U0LPMPL2U : It sounds like you're saying `Http.send identity myRequest` right now?
U0LPMPL2U : You probably want `Http.send SomeMsg myRequest`
U2XRG0UKA : You can handle this with `Task`s
U2XRG0UKA : Use `Http.toTask` on your `Request`
U2XRG0UKA : Then `Task.map2` and finally `Task.attempt`
U40QW928G : ugh I don't think I understand the purpose of these generic types in Elm
U60SXAF96 : <@U2XRG0UKA> -- Perfect. Thanks.
U40QW928G : when I see stuff like `Cmd msg` it feels like I can pass w/e I want buthat seems to be false
U0LPMPL2U : <@U40QW928G> you can pass it whatever you want but you'll be restricted to what the implementation of the function can be
U3SJEDR96 : You can, as long as you give the same type for `msg` in the whole signature
U0LPMPL2U : For example:```
double : List a -&gt; List b
double list =
  List.map (\n -&gt; n * 2) list
```
is not valid because there's on guarantee that the input list is a list of numbers

U0LPMPL2U : as <@U3SJEDR96> mentioned, the `msg` must mean the _same_ thing everywhere in your signature.
For example:The List.member function has the signature: `member : a -&gt; List a -&gt; Bool`

```
List.member "a" [1,2,3]
```

is _not allowed_ because the signature says the first argument `a` must be of the _same type_ as the values in the list `List a`.

U0LPMPL2U : `a` in the signature means "whatever" but all the `a`s must be the _same_ "whatever" if that makes any sense :slightly_smiling_face:
U0T2S9G6S : Any recommended book to learn Elm?
U57KYFW67 : The only book for Elm isn't released quite yet
U0T2S9G6S : How is Elm in Action?
U57KYFW67 : That's the one I'm talking about
U57KYFW67 : "MEAP began July 2016 Publication in Early 2018 (estimated)"
U57KYFW67 : It will probably be great. Feldman is very articulate and he seems very sensitive to the way people learn.
U3SJEDR96 : You can already dive into the chapters that have been written, tho, and totally should!
U23SA861Y : any reason why Random does not have a `toTask`?
```