

U4872964V : problem with haskell is that it's _so much_ going on in there
U2SR9DL7Q : Go... I hear good things. Mainly that it's a dream for concurrent applications.
U4872964V : you don't have to use it all of course, but if you are several people working on the product, then it's a bit of a problem
U4WH8STNX : yeah it does have a lot of overhead
U4WH8STNX : well conceptual overhead that is
U4872964V : maybe not a discussion for <#C192T0Q1E|beginners> though
U4WH8STNX : sure :slightly_smiling_face:, got carried away
U3HQVHERX : <<https://tour.golang.org/welcome/1>>
U4WH8STNX : Fixed: <<https://ellie-app.com/3TrzZNYf4F3a1/1>>
U0F7JPK36 : Oh no this isn't your fault, the error message for this needs to be way better than it currently is
U6FECHN3B : I have a question regarding how data is structured within Elm vs in JavaScript. if I were to pass in an array of Objects from JavaScript to Elm, is a List of Elm records equivalent enough for that to work i.e. passing from JavaScript the following:

...

U6FECHN3B : ````

U6FECHN3B : sorry
U6FECHN3B : ````[{key: key, value: value}, {key: key, value: value}]
...

U65B9414J : I'm using Json.Decode.Pipeline . `

```
type alias User =  
  { name : String  
    , age : Int  
    , githubid : Maybe String  
  }
```

```
userDecoder : Decoder User  
userDecoder =  
  decode User  
    |> required "name" string  
    |> required "age" int  
    --|> optional "githubid" (nullable string) (Just "")  
    |> optional "githubid" (nullable string) Nothing  
`
```

U65B9414J : commented out line works
U6FECHN3B : would that be represented in Elm as
...

```
type alias itemname =  
{ key: String, value: String  
}  
...
```

U65B9414J : replacing Just "" with Nothing fails. Anyone knows why.
U6FECHN3B : something according to that
U6FECHN3B : I'm looking to pass in an array of key value pairs via a port
U48AEBJQ3 : <@U65B9414J> It compiles for me. What error are you getting?