

一般的じゃない、 Python入門

自己紹介

- ・名前: アベヒロキ (@hATrayflood)
- ・職業: 自宅サーバ管理者
- ・URL: <http://rayflood.org/diary/>
- ・住所: 長野市 地元: 東部町

Python

って何？

スクリプト言語

Perlとかか Rubyとかか の仲間

なぜか
日本では
マイナー

古参のPerlや
国産のRubyには
敵わないのか...

概要は

Wikipedia

を参照

インストーラ

リファレンス

作った人

※今回は

Python-2.*系の話

Python-3.*系は

知らん

では早速

リリース

```
print("hello world.")
```

いつものヤツ
ですが、流石に
コレだけじゃ...

```
# -*- coding: utf-8 -*-  
  
if __name__ == "__main__":  
    print(u"こんにちは")
```


ちよつとPython
っぽくなつた

```
> # -*- coding: utf-8 -*-
```

ソースファイルの 文字コードを指定

指定がないとLatin-1
と見なし、MB文字が
ソース中で使えない
コンパイルエラー

```
> if __name__ == "__main__":
```

ifの行末は「:」。括弧は使わない
明示的な閉じステートメントもない

`__name__ == "__main__"`は
スクリプトとして起動した
ときに実行するって目印

```
> print(u"こんにちは")
```

if内のステートメントか
どうかの判断は、次の行の
インデントで決まります

同一レベルのインデントで
同一スコープとみなす
ズレてるものがあると
コンパイルエラー

> u"こんにちは"

文字列リテラルの
先頭に「u」を付けると
Unicodeとして扱う

Pythonは基本的には
内部Unicodeだけど
リテラルを自動的に
Unicodeに変換しない

なんか奇っ怪…

次のお題

```
for i in range(4):  
    print("i = " + str(i))
```

i = 0

i = 1

i = 2

i = 3

`for(i = 0; i < 4; i++)`
みみたいなforは無い、
`range()`関数を使う

range()は数値の配列を生成する関数で、forのinに渡すのが主な使い方

なので、`range()`を
`print()`するとこうなる
> `print(str(range(4)))`
`[0, 1, 2, 3]`

range()の引数を変えれば
大体のことはできる

```
> range(10, 4, -2)  
[10, 8, 6]
```

> "i = " + str(i)

実は妙なところで型が厳密
混ぜるときは型変換が必要
もちろんこの逆も然り

join()は有名すぎるので省略

```
> "-" .join(["a", "b", "c"])  
'a-b-c'
```

正直、join()より
他の言語仕様の
ほうが奇妙...

次のお題

```
t = datetime.now()
t += timedelta(weeks=1)
t.strftime("%Y/%m/%d")
> '2011/05/28'
```

日付計算は
timedeltaが強力

weeks, days,
hours, minutes, seconds,
milliseconds, microseconds
が^レ使いたい放題

各項目併用可能
マイナス値も使用可能

```
t += timedelta(weeks=1, days=-1)  
> '2011/05/27'
```

こんなこともいける

```
t -= timedelta(weeks=1) / 3
t.strftime("%Y/%m/%d %H:%M:%S")
> '2011/05/19 07:20:00'
```

次のお題

hoge.py:

```
def hoge():  
    print("hoge")
```

```
import hoge  
hoge.hogehoge()
```

```
> hogehoge
```

オブジェクトじゃないのに
なんかオブジェクトっぽい
動きする。キモい。

```
from hoge import hoge hoge  
hoge hoge()
```

```
> hoge hoge
```

これはこれでimportが長い
パッケージ名を省略できる
ので後が楽といえれば楽

ところで、hoge.py自身は
パッケージであることを
明示していない

< fuga/_init_.py

< fuga/hoge.py

```
import fuga.hoge
```

```
fuga.hoge.hogehoge()
```

ファイルパスがそのまま
パッケージのパスになる
ただし__init__.pyが必要。
中身は空でもよい。

`__init__.py`にも関数や
クラス書ける。
ただし `from import`に限る

```
fuga/_init_.py:
```

```
def fugafuga():  
    print("fugafuga")
```

< fuga/_init_.py

```
from fuga import fugafuga  
fugafuga()
```

`__init__.py`に何か書く意味は
分からないけど、こんなことも
できるあたり、チェック目的か？

fuga/_init_.py:

```
raise("error") #print("error")
```

```
def fugafuga():  
    print("fugafuga")
```



```
import fuga.hoge
```

```
> raise("error") #print("error")  
> Traceback (most recent call last):  
> File "test.py", line 4, in <module>  
>     import fuga.hoge  
> File "fuga/_init_.py", line 1, in <module>  
>     raise("error") #print("error")  
> error
```

毒 例

実はライブラリ
作ってました。

rfZenHan

```
s = u"012ABCabc ! @ #アイウガダバ(株):"  
rfZenHan.normalize(s)  
> 012ABCabc!@#アイウガダバ(株):
```

※詳細はソースのコメント参照

元ネタ

仕様：[Perl Unicode::Japanese](#)

実装：[Python zenhan.py](#)

車輪の再発明万歳

eyeD3

mp3のID3タグを 編集するライブラリ

他の言語でまともに動く
ライブラリなかったのが
そもそもそのきっかけ

Python

サイスプレ

Google App Engine

Googleのオレオレ仕様な
Webアプリ実行環境
ただしサーバ性能は強力

言語はJava、Python、Go
フレームワークはそれぞれ
DBは独自で非SQL

Trac



言わずと知れた プロジェクト管理 バグ追跡システム

プラグイン豊富
svn以外にもhgや
gitも使えるらしい

Mercurial

バージョン管理システム
gitと同じ分散型
Web画面が標準である

Python、OpenJDK
Mozilla、その他
多数のOSSで採用

Firefoxビルド

高速化のために プロファイル採取 して最適化

つまりビルドの 過程で自動実行

ブラウザなので
Webサーバが必要

Pythonは標準で SimpleHTTPServer が使える

Python

サイスプレ

ま

と

め

奇妙な
言語です

仕様も
使う人も

奇妙さ故か
日本では
マイナーだけど

ニッチなところで
役には立ってるので

ここはあえて
この言葉で

Python

サイスプレ

ありがとうございます
ございました

リンク集

twitter hATrayflood

<http://twitter.com/hATrayflood>

Wikipedia Python

<http://ja.wikipedia.org/wiki/Python>

Python インストーラ

<http://www.python.org/download/>

Python 日本語リファレンス

http://www.python.jp/Zope/links/python_documents

作った人 ガイド・ヴァンロッサム

<http://ja.wikipedia.org/wiki/ガイド・ヴァンロッサム>

rfZenHan

<http://rayflood.org/diary-temp/rfZenHan-0.2.zip>

元ネタ 仕様

半角 \leftrightarrow 全角変換, Unicode::Japanese に
未公開機能が… daily dayflower

<http://d.hatena.ne.jp/dayflower/20060530/1148951624>

元ネタ 実装

blogSetomits Python 全角／半角変換モジュール zenhan.py
アルファベット・記号・数字・カタカナの全角・半角変換をする

<http://straitmouth.jp/blog/setomits/148>

Firefox PGOビルド

<http://mxr.mozilla.org/mozilla2.0/source/build/pgo/profileserver.py>

eyeD3

<http://eyed3.nicfit.net/>

Google App Engine

<http://code.google.com/intl/ja/appengine/>

Trac

<http://trac.edgewall.org/>

Mercurial

<http://mercurial.selenic.com/>