

I'm currently in love with **Promises**. Somehow, only now I get the main concept behind that and the benefits. Mainly, because most of the documentation is a bit too detailed about that.

Let me sum it up for you:

If you're developing asynchronous Javascript (which you should be, when you're actually developing Javascript!), most of the time you're passing callback functions to asynchronous functions:

```
function async(callback) {
  // do something
  if (there_is_an_error) {
    callback(error);
  } else {
    callback(null, { something: 'return' }); // everything's fine
  }
}

function callAsync() {
  async(
    (error) => {
      if (error) {
        // handle error
      } else {
        // keep on working
      }
    }
  );
}
```

This gets very ugly if you're **chaining multiple callbacks after another**. There are helpers for that - for example the wonderful **Async.js** (and that's the first choice when it comes to asynchronous calls in loops), but it still gets ugly.

Promises, however, allow you to chain callbacks and handling errors in a much more cleaner way:

```
function async() {
  return new Promise(function (fulfill, reject) {
    // do something
    if (there_is_an_error) {
      reject(error);
    } else {
      fulfill({ something: 'return' }); // Everything's fine
    }
  });
}

function callAsync() {
  async()
  .catch(
    (error) => {
      // handle error
    }
  )
  .then(
    () => {
      // keep on working
    }
  );
}
```

Well, this example is not that good to show you the benefits, but think about five callbacks in a row. If you're using a column limit in your code conventions, you'll be effectively writing on the right border of your screen. Not with Promises.

To wrap this post up, I have a present for all folks using **node-rest-client**. I've made a Promise wrapper around it using **Bluebird** and I've called it **node-rest-client-promise**. Just grab it using „npm install node-rest-client-promise -save“ instead of node-rest-client and use it like this:

```
var nodeRestClient = require('node-rest-client-promise');

var client = nodeRestClient.client({
  // node-rest-client.Client-options
});

client.getPromise(
  'https://www.google.de'
).catch(
  (error) => {
    // there was an error
  }
).then(
  () => {
    // everything's great.
  }
);
```

Have fun.