

# 熱血講義

프리렉의 열혈강의 시리즈

## Python 파이썬



1

파이썬 (Python)

## Python

❖ 15

: /

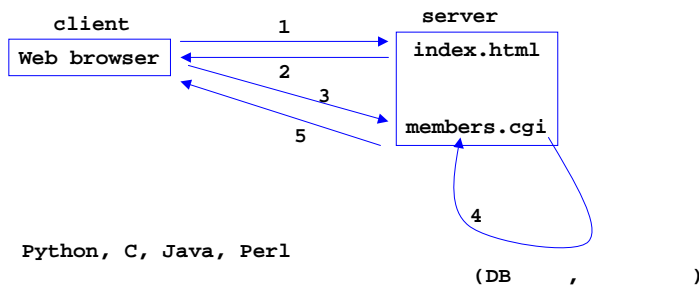
( gslee@mail.gwu.ac.kr<sub>2</sub> )



1.
  1. CGI
    1. ,
    2. form
    - 3.
  2. DB
2.
  1. urllib
  2. httplib

## CGI

- CGI(Common Gateway Interface)?

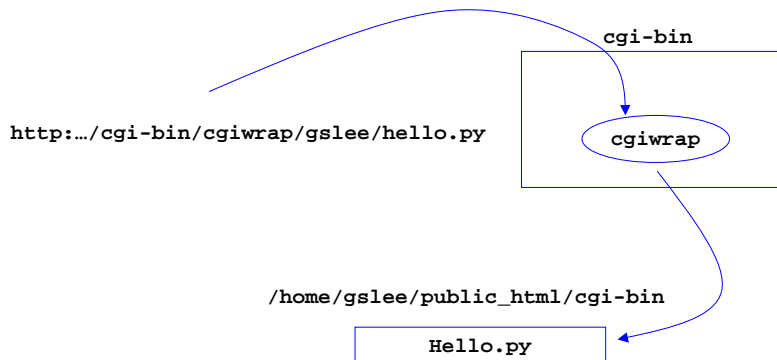


# CGI

- CGI
  - 가
  - `/etc/httpd/conf/httpd.conf`  
( )
  - : `/var/www/cgi-bin/`
- CGI
  - `/public_html/cgi-bin`
  - 가 `cgiwrap`
- `cgiwrap`
  - <http://sourceforge.net/projects/cgiwrap>

5

## • `cgiwrap`



6

- **hello.py**
- **chmod a+x hello.py**

<http://icsh.gwu.ac.kr/cgi-bin/cgiwrap/gslee/hello.py>

```
#!/usr/local/bin/python

print "Content-Type: text/plain\n\n"
print "Hello, Python!" # print a test string
```

# CGI

( )

### CGI

➤ : <http://apache.kr.net/>  
가

- C:/Program Files/Apache Group/Apache/htdocs
- C:/Program Files/Apache Group/Apache/cgi-bin
- 
- 
- <http://localhost/cgi-bin/hello.py>

```
#!c:/python22/python

print "Content-Type: text/plain\n\n"
print "Hello, Python!" # print a test string
```

## CGI

( )

- PWS CGI
  -
- IIS CGI
  - <http://python.kwangwoon.ac.kr/pymoinmoin/moin.cgi>
  - ‘IIS CGI’

## text/html

- count.py

```
#!/usr/local/bin/python
print "Content-Type: text/html\n\n"
html_header = '''<HTML>
<HEAD></HEAD>
<BODY> '''
html_footer = '''</BODY>
</HTML> '''

print html_header
for k in range(10):
    print k, '<br>'
print html_footer
```

## CGI

- - `import cgi`  
`cgi.enable()`
- - `import cgi`  
`cgi.enable(display=0,`  
`logdir="/tmp")`

## CGI

- ```
addr = os.environ['REMOTE_ADDR']
os.environ['NAME'] = 'gslee'
```
- **test.py**

```
#!/usr/local/bin/python
import cgi, os

cgi.test()
```

## cgi

- cgi

- “GET”, “POST”

- 

```
import cgi

form = cgi.FieldStorage()      #
print form.keys(), form.values() #

#
print form["email"].value      # "email"
                                # email 가 exception
email = form.getvalue("email","") # email 가
                                # "" . No exception.
```

13

## Form

- prfield.html

```
<form name="form" method="get"
action="http://icsh.gwu.ac.kr/cgi-
bin/cgiwrap/gslee/prfield.py">

<p>
email:<input type="text" name="email">
name:<input type="text" name="name">
<input type="submit" value=" " >
</p>
</form>
```

14

- prfield.py

```
#!/usr/local/bin/python

import cgi

print "Content-Type: text/plain\n\n"
form = cgi.FieldStorage()
for name in form.keys():
    print "Input: " + name + " value: " + form[name].value
print "Finished!"
```

15

Composer.html

```
<form name="form1" method="post" action="http://icsh.gwu.ac.kr/cgi-
bin/cgiwrap/gslee/composer.py">
<p>          가          </p>
<p><input type="checkbox" name="composer" value="Bach">Bach</p>
<p><input type="checkbox" name="composer" value="Handel">Handel</p>
<p><input type="checkbox" name="composer" value="Mozart">Mozart</p>
<p><input type="checkbox" name="composer" value="Beethoven">Beethoven</p>
<p><input type="checkbox" name="composer" value="Debussy">Debussy</p>
<p><input type="checkbox" name="composer"
        value="Tchaikovsky">Tchaikovsky</p>
<p><input type="submit" value="    "></p>
</form>
```

16



## 파이썬 (Python)

```
#!/usr/local/bin/python
# composer.py
import cgi
def getCheckList():
    form = cgi.FieldStorage()
    checkList = form.getvalue("composer", "")
    if type(checkList) == type([]):
        return checkList
    else:
        return [checkList]
if __name__ == '__main__':
    print 'Content-type: text/plain\n\n'
    checklist = getCheckList()
    for item in checklist:
        print item
```

17

## 파이썬 (Python)

Upload.html

```
<form name="form" method="post"
    action="http://ics.h.gwu..ac.kr/cgi-
    bin/cgiwrap/gslee/upload.py"
    enctype="multipart/form-data">
<p><input type="text" name="filename"></p>
<p><input type="file" name="file" size="35"></p>
<p><input type="submit" value="    "></p>
</form>
```

18

```
#!/usr/local/bin/python
#upload.py
import cgi, os

def upload(form):
    try:
        filename = form['file'].filename
        filestr = form['file'].value
    except:
        return None
    f = open(filename, 'wb')
    f.write(filestr)
    return 1

...
```

19

### ➤ BaseCookie, SimpleCookie, SerialCookie, SmartCookie

```
>>> c = Cookie.SimpleCookie()
>>> c['name'] = 'gslee'          #
>>> print c                      # HTTP
Set-Cookie: name=gslee;
>>> c['name']['expires'] = 100    # 100
>>> print c
Set-Cookie: name=gslee; expires=Mon, 14-Jan-2002 03:22:21 GMT;
```

20

## 파이썬 (Python)

```
#!/usr/local/bin/python

import Cookie

c = Cookie.SimpleCookie()      #
c['name'] = 'gslee'           #
print c                        #
print 'Content-Type: text/plain\n\n'
print '                        .'
```

21

## 파이썬 (Python)

```
#!/usr/local/bin/python

import os
import Cookie

print 'Content-Type: text/plain\n\n'
c = Cookie.SimpleCookie()      #
c.load(os.environ["HTTP_COOKIE"]) #
print c.keys()                 #          가          가
print c['name']                 # 'name'      ?
print c['name'].value           #          value          .
```

```
['name']
Set-Cookie: name=gslee;
gslee
```

22

## DB

- Database Modules
  - <http://www.python.org/topics/databases/modules.html>
- MySQL
  - Linux, Windows
- MS SQL Server
  - Windows

23

- Python DB API Specification 2.0
  - <http://www.python.org/topics/databases>
- MySQLdb
  - <http://sourceforge.net/projects/mysql-python>
  - :

24

- **ODBC** **SQL Server**

- **ODBC**

- Win32all  
(<http://starship.python.net/crew/mhammond/win32/>)< >
- mxODBC
  - <http://www.egenix.com/>

- :

25

## ASP

- - **Win32all**
    - <http://starship.python.net/crew/mhammond/win32/>
  -
- - **Request, Response**

26

- urllib
- httplib

## urllib

- URL
  - URL - urlencode
  - URL Quoting - quote
  - URL Unquoting - unquote
- 가 (urlopen)
  - http 가 (http:)
  - ftp 가 (ftp:)
  - 가 (file:)

- CGI

- GET

```
>>> import urllib
>>> params = urllib.urlencode({'spam': 1, 'eggs': 2, 'bacon': 0})
>>> f = urllib.urlopen("http://www.musi-cal.com/cgi-bin/query?%s" % params)
>>> print f.read()
```

- POST

```
>>> import urllib
>>> params = urllib.urlencode({'spam': 1, 'eggs': 2, 'bacon': 0})
>>> f = urllib.urlopen("http://www.musi-cal.com/cgi-bin/query", params)
>>> print f.read()
```

29

- - **urlretrieve(ur[, filename[, reporthook[, data]])**

30

# httplib

- HTTP

- 
- 
- 

# HTTP

- HTTP/1.1 - RFC 2616

- HTTP Message

- Request

- →

- Response

- →

- Message

- Header

- Body



- Request

1. Request-Line
2. Headers
3. CRLF
4. [Message-Body]

- Request-Line

- Method SP Request-URI SP HTTP-Version CRLF
- Method : GET, POST, HEAD, PUT, ...

GET http://www.w3.org/pub/WWW/TheProject.html HTTP/1.1

- IE 6.0 Request

<http://www.python.or.kr:8080/python>

```
GET /python HTTP/1.1
Accept: */*
Accept-Language: ko
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows 98)
Host: www.python.or.kr:8080
Connection: Keep-Alive
Cookie: TProjs_where="main"; poll_length="long"; ...
```

- **Response**

1. **Status-Line**
2. **Headers**
3. **CRLF**
4. **[Message-Body]**

- **Status-Line**

- **HTTP-Version SP Status-Code SP Reason-Phrase CRLF**

`HTTP/1.1 200 OK`

35

- **Response**

<http://www.python.or.kr:8080/python>

`HTTP/1.1 200 OK`

`Server: Zope/Zope 2.3.1 (binary release, python 1.5.2,  
linux2-x86) ZServer/1.1b1`

`Date: Thu, 26 Sep 2002 11:51:19 GMT`

`Content-Length: 17134`

`Content-Type: text/html`

`<!DOCTYPE HTML PUBLIC "-//w3C//DTD HTML3.2//EN">`

`<html>`

`<head>`

36

# httplib

## • httplib

## Request

```
request = HTTP(address, [port])      # request
request.putrequest(method, file)      # Request-Line
request.putheader(header-type, value) # header
...                                  # header
request.endheaders()                  #
request.send(data)                     #
```

37

# httplib

## • httplib

## Response

```
errorCode, errorMessage, headers = request.getreply()
# errorCode
# header (mimetools.Message - )
fobj = request.getfile() #
html = fobj.read() #
fobj.close()
```

38

# httplib

## • Response header

- **headers.getplist()**
  - Content-Type *name=value*
- **headers.getparam(name)**
  - *name=value* *value*
- **headers.getencoding()**
  - Content-Transfer-Encoding
- **headers.gettype()**
  - Content-Type
  - *"type/subtype"*
- **headers.getmaintype()**
- **headers.getsubtype()**

39

# httplib

## • Request

## Response

```
>>> from httplib import HTTP
>>> request = HTTP('www.python.or.kr', 8080)
>>> request.putrequest('GET', '/python')
>>> request.putheader('Accept', 'text/html')
>>> request.putheader('User-Agent', 'PythonAgent')
>>> request.putheader('Host', 'www.python.or.kr:8080')
>>> request.endheaders()
>>> errorCode, errorMessage, headers = request.getreply()
>>> print errorCode, errorMessage
200 OK
>>> fobj = request.getfile()
>>> html = fobj.read()
>>> fobj.close()
>>> print html      #      print unicode(html, 'mbcs')
```

```
>>> headers.getencoding()
'7bit'
>>> headers.gettype()
'text/html'
>>> headers.getmaintype()
'text'
>>> headers.getsubtype()
'html'
>>> headers.keys()
['date', 'connection', 'content-type', 'content-length', 'server']
>>> headers['date']
'Fri, 27 Sep 2002 00:37:58 GMT'
>>> headers['connection']
'close'
>>> headers['server']
'Zope/Zope 2.3.1 (binary release, python 1.5.2, linux2-x86) ZServer/1.1bl'
```

40

# httpplib

- **cgi**
  - `request.putrequest('GET', '/query.py?' + urllib.quote('name=gslee'))`
- - `request.putheader('Cookie', 'key=value')`
- **POST**
  - `request.putrequest("POST", "....cgi")`
  - `request.putheader("Content-Length", str(len(bodyData)))`
  - `request.endheaders()`
  - `request.send(bodyData)`