U52GHJJTU : Is it possible to move the common parts of two elm apps to a separate js file during compilation?
U52GHJJTU : <@U3SJEDR96> I guess that's what you're proposing
U3SJEDR96 : not possible, but _maaaaybe_ in 0.19, or at least a first step towards it. But yeah, no, not quite what I'm proposing.
U3SJEDR96 : I'm just proposing to work around the naming conflict in a predictable, future-proof way
U52GHJJTU : Ok, that's what I thought
U2NGTHTTR : Summary:There is page about we _generally_ do not know nothing.
Page can contains _generally_ anything incl. legacy versions of our Elm applications.

Now we want to _inject_ into this page another Elm application and avoid merging through common global `Elm` variable because we don't have any control above content of page.

U3SJEDR96 : Well, how do you _know_ any other variable isn't already used?
U3SJEDR96 : - use predictable, versioned name for your app- check if `Elm.MyModule_0_1_1` exists. If it does, it's your app, no question about it
- if it doesn't, fetch your elm app and let it merge

U2NGTHTTR : > how do you _know_ any other variable isn't already used?I don't know of course I still can choose name which is already used but for this glob. variable I can use something like `Elm_5325173517` - this name is used in one line. Just to call `.embed()`. It's easy to change and without any impact on rest of application.

Thank you for your help and ideas

U2D5SAEMN : I have two date ranges (start date, end date) that depend on each other, with a number of constraints:1. range2 can only exist if range1 exists
2. range2 start date is optional
3. range2 end date is optional
4. range2 start date >= range1 start date
5. range2 start date <= range1 end date
6. range2 end date >= range1 start date
7. range2 end date <= range 1 end date
8. range2 end date > range2 start date

So basically range1 is the boundary for range2. I'm figuring out what the best data structure would be:
- union type: `NotSet | BoundedDateRange (Date, Date) (Maybe Date, Maybe Date)`
- record: `{ range1: (Date, Date), range2: (Maybe Date, Maybe Date)}`

Any ideas/remarks?

U3LUC6SNS : Hi all! I am running into performance problems and am thinking of using one of Elm's debouncers. Does anyone have experience on these or advice to offer?
At the moment, each character of user input results in an update to the current document and also the document list. In addition, there is processing in JS-land of the content via ports.

Nothing goes "wrong" but both the source and rendered text windows get very "laggy". Also, I see lots of messages

```
149:1 GET <http://localhost:3000/149> 404 (Not found)
150:1 GET <http://localhost:3000/150> 404 (Not found)
```

U3LUC6SNS : ```
U3LUC6SNS : in the console. I am using webpack and think they must come from it.
U3LUC6SNS : However, I don't understand what they mean.
U3LUC6SNS : in the JS console
U3SJEDR96 : hm, is the code available somewhere? <@U3LUC6SNS>
U3LUC6SNS : <@U3SJEDR96>, here it is: <https://github.com/jxxcarlson/koko_elm_client> Should I point you to a specific place? Keyboard input is processed in `Main` at <https://github.com/jxxcarlson/koko_elm_client/blob/master/src/Main.elm#L233>
U3SJEDR96 : cool, I'll have a look shortly
U3SJEDR96 : <@U3LUC6SNS> I'm not seeing where the random GET's come from... As for the puts, you may indeed

want to batch these (either with a debouncer of a little more manually by keeping a dirty flag and checking that every second or so), _or_, if you're not really all that interested in whether the PUT succeeds or not, you could make these run in an isolated process which should help with "snappiness"
U3SJEDR96 : (by using `Http.toTask`, `Process.spawn` and `Task.perform`)
U3SJEDR96 : then finally, the rendering - I'm not entirely sure if that currently blocks the main thread (i.e. does an outbound port run its callback synchronously?), but I think I can check that quite quickly