

build passing

aggregate-error

Create an error from multiple errors

Install

```
$ npm install aggregate-error
```

Usage

```
const AggregateError = require('aggregate-error');

const error = new AggregateError([new Error('foo'), 'bar', {message: 'baz'}]);

throw error;
/*
AggregateError:
  Error: foo
    at Object.<anonymous> (/Users/sindresorhus/dev/aggregate-error/example.js:3:33)
  Error: bar
    at Object.<anonymous> (/Users/sindresorhus/dev/aggregate-error/example.js:3:13)
  Error: baz
    at Object.<anonymous> (/Users/sindresorhus/dev/aggregate-error/example.js:3:13)
  at AggregateError (/Users/sindresorhus/dev/aggregate-error/index.js:19:3)
  at Object.<anonymous> (/Users/sindresorhus/dev/aggregate-error/example.js:3:13)
  at Module._compile (module.js:556:32)
  at Object.Module._extensions..js (module.js:565:10)
  at Module.load (module.js:473:32)
  at tryModuleLoad (module.js:432:12)
  at Function.Module._load (module.js:424:3)
  at Module.runMain (module.js:590:10)
  at run (bootstrap_node.js:394:7)
  at startup (bootstrap_node.js:149:9)
*/

for (const individualError of error) {
  console.log(individualError);
}
//=> [Error: foo]
//=> [Error: bar]
//=> [Error: baz]
```

API

AggregateError(errors)

Returns an `Error` that is also an [Iterable](#) for the individual errors.

errors

Type: `Array<Error|Object|string>`

If a string, a new `Error` is created with the string as the error message.

If a non-Error object, a new `Error` is created with all properties from the object copied over.