tVar

# Init

require("../tVar.lua")

matrix = require("../matrix")

# Global tVar.

numFormat = "%.3f"

mathEnviroment = "align"

debugMode = "off"

outputMode = "RES" --RES, RES\_EQ, RES\_EQ\_N

numeration = true

# New

tVar:New(0.04,"r\_{se}")

tVec:New({10,2,7},"v\_{1}")

tMat:New({{10,2,5},{2,4,3},{7,4,3}},"a\_{2}")

# Output

:print() -–abh. v OutputMode

:outFull(number[bool],enviroment[bool])

:outHalf([bool],[bool])

:outVar([bool],[bool])

:out() –-nur Wert

# Set [tVar]

:setName([string])

:setUnit([string])

:clean(name[string]) –-berechn. Schritte entf.

Tipp: können verkettet werden

# Misc

[tVar]:bracR() –-Runde Klammern

[tVar]:CRLF([string]) –-neuwline, [string] wird vor und nach Umbruch eingefügt

# Math

tVar.sqrt([tVar],[number])

tVar.PI

[tMat]:T() -–Transponieren

[tMat]:Det()

[tMat]:Inv()

[tVec]:crossP()

# Example

\begin{luacode\*}

require("../tVar.lua")

matrix = require("../matrix")

numFormat = "%.2f"

outputMode = "RES\_EQ\_N”

numeration = false

sigma\_x = tVar:New(11.4,"\\sigma\_{x}")

tau\_xy = tVar:New(2.43,"\\tau\_{xy}")

sigma\_v = tVar.sqrt(sigma\_x^2+3\*tau\_xy^2,2)

sigma\_v:setName("\\sigma\_{v}")

sigma\_v:setUnit("N/m^2")

sigma\_v:print()

\end{luacode\*}