TexVar - CheatSheet
October 22, 2015 - Sebastian Pech

Init

```
require("tVar/init.lua")
```

Input

```
tVar[[
--Define Var and print
a:=10:outRES()
--Do calc and print
b:=(a+10):print()
--Matrices
A:=({{10,2},{2,4},{7,3}}):outRES()
--Output LaTeX
# Variable %%b%% has the value $$b$$
--Functions
f(x,\\sigma):=x^2+2^\\sigma
--Print function equation
x:=nil
\\sigma:=nil
f(x,sigma):outEQ()
]]
tVar.intFile([string])
```

Global

```
tVar.numFormat = "%.6f"
tVar.mathEnviroment = "align"
tVar.debugMode = "off"
tVar.outputMode = "RES_EQ" --RES,
   RES_EQ, RES_EQ_N
tVar.numeration = true
tVar.decimalSeparator = "."
tVar.calcPrecision = 10
tVar.disableOutput = false
tVar.autocutZero = true
tVar.coloredOuput = false
tVar.log = false
tMat.texStyle = "mathbf"
tMat.eqTexAsMatrix = false
tVec.texStyle = "vec"

tPlot.steps = 0.01
```

New

```
tVar:New(0.04,"r_{se}")
tVec:New({10,2,7},"v_{1}")
tMat:New({{10,2},{2,4},{7,4}},"a_{2}")
```

Output

```
[tVar]:print()
[tVar]:outRES_EQ_N(number[bool],
   enviroment[bool])
[tVar]:outRES_EQ([bool],[bool])
[tVar]:outRES([bool],[bool])
[tVar]:outEQ([bool],[bool])
[tVar]:out()
```

Set

```
[tVar]:setName([string])
[tVar]:setUnit([string])
[tVar]:clean(name[string])
```

Misc

```
[tVar]:bracR()
[tVar]:CRLF([string])
[tVar]:CRLFb([string])
[tVar]:copy()
tex.print([string])
```

Math

```
tVar.sqrt([tVar],[number])
tVar.PI
[tMat]:T()
[tMat]:Det()
[tMat]:Inv()
[tVec]:crossP()
```

Converted math functions:

```
[tVar].abs,[tVar].acos, [tVar].cos,
[tVar].cosh, [tVar].asin, [tVar].sin,
[tVar].sinh, [tVar].atan, [tVar].tan,
[tVar].tanh, [tVar].ceil, [tVar].floor,
[tVar].exp, [tVar].log, [tVar].log10,
[tVar].rad, [tVar].deg, [tVar].atan2
```

Plot

```
tPlot:New([tPlot]present)
[tPlot].xlabel = "{/Symbol e}_c"
[tPlot].ylabel = "{/Symbol s}_c"
[tPlot].steps = 0.0001
-- define x axis and range
[tPlot].xtics = "0.001"
[tPlot].xrange = "[0:0.002]"
[tPlot].yrange = "[0:16]"

plotc.conf.size = "4cm,4cm"

[tPlot].add((fun|{{X1,Y1},{X2,Y2}}),
   title,style)
[tPlot].plot()
```

Basic Document

```
\documentclass{article}
\usepackage{luacode}
\usepackage[fleqn]{amsmath}
\usepackage{graphicx}
\usepackage[left=2.5cm,right=2.5cm,top
   =2.5cm,bottom=2cm,includeheadfoot]{
   geometry}
\begin{document}
\begin{luacode}
require("tVar/init.lua")
tVar[[
-- INSERT TVAR CODE
]]
\end{luacode}
\end{document}
```