

U5ZAJ15P0 : The error is `Can't let qualified name`, but I don't get why the gensym'ed variable gets qualified by the syntax quote

U14DSHZ4P : dumb! i was using the wrong function here. Thanks <@U0NCTKEV8>

U0NCTKEV8 : I would double check to make sure the code that is being run actually matches the code you are reading (restart your jvms) and then I would suspect this is actually coming from some other macro

U5ZAJ15P0 : I restarted lein's repl. Is that enough?

U0NCTKEV8 : yes

U0NCTKEV8 : I would put a println at the start of the macro to print out ctx-binding because I suspect it is already being passed in a fully qualified symbol

U0NCTKEV8 : your docstring is also in the wrong place

U5ZAJ15P0 : ah, thanks for pointing that out

U0NCTKEV8 : it is a super common thing to do

U5ZAJ15P0 : ctx-binding seems fine. The println prints `ctx`

U5ZAJ15P0 : which matches how I am using the macro, e.g.``

(deftest test-authentication

(with-scratch-ctx ctx

... ..

U5ZAJ15P0 : Still throwing `:cause "Can't let qualified name: wef-backend.test-util/conn# ""` though

U0NCTKEV8 : is d/db a macro?

U5ZAJ15P0 : no

U5ZAJ15P0 : but I get the same error even if I deleted the whole ~ctx-binding line

U5ZAJ15P0 : I never got that error with uri# or datomic# by the way, it only occurred when I added conn#

U5ZAJ15P0 : It's probably something silly

U0NCTKEV8 : do you have a var named conn# in that namespace?

U051SA920 : <@U5ZAJ15P0> What does macroexpand give you?

U5ZAJ15P0 : <@U0NCTKEV8> no, I checked. Let me check again just to make sure..

U5ZAJ15P0 : `java.lang.RuntimeException: No such var: wef-backend.test-util/conn#`

U0NCTKEV8 : you could fix it immediately by replacing most of the macro with a function that returns the context, and have the macro expand in to invoking that and then deleting it

U0NCTKEV8 : but that wouldn't tell you what is wrong

U5ZAJ15P0 : <@U051SA920>``

(let*

[uri__16604__auto__

(clojure.core/str "datomic:mem://hello-test-" (java.util.UUID/randomUUID))

datomic__16605__auto__

(integrant.core/init-key :wef-backend/datomic {:uri uri__16604__auto__})

wef-backend.test-util/conn#

(:conn datomic__16605__auto__)

ctx

{:auth nil :conn conn__16606__auto__ :db (datomic.api/db conn__16606__auto__)}

(try (+ 1 2) (finally (datomic.api/delete-database uri__16604__auto__))))

... ..

U5ZAJ15P0 : when running``

(macroexpand '(wef-backend.test-util/with-scratch-ctx ctx (+ 1 2)))

... ..

U051SA920 : What happens when you rename it to `some-other-conn#`?

U5ZAJ15P0 : tried that, didn't work. Trying again now just to make sure

U0NCTKEV8 : is that a real pound sign, or some unicode nonsense?

U051SA920 : Well the second usage of `conn#` expands properly. (in the map)

U0NCTKEV8 : yeah, which makes me suspect the first has some weird shenanigans

U5ZAJ15P0 : Wait, I think I found the issue

U5ZAJ15P0 : `some-other-conn# (:conn datomic#)]`

U5ZAJ15P0 : the "space" between # and (has charcode 160