

プログラミング言語 Python に関するメモ。

最終更新時間：2007 年 05 月 31 日 14 時 52 分 45 秒

Python とは

Python とは、Perl や Ruby などとともにスクリプト系言語と呼ばれ、手軽に利用できるプログラミング言語です。特に、Python は文法が簡素であることが特色です。

インタープリタ型言語として使用可能で、ベクトルやマトリックスといったクラスやグラフ描写も実装されているので、簡単な科学技術計算にも応用できます。

ただし、win 上では Python Shell の勝手があまりよくないので、私はコンソールをコマンドプロンプトから使っています。

モジュール (つかっているもの)

インストールについては[こちら](#)を参照のこと。使い方については、サンプルコードをみてください。

Numarray ベクトル・行列

NumPy/SciPy 科学技術計算

Matplotlib/PyLab グラフ

wxPython GUI。デフォルトで Tkinter もあります。

cgi に関しては、デフォルトで cgi というモジュールがあります。

サンプルコード：ベクトルと行列

```
from Numarray import *
vector = array( [1,2,3] )
vector * 5
dot(vector,vector)
matrix1 = array([ [1,2],
                  [3,4] ])
matrix2 = array([ [5,6],
                  [7,8] ])
matrix1*matrix2
dot(matrix1,matrix2)
from numarray.linear_algebra import *
inverse(matrix1)
```

単なる四則演算は、要素ごとの計算になります。内積、行列積の計算には、

dot()

関数をつかいます。また、スカラーとの演算は、すべての要素に影響します。

サンプルコード：グラフ

```
from scipy import *
from pylab import *
x = arange(0,10,0.1)
y = cos(x)
plot(y)
show()
```

RPy

RPy のサンプルコード。Win で Python 2.3 の場合 Python for Windows Extensions もインストールします。R もインストールしてパスをとっておきます。

```
#!/usr/bin/env python
# -*- coding: shift_jis -*-

from rpy import *

degrees = 4
grid = r.seq(0, 10, length=100)
values = [r.dchisq(x, degrees) for x in grid]
r.par(ann=0)
r.plot(grid, values, type='lines')
```

グラフを表示を表示しますが、そのウィンドウが固まっているため、グラフは実用になりません。

その他

```
help()
```

という関数も用意されているようですが、あまり使いやすくはありません。
例えば、

```
help(dot)
```

リンク

[Python Japan User's Group](#)

公式ホームページで、ドキュメント類が充実しています。

[kVerifeir Lab Home Page](#)python での行列・ベクトル数値計算

[紫藤のページ](#)

[Dive Into Python](#)

[Python のページ](#)

Tkinter プログラミング

コマンドプロンプトからつかう

Path をとおす (Windows XP)

コントロール パネル
システム
詳細設定タブ
環境変数
システム環境変数 (下段) 変数から Path を探して選択
編集をクリック
変数値 (下段) の最後に、例えば
;C:¥python24
を書き加える。

あとは、コマンドプロンプト (シェル) から

```
python
```

で起動します。

この場合、スクリプト (例えば上のサンプル) をコンソールにコピペするだけで、プログラムを実行できます。