displaying the data

22. Now that you have a list of persons, display those on your index.html page. (Hint: Re-use the portion of your code that displays a list of persons on the page).

### Refactoring

23. Change your `getPersons` method to receive in a "numberOfPeople" parameter with a default of 5. Use it in `getPersons` to fetch that number of people instead of the hardcoded number you have in there already.
24. Go through and pretty up your code, formatting it, tightening it, using good shortcuts wherever possible (like destructuring, spreading, object property shorthands, etc).