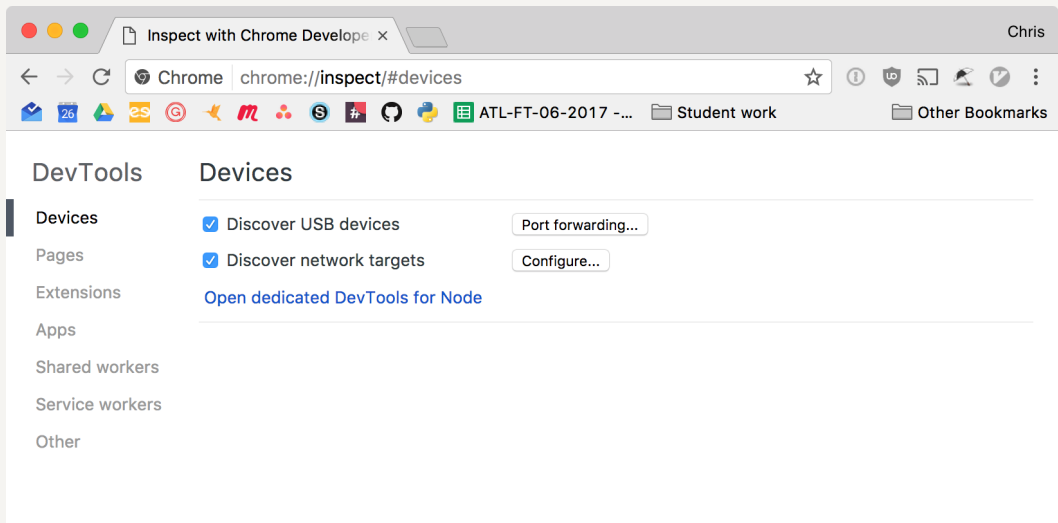


Debugging Node.js

chrome://inspect/#devices

Click "Open dedicated DevTools for Node"



Run your Node program in debug mode

```
node --inspect-brk index.js
```

Using Chrome Dev Tools to debug Node.js

The screenshot displays the Chrome DevTools interface with the 'Node' tab selected in the top bar. The 'Sources' panel on the left shows a file named 'index.js' with the following code:

```
1 (function (exports, require, module, __filename, __dirname) { const Calculator = require('./calculator');
2
3   function main() {
4
5     let calc = new Calculator();
6     calc.add(NaN, Infinity);
7   }
8   main();
9
10
11 });
```

The line number 1 is highlighted, and the status bar at the bottom indicates 'Line 1, Column 11'. The right-hand pane shows the 'Debugger paused' state. The 'Call Stack' section lists the following frames:

- (anonymous) index.js:1
- Module._compile module.js:566
- Module._extensions.js module.js:580
- Module.load module.js:503
- tryModuleLoad module.js:466
- Module._load module.js:458
- Module.runMain module.js:605
- startup bootstrap_node.js:158
- (anonymous) bootstrap_node.js:575

The 'Scope' section shows the local variables:

- exports: Object
- main: function main()
- module: Module
- require: function require(path)
- this: Object
- __dirname: "/Users/chrisaquino/DigitalCrafts/tdd-in"

Add breakpoints as usual!

The screenshot shows the VS Code Developer Tools interface with the 'Sources' tab active. A breakpoint is set on line 6 of the file `index.js`. The code in the editor is as follows:

```
1 (function (exports, require, module, __filename, __dirname) { const Calculator = require('./calculator');
2
3 function main() {
4
5   let calc = new Calculator(); calc = Calculator {}
6   calc.add(NaN, Infinity);
7 }
8 main();
9
10
11 });
```

The status bar at the bottom indicates the cursor is at `Line 6, Column 3`.

On the right side of the interface, the 'Paused on breakpoint' panel is open, showing the following information:

- Threads:** Node: file:///Users/chrisaquino/DigitalCrafts/tdd-intro/index.js paused
- Call Stack:**
 - main index.js:6
 - (anonymous) index.js:8
 - Module._compile module.js:566
 - Module._extensions..js module.js:580
 - Module.load module.js:503
 - tryModuleLoad module.js:466
 - Module._load module.js:458
 - Module.runMain module.js:605
 - startup bootstrap_node.js:158
 - (anonymous) bootstrap_node.js:575
- Scope:**
 - Local:**
 - calc: Calculator
 - this: global
 - Closure

Use the Console to run code in context

The screenshot displays the Chrome DevTools interface with the 'Developer Tools - Node' title. The 'Sources' panel is active, showing a file named 'index.js' with the following code:

```
1 (function (exports, require, module, __filename, __dirname) { const Calculator = require('./calculator');  
2  
3 function main() {  
4  
5   let calc = new Calculator(); calc = Calculator {};  
6   calc.add(NaN, Infinity);  
7 }  
8  
9 } Line 6, Column 3
```

The code is paused on line 6, column 3. The right-hand pane shows the 'Paused on breakpoint' status, with a 'Threads' section indicating 'Node.js Main Context' is paused. Below this, 'Watch' and 'Call Stack' sections are visible.

The 'Console' panel is also open, showing the following output:

```
> calc  
< Calculator {}  
> calc.add(1, 1)  
< 2  
> |
```

Using npm scripts

```
"scripts": {  
  "start": "node index.js",  
  "dev": "nodemon index.js",  
  "test": "mocha",  
  "test-dev": "mocha -w",  
  "debug": "node --inspect-brk index.js"  
},
```

Also works with nodemon

```
"scripts": {  
  "start": "node index.js",  
  "dev": "nodemon index.js",  
  "test": "mocha",  
  "test-dev": "mocha -w",  
  "debug": "node --inspect-brk index.js",  
  "debug-dev": "nodemon --inspect-brk index.js",  
},
```