

Default Parameters

Default Parameters are a great addition to the language. They allow you to, well, add default parameters. And this solves a very simple problem.

```
function getData(url) {  
  url = url || 'http://api.data.com';  
  return fetch(url)  
    .then(res => res.json);  
}
```



Consider the above, a simple function to get data from a source. If a URL is passed in, it will go and get the data from that URL. However if no URL is provided, it will get the data from `http://api.data.com`, a default source.

With the new ES6 Default Params we are able to add that to our function definition and make it more apparent what is happening.

```
function getData(url = 'http://api.data.com') {  
  return fetch(url)  
    .then(res => res.json);  
}
```



This really cleans things up. With the ability to create Default Params, we can have more flexible and reliable functions.

The way Default Params works is that if nothing is passed in it will take the value stored in the function definition, in this case `url = 'http://api.data.com'`. If you do pass in a value for that position it will override the section.

When working with Default Params you are able to mix and match, allowing you to create easy-to-understand optional parameters for your functions.

```
function dataHandler(method, url = 'http://api.data.com') {  
  return fetch(url)
```



```
return fetch(uri)
  .then(res => res.json);
}
```

```
dataHandler('POST');
dataHandler('GET');
```