

U5AR0LDQA : Change the name
 U601ELFEG : is this a known bug in Elm?
 U0EUHKVGB : Because it correctly figures out what update you are using, but generates the wrong code.
 U0EUHKVGB : Ye.s
 U0EUHKVGB : But please don't do that anyway: it makes your code very hard to read
 U5AR0LDQA : :point_up:
 U601ELFEG : No, sure, I wouldn't do that by default - and didn't notice
 U601ELFEG : and will change it - just I was factoring out that operation (it is used in several other Msg values that I didn't list -) and didn't realize the name conflict - I was stuck thinking it was somehow my polymorphic use of `clearHoveredSlot`
 U601ELFEG : yay - no more hangs - thanks all!
 U17PWHU4D : hossman333: thanks <@U5AR0LDQA> , is there a way to set this inside a let instead? I realize this is still the js mindset, but:
 ...

```
let
  if model.isActive && somethingElse then
    activeVal = "1"
  else
    activeVal = "0.5"
...
```

struggling to see how you would dynamically set a value inside let.

U5AR0LDQA : Yeah so you would do something like this.
 ...

```
let
  isActive = if model.isActive && somethingElse then
    "1"
  else
    "0.5"
in
...
```

U5AR0LDQA : `isActive` becomes the variable
 U5AR0LDQA : So then you can use it in `in` <@U17PWHU4D>
 U17PWHU4D : ooh! nice. thanks <@U5AR0LDQA> trying that out now!
 U5AR0LDQA : You're welcome! :smiley: <@U17PWHU4D>
 U62P4HLCA : I'm banging my head trying to decode a nested json
 {"error":[],"result":{"DASH":{"aclass":"currency","altname":"DASH","decimals":10,"display_decimals":5}}
 U4F64AKQV : <@U62P4HLCA> You probably want to use something like `<http://Json.Decode.at|Json.Decode.at>`
 (<<http://package.elm-lang.org/packages/elm-lang/core/5.1.1/Json-Decode#at>>)
 U4F64AKQV : <<https://noredink.github.io/json-to-elm/>> might also be helpful
 U62P4HLCA : ah thanks... I'll take a look
 U62P4HLCA : perfect thanks
 U5GJ5FEAU : after number of encounters, i think i have figured out json encoding and decoding now :smile:
 U5D4VHEN7 : I can't for the life of me get `Debug.log` to work. It doesn't do absolutely anything. I'm calling it like so (to just make sure it is being called):``

```
main =
  let _ =
    Debug.log "Hello"
  in
  program
    { init = init
    , update = update
    , subscriptions = always Sub.none
    , view = view
    }
...
```

I am using Webpack with `elm-hot-loader` and `elm-webpack-loader` and the options:
 ...

```
{
  verbose: true,
  debug: true,
  warn: true,
}
...
```

Is there some voodoo I need to do to get this to work? Everything else in my application runs absolutely fine.

```
U5D4VHEN7 : also:``
  "elm-hot-loader": "^0.5.4",
  "elm-webpack-loader": "^4.3.0",
...
```

U5D4VHEN7 : Oh goodness, not a Webpack issue at all. Solution was the first response here:
<https://www.reddit.com/r/elm/comments/4pzwrd/using_debuglog/>

```
U5D4VHEN7 : Instead of:``
Debug.log "Whatever, this is just like console.log right?"
...
```

```
Do:
...
Debug.log "Some Label:" "Of course not! This is Elm!"
...
```

Basically, needs two arguments, label and value

U5D4VHEN7 : Confused as to why that doesn't throw a compiler error...

U48AEBJQ3 : Elm has partial application and first-class functions. `Debug.log "Foo"` produces a function of type (a -> a) which you assign to `__` so it gets promptly ignored.

U5D4VHEN7 : Ah, that definitely makes sense. Thanks for the insight <@U48AEBJQ3> :slightly_smiling_face:

U48AEBJQ3 : It could be something that produces a warning, though. You might add an issue on
<<https://github.com/elm-lang/error-message-catalog>>

U5D4VHEN7 : Sweet :slightly_smiling_face: Will do. Given that I spent about an hour smashing my head against that, seems like it could be valuable warnings for beginners

U3LT1UTPF : <@U5XHTBFS6> I'm writing an app with leaflet as the main character. I tried using ports, but after watching richard feldman's talk, I opted for writing a custom element. Not sure if that's convinient for all cases, though.