



MODULE 6

WRITING NODE MODULES

Node Modules

- > Node.JS - a Common.JS Module Implementation
- > Your code uses **require** to include modules.
- > Modules use **exports** to make things available.

COMMON.JS

- > An ecosystem for JavaScript outside the browser
 - Modules
 - Promises
 - Binary
 - Filesystem
 - Console
 - System
 - Testing

What Server Side JavaScript needs

"What I'm describing here is not a technical problem. It's a matter of people getting together and making a decision to step forward and start building up something bigger and cooler together."

Kevin Dangoor

COMMON.JS Implementations

- > MongoDB
- > CouchDB
- > SproutCore
- > Node.JS

The Simplest Module

```
// hello.js  
console.log('Hello World');
```

```
// app.js  
require('hello.js');
```

Define a Global Module

```
// foo.js
foo = function () {
  console.log('foo!');
}
```

```
// app.js
require('./foo.js');
foo();
```

Define a Global Module

```
// foo.js
foo = function () {
  console.log('foo!');
}
```

```
// app.js
require('./foo.js');
foo();
```

don't pollute the global space

Export an Anonymous Function

```
// bar.js
module.exports = function () {
  console.log('bar!');
}
```

```
// app.js
var bar = require('./bar.js');
bar();
```

Export a Named Function

```
// fiz.js
exports.fiz = function () {
  console.log('fiz!');
}
```

```
// app.js
var fiz = require('./fiz.js').fiz;
fiz();
```

Export an Anonymous Object

```
// buz.js
var Buz = function () {};
Buz.prototype.log = function () {
  console.log('buz!');
};
module.exports = new Buz();
```

```
// app.js
var buz = require('./buz.js');
buz.log();
```

Export a Named Object

```
// baz.js
var Baz = function () {};

Baz.prototype.log = function () {
  console.log('baz!');
};

exports.Baz = new Baz();
```

```
// app.js
var baz = require('./baz.js').Baz;
baz.log();
```

Export an Anonymous Prototype

```
// doo.js
var Doo = function () {}

Doo.prototype.log = function () {
  console.log('doo!');
}

module.exports = Doo;
```

```
// app.js
var Doo = require('./doo.js');
var doo = new Doo();
doo.log();
```

Export a Named Prototype

```
// qux.js
var Qux = function () {}

Qux.prototype.log = function () {
  console.log('baz!');
};

exports.Qux = Qux;
```

```
// app.js
var Qux = require('./qux.js').Qux;
var qux = new Qux();
qux.log();
```

Pros and Cons

- > Named exports
 - One module, many exported things
- > Anonymous exports
 - Simpler client interface