Instructions to setting up [Typescript](https://www.typescriptlang.org/).

Install with [npm](https://docs.npmjs.com/)

$ npm install -g typescript

$ npm install -g webpack

$ npm install -g ts-loader // links webpack to complie typescript

$ npm install -g jquery // to use jquery in your project

$ npm install -g http-server // to have a local server run

==========

Configure [Typescript](https://www.typescriptlang.org/)

1. Create new Typescript file main.ts:
2. class SweetSweetClass {
3. constructor() {
4. console.log("Even sweeter")
5. }
6. }
7. let basil = new SweetSweetClass()
   1. Setup tsconfig.json:

{

"compilerOptions": {

"module": "commonjs",

"outDir": "dist/",

"noImplicitAny": true,

"removeComments": true,

"preserveConstEnums": true

},

"include": [

"\*"

],

"exclude": [

"node\_modules",

"\*\*/\*.spec.ts"

]

}

tsc --watch

Notice that the directory dist and the file main.js were created.

* 1. Setup HTML Page:

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <title></title>
5. </head>
6. <body>
7. <script src='/dist/main.js'></script>
8. </body>
9. </html>

Run

$ http-server -c-1

or

$ http-server

Using the flag -c-1 makes the browser never cache the files.

Open up the **Javascript Console** and find the statement "Even sweeter"

Run & Compile with Webpack

Setup webpack.config.js:

1. var path = require('path');
2. module.exports = {
3. entry: './main.ts',
4. resolve: {
5. extensions: ['.webpack.js', '.web.js', '.ts', '.js']
6. },
7. module: {
8. loaders: [
9. { test: /\.ts$/, loader: 'ts-loader' }
10. ]
11. },
12. output: {
13. filename: 'bundle.js',
14. path: path.resolve(\_\_dirname, 'dist')
15. }
16. }

Run webpack:

Non minified

$ webpack --watch

As minified

$ webpack --watch --optimize-minimize

ERROR in Entry module not found: Error: Can't resolve 'ts-loader'

Use below commands incase you get above error

npm install ts-loader

npm link typescript

webpack --watch --optimize-minimize

Add new [Typescript](https://www.typescriptlang.org/) file called getcoffee.ts:

1. export class MustHaveCoffee {
2. constructor() {
3. console.log("yeah me too!")
4. }
5. }

Open main.ts:

import {MustHaveCoffee} from "./getcoffee"

class SweetSweetClass {

constructor() {

console.log("Even sweeter")

}

}

let basil = new SweetSweetClass();

let coffee = new MustHaveCoffee();

Update index.html:

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <title></title>
5. </head>
6. <body>
7. <script src='/dist/bundle.js'></script>
8. </body>
9. </html>

Include [jQuery](http://jquery.com/) with [Typings](https://github.com/typings/typings)

1. Install typings:

npm install typings -g

1. In your project, run:

typings install dt~jquery --global --save

1. Run:

typings install

1. Update tsconfig.json:
2. {
3. "compilerOptions": {
4. "module": "commonjs",
5. "outDir": "dist/",
6. "noImplicitAny": true,
7. "removeComments": true,
8. "preserveConstEnums": true
9. },
10. "include": [
11. "src/\*",
12. "typings/\*"
13. ],
14. "exclude": [
15. "node\_modules",
16. "\*\*/\*.spec.ts"
17. ]
18. }

Import *jQuery* in main.ts

import \* as $ from "jquery";

import {MustHaveCoffee} from "./getcoffee"

class SweetSweetClass {

constructor() {

console.log("Even sweeter");

$('body').css('background-color','blue');

}

}

let basil = new SweetSweetClass();

let coffee = new MustHaveCoffee();

*jQuery* works!

