ipynb Group

John D. Baker

https://github.com/bakerjd99/jacks/blob/master/ipynbfrjod/ipynb.ijs

SHA-256: fc66e29e8ee45720981412540b9dc901469f38195b2a23ce7a69de81c81d78a0

February 25, 2024

Contents

ipynb Overview **2**

ipynb Interface 2

Installing ipynb 2

Using ipynb 3

ipynb Source Code **4**

=: Index **11**

ipynb Overview

ipynb is a J script that extracts J words from JOD dictionaries and inserts them in blank jupyter notebooks.

ipynb Interface

```
ipynbfrjod [8] extract J words from JOD and insert in blank jupyter notebook
```

Installing ipynb

If you have a current vesion of J (9.0x+ or later) installed ipynb can be downloaded as a J addon script by typing the following commands into a JQt or JHS session.

```
NB. install addon files in ~addons/jacks
install 'github:bakerjd99/jackshacks'

NB. installed files
dir '~addons/jacks'

NB. load script
load '~addons/jacks/ipynb.ijs'
```

To get the lastest vesions of ipynb and other addon scripts in addons/jacks simply reinstall.

Using ipynb

To use `ipynb` do the following:

1. Install the JOD addon. `ipynb` uses JOD. J addons are easily installed with pacman.
2. Create or load JOD dictionaries containing code you are interested in. `ipynb` is a group in the `docs` directory. Many JOD dictionaries are available as JOD dump scripts on GitHub here.
3. Start J and do:

```
require 'general/jod'

NB. open dictionaries containing example code
od ;:'docs utils'

NB. load (ipynb)
load '~addons/jacks/ipynb.ijs'

NB. notebook from (ipynb) code
nbj=: ipynbfrjod }. grp 'ipynb'
nbj write 'c:\your\jupyter_notebooks\ipynb_onself.ipynb'
```

Examples of `ipynbfrjod` outputs are available here `ipynb_onself.ipynb` and here `ipynb_onself.pdf`.

ipynb Source Code

```
NB.*ipynb s-- insert j code in jupyter notebooks.
NB.
NB. verbatim: interface word(s):
NB. ---------------------------------------------------------------------------
NB.  ipynbfrjod - extract J words from JOD and insert in blank jupyter notebook
NB.
NB. created: 2022jul23
NB. ---------------------------------------------------------------------------

coclass 'ipynb'

NB.*dependents
NB. (*)=: PYESCAPECHRS REVPYESCAPECHRS NBHEADER NBTRAILER NBJCELLBEGst NBJCELLBEGen NBJCELLEND
NB.*enddependents

NB. common python string escape characters  - order matters
PYESCAPECHRS=:  ;(254{a.) ,&.> <;._1 ' \'' '' \\ \ \" "'

NB. reverse python escapes - excluding single quote - order matters
REVPYESCAPECHRS=:  ;(254{a.) ,&.> }. |."1 ] _2 ]\ <;._1 PYESCAPECHRS

NB. blank notebook json cell templates
NBHEADER=:  (0 : 0)
{
 "cells": [
)

NBTRAILER=:  (0 : 0)
],
 "metadata": {
  "kernelspec": {
   "display_name": "J",
   "language": "J",
   "name": "jkernel"
  },
  "language_info": {
   "file_extension": ".ijs",
   "mimetype": "text/J",
   "name": "J"
  }
 },
 "nbformat": 4,
 "nbformat_minor": 5
}
)

NBJCELLBEGst=:  (0 : 0)
  {
   "cell_type": "markdown",
   "metadata": {},
   "source": [
)
```

```
NBJCELLBEGen=:  (0 : 0)
   ]
  },
  {
   "cell_type": "code",
   "execution_count": null,
   "metadata": {},
   "outputs": [],
   "source": [
)

NBJCELLEND=:  (0 : 0)
   ]
  },
)


NB.*end-header

NB. interface words (IFACEWORDSipynb) group
IFACEWORDSipynb=: ,<'ipynbfrjod'

NB. prefix for markdown sections - converts to easily found strings in notebooks
JWORDMARK=: ' :::jword::: '

NB. line feed character
LF=: 10{a.
```

```
NB. markdown cell section marker
MDSECTION=: '###'

NB. root words (ROOTWORDSipynb) group
ROOTWORDSipynb=: <;._1 ' IFACEWORDSipynb PYESCAPECHRS ROOTWORDSipynb VMDipynb ipynbfrjod'

NB. version, make count and date
VMDipynb=: '0.8.0';16;'25 Feb 2024 12:21:03'

NB. retains string (y) before last occurrence of (x)
beforelaststr=: ] {.~ 1&(i:~)@([ E. ])

changestr=: 4 : 0

NB.*changestr v-- replaces substrings - see long documentation.
NB.
NB. dyad:  clReps changestr cl
NB.
NB.   NB. first character delimits replacements
NB.   '/change/becomes/me/ehh' changestr 'blah blah ...'

pairs=. 2 {."(1) _2 [\ <;._1 x       NB. change table
cnt=._1 [ lim=. # pairs
while. lim > cnt=.>:cnt do.          NB. process each change pair
  't c'=. cnt { pairs                NB. /target/change
  if. +./b=. t E. y do.              NB. next if no target
    r=. I. b                         NB. target starts
    'l q'=. #&> cnt { pairs         NB. lengths
    p=. r + 0,+/\(<:# r)$ d=. q - l NB. change starts
    s=. * d                         NB. reduce < and > to =
    if. s = _1 do.
      b=. 1 #~ # b
      b=. ((l * # r)$ 1 0 #~ q,l-q) (,r +/ i. l)} b
      y=. b # y
      if. q = 0 do. continue. end.  NB. next for deletions
    elseif. s = 1 do.
      y=. y #~ >: d r} b            NB. first target char replicated
    end.
    y=.(c $~ q *# r) (,p +/i. q)} y  NB. insert replacements
  end.
end. y                              NB. altered string
)

NB. enclose all character lists in blcl in " quotes
dblquote=: '"'&,@:(,&'"')&.>

ipynbfrjod=: 3 : 0

NB.*ipynbfrjod v-- extract J  words from JOD  and insert in blank
NB. jupyter notebook.
NB.
NB. monad:  clIpynb =. ipynbfrjod blclNames
NB.
NB.   NB. examples use docs and utils
NB.   require 'general/jod'
NB.   od ;:'docs utils'
NB.
NB.   nbj=: ipynbfrjod ;:'sha1 sha1dir'
NB.   nbj write 'C:\Users\baker\jupyter_notebooks\test0.ipynb'
NB.
NB.   nbj=: ipynbfrjod }. grp 'ipynb'
NB.   nbj write 'C:\Users\baker\jupyter_notebooks\ipynb_onself.ipynb'
NB.
NB.   NB. macOS/linux
NB.   nbj write jpath '~/jupyter_notebooks/test0.ipynb'

NB. require 'general/jod' !(*)=. disp
jc=. disp&.> y

NB. markdown sections with word name
sec=. dblquote (<MDSECTION,JWORDMARK) ,&.> y
sec=. (<NBJCELLBEGst) ,&.> sec ,&.> <NBJCELLBEGen

NB. j code to quoted list of python strings notebook format
nbj=. <;._2@(REVPYESCAPECHRS&changestr)@tlf&.> jc
nbj=. ;&.> '"' , L: 0 (<'\n",',LF) ,~ L: 0 nbj
nbj=. ,&'"'&.> '\n",'&beforelaststr&.> nbj
nbj=. sec ,&.> nbj ,&.> <NBJCELLEND
toJ NBHEADER , (LF ,~ ','&beforelaststr ;nbj) , NBTRAILER
)

NB. appends trailing line feed character if necessary
```

```
tlf=: ] , ((10{a.)"_ = {:) }. (10{a.)"_

NB. converts character strings to J delimiter LF
toJ=: ((10{a.) I.@(e.&(13{a.))@]}  ])@:(#~ -.@((13 10{a.)&E.@,))

NB. writes a list of bytes to file
write=: 1!:2 ]`<@.(32&>@(3!:0))

NB.POST_ipynb ipynb post processor

(".;(0=nc <'SHOWSMO_ijod_'){'1';'SHOWSMO_ijod_') smoutput IFACE=:  (0 : 0)
NB. (ipynb) interface word(s): 20240225j122103
NB. --------------------------
NB. ipynbfrjod  NB. extract J words from JOD and insert in blank jupyter notebook
)

cocurrent 'base'
coinsert  'ipynb'
```

Index

beforelaststr, 7

changestr, 7

dblquote, 8

IFACE, 10
IFACEWORDSipynb, 6
ipynbfrjod, 8

JWORDMARK, 6

LF, 6

MDSECTION, 7

NBHEADER, 4
NBJCELLBEGen, 6
NBJCELLBEGst, 5
NBJCELLEND, 6
NBTRAILER, 5

PYESCAPECHRS, 4

REVPYESCAPECHRS, 4
ROOTWORDSipynb, 7

tlf, 10
toJ, 10

VMDipynb, 7

write, 10