

WEB編 環境変数で「引数」をわたす

[TODO: 動物さん]

CGIスクリプトが実行されて、その結果がブラウザに渡るんだというところまで分かっていただけましたか。今回は、これに実行用パラメータを渡す方法を説明します。関数を実行するときに、引数によって動作を変えることができる点になぞらえて、不正確ながらこれを「引数」とか呼んだりすることもあります。イメージはわかりやすいでしょ。

こういうのが必要になる場面は、たくさんありますが、例えばこんなときです。電子書籍みたいなものを読ませるWEBアプリを書くとしたら、今表示されているページがどこかという情報を与えてもらわないといけません。関数で例えるなら、こんな動きをするものが必要でしょう。

```
def show_page(page_num):  
    # ここで、page_num ページ目を表示する
```

でもCGIスクリプトはそれ自体は関数みたいに書かれてはいません。こういう情報をどうやって受け渡せばよいでしょう。

その答えの一つが、「環境変数を通じて伝える」です。環境変数ってのは、pythonでいうと辞書みたいなもので、動いているプログラム（プロセス）ごとにこっそり与えられている情報です。ほとんどの場合意識したことはないでしょうが、人によっては、Windowsのシステムプロパティでこんな用語を目にしたこともあるかも知れません。

対話シェルを使って、pythonのプログラムが得られる環境変数をのぞいてみましょう。pythonでは、os モジュールにある environ という変数（関数じゃないです）の中身を見ると、環境変数が見られるようになっています。

```
>>> import os  
>>> print os.environ  
(表示結果はコンピュータごとに異なります)
```

かなり色々な情報が環境変数として用意されていることがわかりますね。この environ はpython辞書そのものですから、特定のキーに対応する値を見ることも簡単です。

```
>>> print os.environ.get('USERNAME', '?')  
(表示結果はコンピュータごとに異なります)
```

などとすればいいですね。

ふーん。じゃあ、さっきのCGIの例でいう「表示中のページ番号」なんてのがあればいいには、どういうキーで入ってくるんだろう。っていうかそもそも、WEBでアクセスするときに「何ページを表示しろ」っていう指示はどうやって与えたらいいんだろう。

その答えを知るために、まずは下のようなCGIスクリプトを作って、cgi-bin フォルダの中にセットしてください。ファイル名は、env.cgi とでもしましょうか。（スクリプトの一行目は、pythonのバージョンなどに応じて書き換えてくださいね。以下同様です。）

```
env.cgi
```

```
#!/C:¥python26¥python.exe

import os
ks = os.environ.keys()
ks.sort()
print "Content-Type: text/plain"
print ""

print "** show environment variables **"
print ""

for k in ks:
    print "%s : %s" % (k, os.environ[k])
```

この実行結果を知るためにアクセスすべきURLは、 <http://localhost/cgi-bin/env.cgi> [http://localhost/cgi-bin/env.cgi]。HTML形式で表示を整えるのが面倒なので、ここではただのテキストとして出力しています。(Content-Type: text/plain)

さっき対話シェルで見たときと比べて、情報がちょっと増えているのに気付かないでしょうか。SERVER_ とか REMOTE_ とかで始まるキーなんかがあったりして。これもCGIスクリプトの動作仕様のひとつなのですが、WEBサーバーのほうが、一定の環境変数をセットした状態でスクリプトを実行してくれるようになっているのです。

たとえばこの中の REMOTE_ADDR なんてのは、サーバーにアクセスしてきたコンピュータのIPアドレスを表します。また、SCRIPT_NAME なんてのが入っていて、これはアクセスされたURLのうち、パスの部分を表します。その他に色々ありますが、まあ、全部が必要なものではありません。

その中で、QUERY_STRING というキーがあるか探してみてください。あるとしたら、値はカラっぽですよ。または、キー自体が見つからないかもしれません。ここが、「引数」を受け取るための環境変数です。

今アクセスしているURLを、アドレス欄を直接書き換えて、下のようになさってください。編集が終わったら[Enter]。

<http://localhost/cgi-bin/env.cgi?page=25> [http://localhost/cgi-bin/env.cgi?page=25]

QUERY_STRING の値はどうなっていますか。page=25 っとなっているでしょう。この文字列から 25 という数字を取り出すのは、今まで練習問題をこなしてくれた人ならきっと楽勝ですね。「?」以降の文字をいろいろいじって、そのたびに QUERY_STRING の値が変わることを確認してください。

これが、ひとつのCGIスクリプトに色々な動作をさせるための秘密です。URLには「?」以降に何かの値を渡して、CGIスクリプトはその値を QUERY_STRING 環境変数を見ることで読みとる。で、その文字列の中身を分析して、それに従って、表示させるものをちょっとづつ変化させればいいのです。

この、何々=何々 という書き方は、WEBアプリに指示を渡すための共通の方法ということになっているようです。ために、よくあるWEBサイトも同じ方法を採用しているのか試してみましょう。googleの検索サービスなんてのは、「ザ・WEBアプリ」と呼んでいいほどのメジャーなシステムですね。なのでこれを使いましょう。さっそく <http://www.google.co.jp/> [http://www.google.co.jp/] にアクセスして、適当な検索語を調べてください。筆者は「python自習テキスト」で調べました。

で、調べ終わった後のアドレス欄はこんな風です。

```
http://www.google.co.jp/search?source=ig&hl=ja&rlz=&=&q=python%E8%87%AA%E7%BF%92%E3%83%86%E3%82%AD%E3%82%B9%E3%83%88&btnG=Google+%E6%A4%9C%E7%B4%A2&aq=f&oq=
```

(ブラウザによっては、日本語が直接アドレス欄の一部に表示されているでしょう。でも、メモ帳とかにコピーしてみると分かりますが、本当はこういう値ですよ)

このアドレスの、「?」以降を見ます。ついでにちょっと作業をしてください。まずはメモ帳に、「?」以降をコピーして、次に「&」という文字列を見つけて行をその前後で切り離してください。下みたい

```
source=ig
&
hl=ja
&
rlz=
&
=
&
q=python%E8%87%AA%E7%BF%92%E3%83%86%E3%82%AD%E3%82%B9%E3%83%88
&
btnG=Google+%E6%A4%9C%E7%B4%A2
&
aq=f
&
oq=
```

この QUERY_STRING の意味はこうです。「source は 'ig'、hl は 'ja'、rlz は空、q は 'python%E8%87%AA%E7%BF%92%E3%83%86%E3%82%AD%E3%82%B9%E3%83%88'、btnG は 'Google+%E6%A4%9C%E7%B4%A2'、aq は 'f'、oq は空、という条件でWEBアプリケーションを動作させてください」と。(途中の、単なる = になってる場所はきつと意味がありません)

こんな風に、何々=何々 という条件を、さらに「&」をくっつけて複数指定をするように書かれています。これで、かなり細かい条件の動作指定ができることがわかりますね。

- この、%E8%87%AA%E7%BF%92%E3... とかいう書き方は何なんだ、と思うことでしょう。URLの中に日本語などをそのまま含むことは禁止されていますので、電子メールのときみたいに、許可された文字の組み合わせでうまく書き表す必要があるのですよ。電子メールのときは違う、独特の変換ルールがここでは使われています。通称「URLエンコード」と呼ばれる変換ルールです。

さあ、あなたが書いたCGIスクリプトで、上のような複雑な QUERY_STRING を受け取ったら、上のように文字列をバラしながら「引数」を受け取ればよいというわけです。「&」でまず文字列全体をsplitし、それらをまた「=」でsplitし、値の部分がURLエンコード形式だったらそれをもとの日本語などに戻して...

...

そこそこやりごたえのある仕事になりそうですね。まあ、面倒だなあと思われても仕方ありません。さらに、こんなの誰が書いても同じなんだし、どっかに動くサンプルがあるのならそいつを流用して済ませたくなるところです。

はい、そういう仕事をしてくれるモジュールが、pythonには標準配布でそなわっていますよ。その名もcgiモジュールです。

cgiモジュール

cgiモジュールは、CGIスクリプトを書くときにいろいろと必要になるお決まりの処理が簡単にできるよう準備されているものです。いくつか機能がありますが、ここでは「実行パラメータ」を簡単に利用するための機能を紹介します。さっき説明した、QUERY_STRING の分析を自動的にやってくれるとかそういうものです。

cgiモジュールの中の、FieldStorage という関数がそれです。まずは対話シェルで使い方を見てみましょう。

```
>>> import cgi
>>> f = cgi.FieldStorage()
>>> print f
FieldStorage(None, None, [])
```

こんな感じで使います。cgi.FieldStorage関数の呼び出しに引数は要りません。これの実行結果として、FieldStorage型のオブジェクトが発生しますから、適当な変数（ここではf）に入れて受け止めるようにしてください。その後、fの値を色々調べると、どんな実行パラメータが指定されたのかがわかるようになっています。対話シェルでは全く意味のある内容は得られませんが、CGIの実行中なら、この例でfに入ったオブジェクトに下のような操作をすることで、それぞれの値を調べることができます。%E8%87%AA%E7%BF%92... とかいった風に変換されてしまっていた日本語も、元に戻った形で得られます。

なんせこれだけで、fの中身には、QUERY_STRINGを分析し終わった結果が入ります。だったら最初から環境変数が云々なんて言わないでこれだけ説明したらいいじゃないかと言われれば、ごもっとも。でもこのくらいの原理を知っておくのは別に損じゃないと思いますよ。

で、このfに入っているオブジェクトから好きなキーを使って値を調べるのは、下のような書き方です。

```
>>> f.getfirst('page', '')
```

'page' が、キーの名前。次の '' が、キーがなかったときにかわりに使う値です。なんでfirstとかいう語が混じっているのかというと、複数の値が指定されることもありうるからです。でも、今のところはそこらへんはいいでしょう。これだけ覚えておけば大体OKです。

例では対話シェルから試しているのが当然意味のある値が取得できませんでしたが、これがCGIスクリプトとして動くならもうちょっと面白いです。下のスクリプトを page.cgi として cgi-bin フォルダに置きましょう。

page.cgi

```
#!/C:¥python26¥python.exe

import cgi
f = cgi.FieldStorage()
pagenum = f.getfirst('page', '(default)')

print "Content-Type: text/plain"
print ""

print " show page # %s !" % pagenum
```

で、<http://localhost/cgi-bin/page.cgi> [<http://localhost/cgi-bin/page.cgi>] にアクセスしましょう。このままだと、page # (default) とかいうメッセージが出てきますが、アドレス欄に ?page=25 とかなんとか文字を付け足すと、それに対応したメッセージが出てくるようになりました。いかがでしょうか。

今回はHTMLでないテキストで出力しましたが、これだと面白さも今一步です。今度はこいつをちゃんと整形したHTMLとして出力して、表示しているページから前後のページに移動できるようなものをサンプルとしてつくってみましょうか。

そのためには、HTMLの重要な特徴である「ハイパーリンク」が表現できるようになる必要があります。考えてみれば、これはまだ一度も説明していませんでした。だから次回にこれの基本を説明しつつ、CGIで「画面遷移」を実現するところを説明したいです。