

Org-babel の使用例

osada-yum

2021 年 12 月 21 日

目次

1	Org-babel の設定	1
1.1	言語の設定	1
1.2	言語の編集のときの major-mode の設定	2
1.3	Fortran compiler の設定	2
2	Fortran code block	2
2.1	Fortran コードブロック	2
2.2	compiler flags	3
2.3	Fortran to R	3
2.4	adjustl, adjustr, trim	5

1 Org-babel の設定

1.1 言語の設定

C-c C-c で実行. `org-babel-do-load-languages` を使う.

```
1 (org-babel-do-load-languages
2   'org-babel-load-languages
3   '((emacs-lisp . t)
4     (fortran    . t)
5     (R          . t)))
6 (print org-babel-load-languages)
```

```
((emacs-lisp . t) (fortran . t) (R . t))
```

1.2 言語の編集のときの major-mode の設定

org-src-lang-modes alist を追加する.

```
1 (add-to-list 'org-src-lang-modes '("fortran" . f90))
2 (print org-src-lang-modes)
```

```
(("fortran" . f90) ("redis" . redis) ("php" . php) ("arduino" . arduino) ("C" . c)
lisp) ("ocaml" . tuareg) ("screen" . shell-script) ("shell" . sh) ("sqlite" . sql))
```

1.3 Fortran compiler の設定

org-babel-fortran-compiler を変更する.

```
1 ;; (setq org-babel-fortran-compiler "ifort")
2 ;; (customize-set-value 'org-babel-fortran-compiler "ifort")
3 (print org-babel-fortran-compiler)
```

```
"gfortran"
```

2 Fortran code block

2.1 Fortran コードブロック

C-c C-c で実行.

```

1 use, intrinsic :: iso_fortran_env
2 print'(es30.15)', acos(-1.0_real32) , acos(-1.0_real64)
3 print'(es30.15)', 4*acos(-1.0_real32), 4*acos(-1.0_real64)

```

```

3.141592741012573E+00
3.141592653589793E+00
1.256637096405029E+01
1.256637061435917E+01

```

2.2 compiler flags

スペースで区切る flag1 flag2 かリスト '("flag1" "flag2") で渡す.

```

1 print*, __FILE__, __LINE__
2 print*, F00, BAR

```

```

/tmp/babel-q9yRPY/fortran-src-xSuWnh.F90          4
                                                    3  1062752.88

```

ただ, リストで渡すと :exports code ができない?

```

/tmp/babel-q9yRPY/fortran-src-OZ4pPb.F90          4
                                                    3  5.54448402e+29

```

2.3 Fortran to R

表の出力の長さを変える. org-table-convert-region-max-lines を変更. 表の出力が長すぎると, org-table-convert-region が遅くなって Emacs がハングするかも.

```

1 (customize-set-variable 'org-table-convert-region-max-lines 1000)
2 (print org-table-convert-region-max-lines)

```

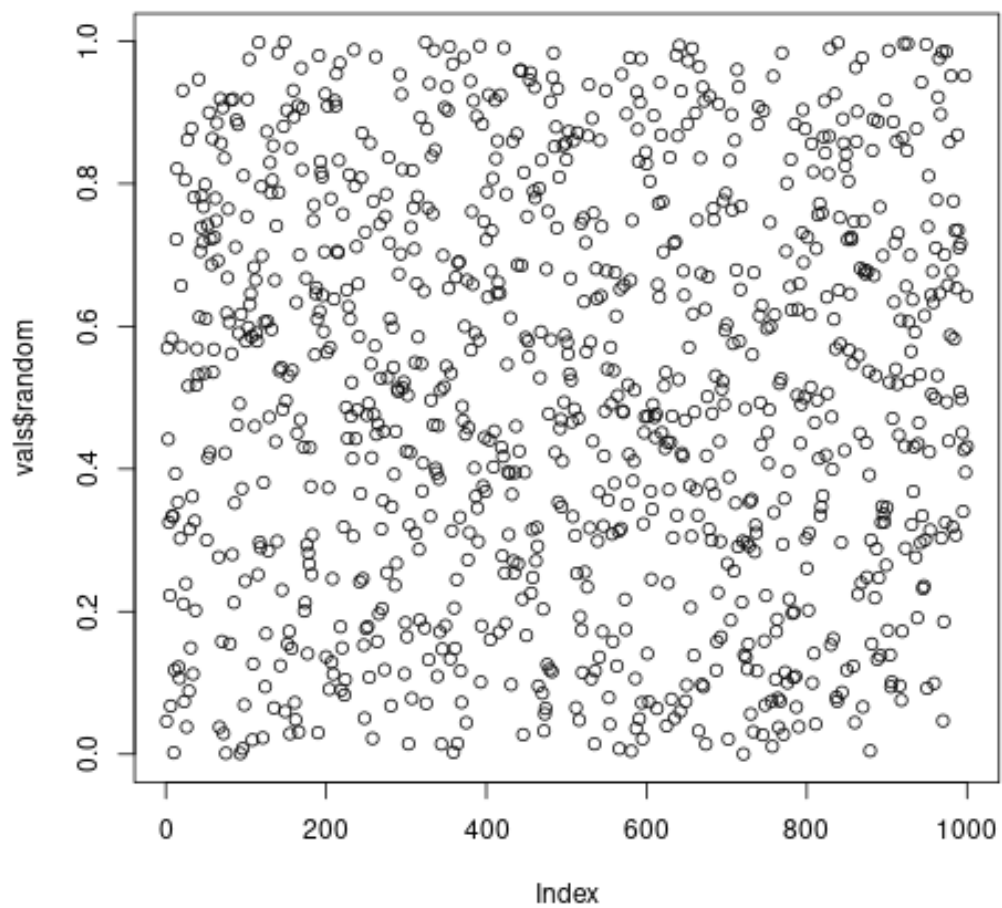
```

1000

```

```
1 real(8) :: rnd(n)
2 call random_number(rnd)
3 print'(i0, a, es20.8)', (i, " ", rnd(i), i = 1, n)
```

```
1 colnames(vals)[1:2] <- c("iterate", "random")
2 plot(vals$random)
```



2.4 adjustl, adjustr, trim

```
1  character(len=20) :: str = " gfortran "  
2  print'(2a)', "adjustl: ", "|"//adjustl(str)           //"|"  
3  print'(2a)', "adjustr: ", "|"//adjustr(str)           //"|"  
4  print'(2a)', "trim   : ", "|"//trim(str)              //"|"  
5  print'(2a)', "|      |", "|"//"
```

```
adjustl: |gfortran          |  
adjustr: |                  gfortran|  
trim   : | gfortran|  
|      ||                  |
```