U6C09T8HX : ...with the same name?
U0JFGGZS6 : <@U3F8KG1DE> you may find this helpful for getting to the next sibling and its scrollHeight: <http://package.elm-lang.org/packages/debois/elm-dom/1.2.3/DOM>
U0JFGGZS6 : without resorting to ports
U48AEBJQ3 : <@U6C09T8HX> It is an alias for a type defined in another module. In this case, a module which is not exported from the package.
U6C09T8HX : hmmm
U6C09T8HX : Forgive my ignorance gilbert, but are you saying that the http package contains a module somewhere within it which defines the Request type but since it is not exported by that module we see this: "type alias Request a = Request a"? So all I need to take away from that like is that something can be tagged as a request?
U48AEBJQ3 : It seems that there are two issues in your question. First is that there is no special way to reexport types and functions from a module and sometimes you end up having something in a module you don't want to export. Using this type alias is effectively exporting just this type from the other module.
Also, it is very common to *not* export the constructors of a data type. By leaving them hidden, the maintainer can change implementation details without changing the exposed API. It also helps to prevent people from creating bad states by forcing them to use controlled functions to get an instance of a type.

U3LUC6SNS : Hello, I have two functions that I would like to run in sequence in the `update` function:
First is
```
login2 : Model -&gt; (Model, Cmd Msg)
login2 model =
  ( login model,  Request.User.loginUserCmd model Request.Api.loginUrl)
```
and second is
```
search : SearchDomain -&gt; String -&gt; Page -&gt; Model -&gt; ( Model, Cmd Msg )
search searchDomain query page model = ...
```
How would I go about doing this?

U6C09T8HX : oThank you very much for taking the time to answer my question Gilbert! It's very much appreciated. How can I repay you?
U6C09T8HX : ...and what motiates you to hang out here and answer questions?
U153UK3FA : <@U3LUC6SNS> ```let
  (model1, cmds1) = login2 model
  (model2, cmds2) = search searchdomain something somepage model1
in
  (model2, Cmd.batch [cmds1, cmds2])
```

U153UK3FA : <http://package.elm-lang.org/packages/Fresheyeball/elm-return/latest> is a package that can remove some of the boilerplate
U3LUC6SNS : Thankyou <@U153UK3FA>!!
U153UK3FA : <@U3LUC6SNS> I wasn't sure if you wanted to sequence the Cmds (which you can't do)
U3LUC6SNS : <@U153UK3FA>, that works fine - actually, it works beautifully!
U153UK3FA : just to be clear, there is no guarantee that your login will complete before your search request.
U153UK3FA : if they depend that ordering (implicitly passing state through the model) then they'll likely fail under different network conditions
U48AEBJQ3 : <@U6C09T8HX> Just try to be a decent person who makes the world a bit better. I don't have a rationalization for why I do this.
U6CEJALSF : hey folks! new to elm and taking the knowthen course on elm beginners concepts.  I'm a JavaScript developer by day, which is a pretty natural progression into elm, I think.(link to _free_ course: <http://courses.knowthen.com/>)

U6CEJALSF : I hope to get to know Elm a lot better over the coming months, partly for self-enrichment, partly in order to convince my team that Elm is a big step in the right direction for web UIs.
U0JBSEGHY : I've got this message. ```type Msg    = Answer String
    | MakeCall String
    | IncomingCall Maybe String```
```

U0JBSEGHY : My update is giving me this error ```Pattern Main.IncomingCall has too few arguments.
220| IncomingCall caller -&gt;
            ^^^^^^^^^^^^^^^^^^^
Expecting 2, but got 1.```

U153UK3FA : <@U0JBSEGHY> `IncomingCall (Maybe String)`
U0JBSEGHY : I'm calling it from a subscription. ```subscriptions model =   Sub.batch
        [incomingCall IncomingCall ....```

U0JBSEGHY : Oh
U0JBSEGHY : ```The argument to function `incomingCall` is causing a mismatch.
276| incomingCall IncomingCall
                           ^^^^^^^^^^^^
Function `incomingCall` is expecting the argument to be:

    String -&gt; msg

But it is:

    Maybe String -&gt; Msg
```

U0JBSEGHY : I just needed to add Maybe String to my port it seems
U0JBSEGHY : slowly getting the hang of this Types thing :slightly_smiling_face:
U3F8KG1DE : I thought so too but this way feels like a workaround. I mean perhaps there is another way to acheive the same goal i.e. to animate some div element?
U3F8KG1DE : <@U0JFGGZS6> I thought so too but this way feels like a workaround. I mean perhaps there is another way to acheive the same goal i.e. to animate some div element?
```