U5JEJN1CP : With the macro `(keyed [a b  c d])` becomes `{:a a :b b :c c :d d}`
U644Z8N9H : I see
U04V1HS2L : <@U3L6TFEJF> <@U0NCTKEV8> I'm the author, but frankly speaking, haven't used it for long :slightly_smiling_face: can we solve this?
U04V70XH6 : <@U04V1HS2L> looks like you'd need exclusions in there to avoid certain `sym` values (any primitive Java type symbols)
U65F82796 : I am new to clojure-api. I develop small app using clojure-api. I have done DB connection, Logs capturing mechanism in my APP. I need help in exception handling mechanism. Is there any library to catch exception all types of exception and to send exception to developer mail?
U04V70XH6 : <@U65F82796> A Google/Bing search reveals lots of Clojure wrappers for SMTP mail but, to be honest, I'd just use javax.mail directly. It's not hard.
U65F82796 : <@U04V70XH6> Okay. Thanks for quick reply.
U04V70XH6 : Sending email to developers every time you get an exception is... less than best practice... You could cause a denial of service attack on your own email provider if you app goes wrong. I definitely would not recommend that approach.
U0524B4UW : <@U65F82796> we use <http://sentry.io|sentry.io> for such exception capturing purposes... and there are many similar services out there
U65F82796 : <@U0524B4UW>, <@U04V70XH6> Okay. Thanks for valuable suggestions.
U60259LCB : How would I wrap this in a swap! ? ```(map #(make-boolean! %) @items)```

U2VNB2S1Z : `(swap! items #(map make-boolean! %))`
U2VNB2S1Z : that's one way to do it
U0ALP2929 : Is there a way to tell `lein repl` to not go check the dependencies? (if I have no internet or the artefact server is down?)
U64FCR0E5 : `-o` for offlinemode
U0ALP2929 : `export LEIN_OFFLINE=true &amp;&amp; lein repl` seems to work.Tried `lein -o repl` and `lein repl -o` but they still seem to go look for dependencies

U06F82LES : <@U0ALP2929>, do you have SNAPSHOT dependencies in your project.clj?
U0ALP2929 : Had one, thanks :slightly_smiling_face:
U06F82LES : <@U0ALP2929>, you can pin SNAPSHOT deps: <https://www.martinklepsch.org/posts/maven-snapshots.html>
U0ALP2929 : Yes, just did that :slightly_smiling_face:
U06F82LES : :dart:
U06F82LES : <@U51QL32TT> looks great as a series of forms typed into the repl. For me the next step would be to copy it to a file and refactor into functions. Idiomatic clojure meant for .clj files (i.e. not typed ad hoc into a repl) eschews dynamic `def`s
U06F82LES : in this case, it could look like this: ```(let [cc (read-line), startswith (first cc), credit-card (case ...)] (println "credit card:" credit-card))```

U06F82LES : (of course you shouldn't type it all in a single line, I was too lazy to use an editor)
U51QL32TT : <@U06F82LES> thanks for that. I'm stuck on a my chromebook using an online repl So I think I'll have to try it on an editor later. I appreciate the insight
U06F82LES : ah nice, which online repl do you use?
U51QL32TT : <http://repl.it|repl.it>
U51QL32TT : It has its limitations but is sufficient enough to play a bit with clojure
U06F82LES : nice!
U06F82LES : it's quite fast
U51QL32TT : It's the best I've found
U51QL32TT : It's been a bit rough learning clojure. Probably because it's new enough that there aren't as many resources available compared to other languages