Handle The Response

Ok, you've learned about making a Fetch request, and you've sent a few of them off...but nothing happened because we didn't tell our code to handle the response. Let's get our code ready to handle the response.

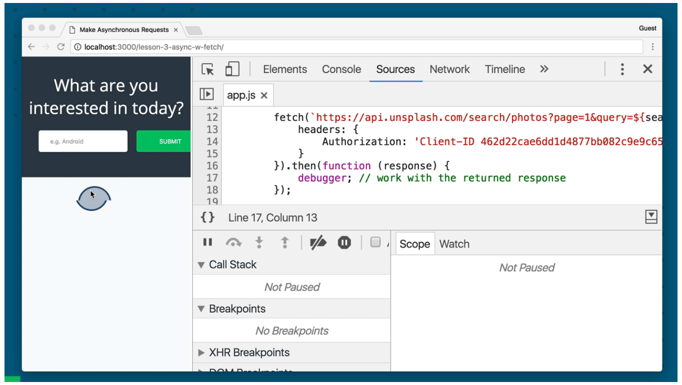
Remember that Fetch is Promise-based. This means that when we fire of the Fetch request, it will automatically return a promise that we can use to listen for the response.

> ### Javascript Promises Reminder
>
> Dealing with the returned data from a Fetch request is all done with Promises, so if any of this looks confusing or you don't know how `.then()` works or why it's needed, check out our course on JavaScript Promises.

Since a Fetch request returns a Promise, then all you have to do is call `.then()` on that Promise.

```
fetch(`https://api.unsplash.com/search/photos?page=1&query=${searchedForText}`, {
    headers: {
        Authorization: 'Client-ID abc123'
    }
}).then(function(response) {
    debugger; // work with the returned response
});
```

If you haven't already, put the code above in our JavaScript file and search for something. Because we added a `debugger` line inside the `.then()` function, the code will pause on the `debugger` line when the response is finally returned.



```
        fetch(`https://api.unsplash.com/search/photos?page=1&query=${sea
            headers: {
                Authorization: 'Client-ID 462d22cae6dd1d4877bb082c9e9c65
            }
        }).then(function (response) {
            debugger; // work with the returned response
        });
```

Browser showing the app with DevTools loaded. A search for "trees" is made. The browsers pauses at the debugger line. The response is a Response object.

QUIZ QUESTION

We've successfully made the request, and you should be able to see the response in your console. Which property has the actual JSON data of the images?

○ `.data`

○ `.images`

○ `.response`

Mentorship
Get support and stay on track 1

Lesson 3:
Ajax with Fetch

Handle The Response

none of the above

SUBMIT

NEXT

Mentorship

Get support and stay on track

1