U1G51S63S : And now I am able to use this string via `innerHtml` but thought maybe it's possible to get more :slightly_smiling_face:
U3SJEDR96 : Yeah, that's either innerHtml or a custom virtual-dom node (which involves some native to hook things up)
U4872964V : <@U1G51S63S> is it an alternative to use an `&lt;img src="data:image/svg+xml;utf8,&lt;svg&gt; ... &lt;/svg&gt;"/&gt;`?
U1G51S63S : <@U4872964V> I want to keep them styleable. looking at `object` now and svg defs as well
U0EUHKVGB : <@U1G51S63S> Why are you trying to do this?
U1G51S63S : <@U0EUHKVGB> to not hardcode svg paths into code and use webpack loader instead. since I will need to style SVG's and inlining is the most customizable way to style it later
U0EUHKVGB : <@U1G51S63S> Pass them in via flags or load them via json.
U1G51S63S : I have no problems with receiving svg code at all (just using elm-assets-loader)
U0EUHKVGB : I know you are trying to avoid boilerplate. But you are making your code more fragile and prone to breaking randomly by doing this.
U0EUHKVGB : Use flags, or json to receive the paths. Generate the svg views from that at runtime.
U1G51S63S : so I have `Icons.elm` with all my icons now ```portfolioManagement : Arg msgportfolioManagement =
    icon257 "portfolio-management"
        """&lt;path fill="#D1D3D4" d="M128.5 115.862l-90.205 27.51v37.177l90.205 27.508 90.205-27.508v-37.177z"/&gt;&lt;path fill="#808285" d="M128.5 48.943L38.295 76.455v37.176l90.205 27.51 90.205-27.51V76.455z"/&gt;&lt;path opacity=".5" fill="#D1D3D4" d="M87.054 128.503l41.446 12.638 41.445-12.638-41.445-12.641z"/&gt;"""```

U1G51S63S : but it's about 2000 lines already and I should add 300 more icons now :smile:
U1G51S63S : icon257/icon are ```icon : String -&gt; String -&gt; String -&gt; String -&gt; Svg msgicon viewbox name body className =
    let
        attributes =
            [Svg.Attributes.viewBox viewbox
            , Svg.Attributes.class ("icon icon--" ++ name ++ " " ++ className)
            , innerHtml body
            ]
    in
        Svg.svg attributes []

icon257 : String -&gt; String -&gt; String -&gt; Svg msg
icon257 =
    icon "0 0 257 257"
```

(className is a string because `Svg.Attributes.class` was not special case like in `Html` and not allowed to use twice for example. looks like fixed already, need to rewrite)

U1G51S63S : but this code requires a lot of work - need to `svgo` icons, extract paths (and avoid root `svg` tag), paste it here. I think it will be easier when webpack will do it
U1G51S63S : and it will be a hell when icons will change
U1G51S63S : I am really just considering other ways now - `object`, svg sprites, etc.
U1G51S63S : looks like `object` allows some customizations too
U6D1E91QV : is there special magic involved in the `Json.Decode` functions or can they be implemented in pure elm?
U0LPMPL2U : You can implement json decoders in pure elm
U0LPMPL2U : under the hood, `Json.Decode` may have some fancy stuff but you never need to deal with that
U6D1E91QV : i'm specifically looking for that fancy stuff though
U6D1E91QV : i am trying to construct a record piece by piece from results
```