

Getting to the bottom of: mix tailwind exited with 1 in Phoenix

Written by Seth Corker on Benevolent Bytes

Context

I've been playing with Phoenix lately and it's been a bit of adjustment coming from Next.js, Django and Rails. Most issues I've come up against have been entirely user error, I read some more and figure out the issue was me all along. One issue I ran into recently however, was when adding Tailwind to a new Phoenix project.

What happened?

I went through the instructions to add Tailwind without dealign with `nom` and `node`. But I kept running into the same issue.

```
mix tailwind --runtime-config default
Generated tailwind_test app
node:internal/modules/package_json_reader:4
const internalModuleReadJSON = function (f) { return require('fs').internalModuleReadJSON(f); };
      ^
```

```
TypeError: require(...).internalModuleReadJSON is not a function
    at internalModuleReadJSON (node:internal/modules/package_json_reader:4:68)
    at Object.read (node:internal/modules/package_json_reader:21:42)
    at readPackage (node:internal/modules/cjs/loader:296:36)
    at readPackageScope (node:internal/modules/cjs/loader:329:19)
    at trySelf (node:internal/modules/cjs/loader:444:40)
    at Function.Module._resolveFilename (node:internal/modules/cjs/loader:910:24)
    at Function.Module._load (node:internal/modules/cjs/loader:778:27)
    at Module.require (node:internal/modules/cjs/loader:1005:19)
    at Module._preloadModules (node:internal/modules/cjs/loader:1276:12)
    at loadPreloadModules (node:internal/bootstrap/pre_execution:487:5)
** (Mix) `mix tailwind default` exited with 1
```

Where is the issue?

After a trying different versions of the mix library, then the tailwind-cli directly, I finally found the problem.

An issue on the [vercel/pkg](#) repo reveals the problem, Packaged apps crash with zero messaging when `NODE_OPTIONS` is set

I usually run my projects in the VS Code terminal, what I didn't realise is that the `NODE_OPTIONS` environment variable is set to make debugging easier and this causes issues with the tailwind-cli.

Why does it happen?

The reason this happens is due to the tool, **vercel/pkg**. It's a tool for packaging a node app into a single binary and Tailwind uses this to create their CLI. The benefit to users is that if you don't want to deal with Node and NPM directly, you can use **tailwind-cli** without any dependencies.

How did I work around this?

Run it from another terminal

The VS Code terminal is convenient but with this issue, the easiest workaround is to run the commands in a standalone terminal of your choice.

Unset the environment variable

Running `unset NODE_OPTIONS` in the VS Code integrated terminal before `mix tailwind default` worked for me and lasts for the session.

VS Code settings

I also had success adding the following to the workspace settings - `.vscode/settings.json`

```
{
  "debug.javascript.terminalOptions": {
    "env": null
  }
}
```

This will disable the ability to attach to code in the project automatically but for a Phoenix project where I was avoiding using Node anyway, this isn't a problem.

There you have it. Hopefully this issue will be resolved in future and this workaround won't be needed but as of Tailwind 3.0.24, it's useful to know when running from a VS Code integrated terminal. I used `mix tailwind --runtime-config default` mainly for testing my tailwind setup, if you're using this with phoenix you'll likely be running `mix phx.server`.