

검색어를 입력하세요.



 <b>점프 투 파이썬 - 라이브러리 예제 편</b> (/book/5445)
00장 들어가기 전에
00-01 머리말
00-02 저자소개
00-03 주요변경이력
00-04 책 구입 안내
00-05 이 책을 읽기 전에
01장 텍스트 다루기

001 문자열을 줄여 표시하려면? — <code>textwrap.shorten</code>
002 긴 문장을 줄 바꿈하려면? — <code>textwrap.wrap</code>
003 정규표현식으로 개인정보를 보호하려면? — <code>re</code>
02장 바이너리 데이터 다루기
004 C로 만든 데이터를 출력하려면? — <code>struct</code>
03장 다양한 데이터 다루기
005 날짜를 계산하고 요일을 알려면? — <code>datetime.date</code>
006 두 날짜의 차이를 알려면? — <code>datetime.timedelta</code>
007 2월이 29일인 해를 알려면? — <code>calendar.isleap</code>
008 앞뒤에서 자료를 넣고 빼려면? — <code>collections.deque</code>
009 자료에 이름을 붙이려면? — <code>collections.namedtuple</code>
010 사용한 단어 개수를 구하려면? — <code>collections.Counter</code>
011 딕셔너리를 한 번에 초기화하려면? — <code>collections.defaultdict</code>
012 수상자 3명을 선정하려면? — <code>heapq</code>
013 데이터를 보기 좋게 출력하려면? — <code>pprint</code>
014 점수에 따른 학점을 구하려면? — <code>bisect</code>
015 숫자에 이름을 붙여 사용하려면? — <code>enum</code>
016 수강할 과목의 순서를 구하려면? — <code>graphlib.TopologicalSorter</code>
04장 수학과 숫자 다루기
017 과자를 똑같이 나누어 담으려면? — <code>math.gcd</code>
018 버스가 동시에 도착할 시각을 알려면? — <code>math.lcm</code>
019 소수점을 정확하게 계산하려면? — <code>decimal.Decimal</code>
020 분수를 정확하게 계산하려면? — <code>fractions</code>

021 로또 번호를 뽑으려면? — random

022 시험 결과의 평균값과 중앙값을 알려면? — statistics

## 05장 함수형 프로그래밍 다루기

023 상담원을 순서대로 배정하려면? — itertools.cycle

024 연간 매출액을 계산하려면? — itertools.accumulate

025 키값으로 데이터를 묶으려면? — itertools.groupby

026 부족한 것을 채워 묶으려면? — itertools.zip\_longest

027 순서를 생각하며 카드를 뽑으려면? — itertools.permutations

028 로또의 모든 가짓수를 구하려면? — itertools.combinations

029 순서대로 좌표를 정렬하려면? — functools.cmp\_to\_key

030 웹 페이지를 임시로 저장하려면? — functools.lru\_cache

031 기존 함수로 새로운 함수를 만들려면? — functools.partial

032 함수를 적용하여 하나의 값으로 줄이려면? — functools.reduce

033 래퍼 함수의 속성을 유지하려면? — functools.wraps

034 다양한 기준으로 정렬하려면? — operator.itemgetter

## 06장 파일과 디렉터리 다루기

035 파일 경로를 객체로 다루려면? — pathlib

036 디렉터리의 구성을 알려면? — os.path

037 여러 개의 파일을 한꺼번에 읽으려면? — fileinput

038 디렉터리와 파일을 비교하려면? — filecmp

039 임시로 만든 파일을 이용하려면? — tempfile

040 파일을 찾으려면? — glob

041 특정 파일만 찾으려면? — fnmatch
042 파일에서 무작위로 한 줄만 가져오려면? — linecache
043 파일을 복사하거나 이동하려면? — shutil
07장 데이터 저장하고 관리하기
044 객체를 파일로 저장하고 불러오려면? — pickle
045 객체 변경에 따른 오류를 방지하려면? — copyreg
046 딕셔너리를 파일로 저장하려면? — shelve
047 블로그 데이터를 저장하려면? — sqlite3
08장 데이터 압축하고 보관하기
048 데이터 크기를 줄여 전송하려면? — zlib
049 데이터를 압축하여 파일로 저장하려면? — gzip
050 bzip2 알고리즘으로 압축하려면? — bz2
051 LZMA 알고리즘으로 압축하려면? — lzma
052 여러 파일을 zip으로 합치려면? — zipfile
053 여러 파일을 tar로 합치려면? — tarfile
09장 다양한 형식의 파일 다루기
054 CSV 파일을 읽고 쓰려면? — csv
055 설정 파일에서 정보를 읽으려면? — configparser
10장 암호문 다루기
056 비밀번호를 암호화하여 저장하려면? — hashlib
057 메시지 변조를 확인하려면? — hmac
058 안전한 난수를 생성하려면? — secrets

## 11장 운영체제 다루기

059 문자열을 파일처럼 다루려면? — `io.StringIO`

060 명령행 옵션을 지정하여 실행하려면? — `argparse`

061 디버깅용 로그를 남기려면? — `logging`

11-04 `getpass` - 에코없이 암호 입력하기

11-05 `curses` - 터미널 그래픽 애플리케이션

11-06 `platform` - 시스템 정보 확인하기

11-07 `ctypes` - C 라이브러리 사용하기

## 12장 동시실행

12-01 `threading` - 스레드 기반의 병렬처리

12-02 `multiprocessing` - 프로세스 기반의 병렬처리

12-03 `concurrent.futures` - 병렬작업 실행하기

12-04 `subprocess` - 시스템 명령어 실행

12-05 `sched` - 이벤트 스케줄러

## 13장 네트워킹과 프로세스간 통신

13-01 `asyncio` - 비동기 I/O

13-02 `socket` - 저수준 네트워킹 인터페이스

13-03 `ssl` - SSL이 적용된 소켓

13-04 `select` - I/O 멀티플렉싱

13-05 `selectors` - 고수준 I/O 멀티플렉싱

13-06 `signal` - 시그널 처리

## 14장 인터넷 데이터 처리

14-01 json - JSON 데이터 처리
14-02 base64 - 바이너리 데이터를 문자열로 변환
14-03 binascii - 16진수 문자열 처리
14-04 quopri - quoted-printable 인코딩
14-05 uu - 바이너리 파일을 텍스트 파일로 변환
15장 구조화된 마크업 처리도구
15-01 html - HTML 문자 이스케이프 처리하기
15-02 html.parser - HTML 문서 분석하기
15-03 xml.etree.ElementTree - XML 문서 작성하기
15-04 xml.etree.ElementTree - XML 문서 분석하기
16장 인터넷 프로토콜과 지원
16-01 webbrowser - 편리한 웹 브라우저 제어기
16-02 cgi - CGI 프로그램 작성하기
16-03 cgiitb - CGI 오류 확인하기
16-04 wsgiref - WSGI 프로그램 작성하기
16-05 urllib - URL 처리하기
16-06 http.client - HTTP 클라이언트
16-07 ftplib - FTP 클라이언트
16-08 poplib - POP3 이메일 확인하기
16-09 imaplib - IMAP4 이메일 확인하기
16-10 nntplib - 뉴스 그룹 조회하기
16-11 smtplib - 파일 첨부하여 이메일 보내기

16-12 telnetlib - 텔넷 클라이언트
16-13 uuid - 유일한 ID 생성
16-14 socketserver - 소켓서버 프레임워크
16-15 http.server - HTTP서버
16-16 xmlrpc - XMLRPC 서버와 클라이언트
17장 기타 서비스
17-01 imghdr - 이미지 유형 판단
17-02 turtle - 터틀 그래픽
17-03 cmd - 사용자 친화적인 명령행 프로그램 만들기
17-04 shlex - 간단한 어휘 분석
17-05 tkinter - 편리한 GUI 툴킷
17-06 unittest - 단위테스트
17-07 doctest - 독스트링을 이용한 테스트
17-08 timeit - 간단한 코드의 실행시간 측정하기
17-09 pdb - 파이썬 디버거
17-10 sys.argv - 파이썬 스크립트에 파라미터 전달
17-11 dataclasses - 데이터클래스
17-12 abc - 추상클래스
17-13 atexit - 종료 처리기
17-14 traceback - 오류 추적하기
17-15 typing - 타입 힌트 지원
18장 외부 라이브러리

18-01 pip - 파이썬 라이브러리 설치
18-02 requests - 간편한 HTTP 클라이언트
18-03 diff_match_patch - 문자열의 차이
18-04 faker - 가짜 데이터 생성기
18-05 sympy - 방정식 풀이
18-06 pyinstaller - 파이썬 프로그램을 exe 파일로 배포하기
마치며
부록
01 파이썬과 유니코드
02 클로저와 데코레이터
03 이터레이터와 제너레이터
04 파이썬 타입 어노테이션
05 str과 repr

Published with WikiDocs (/)



■ 점프 투 파이썬 - 라이브러리 예제 편 (/book/5445) / 16장 인터넷 프로토콜과 지원 (/128226)  
/ 16-02 cgi - CGI 프로그램 작성하기 (/129281)

🏠 WikiDocs (/)

# 16-02 cgi - CGI 프로그램 작성하기

cgi 모듈은 CGI 프로그램을 만들기 위한 도구를 제공하는 모듈이다.



CGI란 공통 게이트웨이 인터페이스(Common Gateway Interface)의 약어로 웹서버와 외부 프로그램 사이에서 정보를 주고받는 방법이나 규약들을 말한다.

## 문제

두 수 a, b의 입력을 받아서 그 곱을 리턴해 주는 CGI 프로그램을 작성하시오.

예를 들어 서버의 주소가 52.78.8.100 이고 아파치 웹 서버가 8088 포트로 서비스를 하고 있을때 다음과 같은 URL을 요청 하면

```
http://52.78.8.100:8088/cgi-bin/multiple.py?a=3&b=5
```

다음과 같은 결과가 출력되도록 CGI 프로그램을 작성해야 한다.

```
Result: 15
```

※ 아파치 서버에서 파이썬 프로그램을 실행하는 방법은 아래 "아파치 설치 및 설정하기"를 참고하자.

## 풀이

URL로 전달받은 두 개의 값 a, b를 얻기 위해서는 `cgi.FieldStorage` 클래스가 필요하다.

`cgi.FieldStorage` 클래스를 사용하면 다음과 같이 a, b 값을 얻을 수 있다.

```
import cgi
form = cgi.FieldStorage()

a = form.getvalue('a')
b = form.getvalue('b')
```

`cgi.FieldStorage()` 로 생성한 form객체에 `getvalue(파라미터명)` 를 호출하여 URL로 전달된 값을 얻을 수 있다. 이 때 URL로 얻은 2개의 값은 숫자가 아닌 문자열이므로 다음과 같이 숫자로 바꾸어 곱해야 한다.

```
result = int(a) * int(b)
```

그리고 웹 브라우저에 결과값을 출력하기 위해서 HTTP 규약에 의해 "Content-type" 항목과 빈줄을 포함하여 다음처럼 출력해야 한다.

```
print('Content-type: text/plain')
print()
print(f'Result:{result}')
```

두 수의 곱을 계산하기 위한 최종 CGI 프로그램인 `multiple.py` 는 다음과 같다.

[파일명: `/var/www/cgi-bin/multiple.py`]

```
#!/usr/bin/python3
import cgi
form = cgi.FieldStorage()

a = form.getvalue('a')
b = form.getvalue('b')

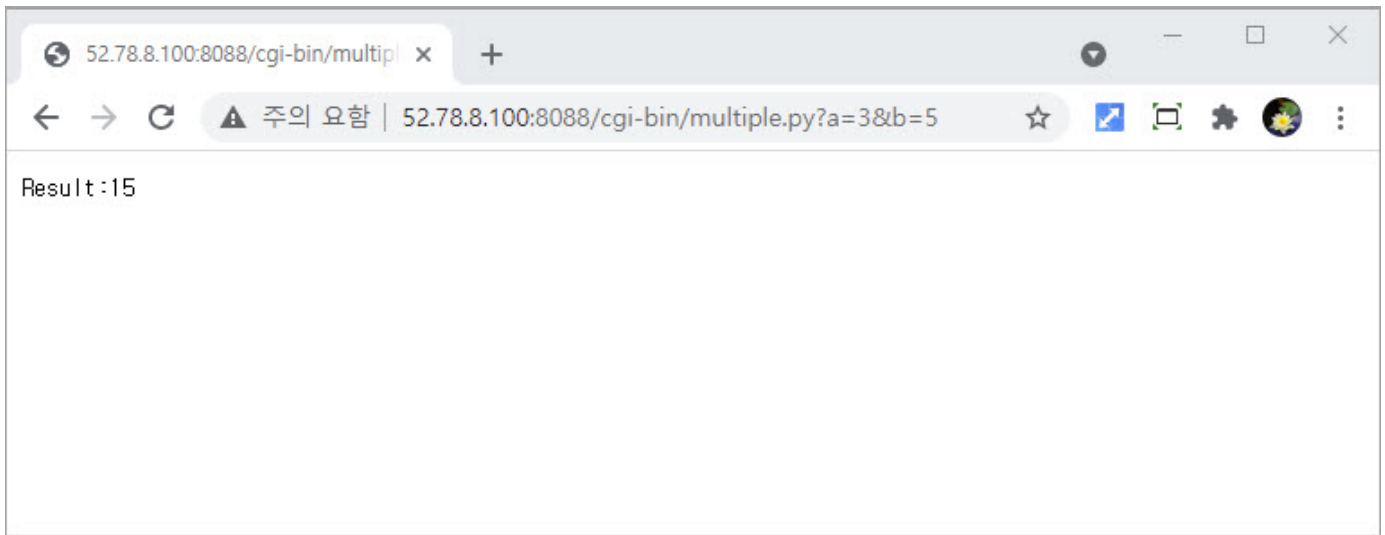
result = int(a) * int(b)

print('Content-type: text/plain')
print()
print(f'Result:{result}')
```

파일명이 `/var/www/cgi-bin/multiple.py` 인 이유는 아파치가 CGI 프로그램을 실행할 수 있는 디렉터리로 `/var/www/cgi-bin` 을 설정했기 때문이다. 아파치 설정에 대한 자세한 내용은 아래에서 확인해 보자.

맨 윗줄의 `#!/usr/bin/python3` 의 의미는 아파치가 `multiple.py` 파일을 호출할 때 사용할 파이썬 프로그램의 경로를 의미한다. 즉, `multiple.py` 파일을 `/usr/bin/python3` 파일로 실행하겠다는 의미이다. 유닉스 환경에서는 파일을 단독으로 실행하기 위해서 이렇게 파일 상단에 해당 파일을 실행할 때 호출해야 하는 프로그램의 전체 경로(#! 를 포함한 경로)를 적는데, 이것을 쉬방(shebang) 커맨드라고 한다.

호출한 결과는 다음과 같다.



## 아파치 설치 및 설정하기

리눅스(예: 우분투)에서 파이썬 CGI 프로그램을 사용하기 위해서는 다음과 같은 절차를 따라야 한다.

### tip

#### 파이썬 CGI 서버 만들기

아파치와 같은 웹 서버 설치없이 파이썬만으로 웹 서버를 구동하는 방법은 다음을 참고하자.

참고: 16-15 http.server - HTTP서버 (<https://wikidocs.net/131607>)

## 아파치 설치

먼저 다음처럼 아파치를 설치한다.

```
$ sudo apt-get install apache2
```

## 포트 변경

기본 HTTP 포트인 80과 SSL 포트인 443 대신 다른 포트를 사용해야 한다면 다음 파일을 변경한다.

[파일명: /etc/apache2/ports.conf]

```
Listen 8088

<IfModule ssl_module>
    Listen 8443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 8443
</IfModule>
```

80을 8088로 443을 8443으로 변경해 주었다.

## CGI 설정

아파치가 파이썬 프로그램을 호출하려면 다음과 같은 설정이 필요하다.

```
ScriptAlias /cgi-bin/ /var/www/cgi-bin/
```

ScriptAlias는 `http://52.78.8.100:8088/cgi-bin/multiple.py` 와 같은 `/cgi-bin/` 으로 시작되는 URL을 호출했을 때 `/var/www/cgi-bin/` 디렉터리의 파일을 읽게 하는 설정이다. 따라서 위와 같이 설정한 후 웹 브라우저에서 `http://52.78.8.100:8088/cgi-bin/multiple.py` URL을 호출하면 서버의 `/var/www/cgi-bin/multiple.py` 파일이 호출될 것이다.

그리고 `/var/www/cgi-bin` 디렉터리는 다음과 같이 설정해야 한다.

```
<Directory /var/www/cgi-bin>
    Options +ExecCGI
    AddHandler cgi-script .py
</Directory>
```

Options +ExecCGI 는 `/var/www/cgi-bin` 디렉터리가 CGI 파일을 실행할 수 있는 경로라는 의미이고 AddHandler cgi-script .py 는 CGI 파일로 .py 확장자에 해당되는 파이썬 스크립트를 사용하겠다는 의미이다.

이러한 CGI 설정들을 적용하려면 아파치의 다음 파일을 수정해야 한다.

[파일명 : `/etc/apache2/sites-enabled/000-default.conf`]

```
<VirtualHost *:8088> # 80 포트를 8088 포트로 변경
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    ScriptAlias /cgi-bin/ /var/www/cgi-bin/
    <Directory /var/www/cgi-bin>
        Options +ExecCGI
        AddHandler cgi-script .py
    </Directory>
</VirtualHost>
```

80 포트 대신 8088로 변경했다면 위와 같이 `<VirtualHost *:8088>` 로 변경해 주어야 한다.

그리고 마지막으로 아파치가 cgi 기능을 사용할 수 있도록 `cgi.load` 파일을 `enable`해 준다.

```
$ cd /etc/apache2/mods-enabled
$ sudo ln -s ../mods-available/cgi.load
```

그리고 `cgi-bin` 디렉터리를 다음과 같이 생성한다.

```
$ cd /var/www/
$ sudo mkdir cgi-bin
```

그리고 작성한 `multiple.py` 파일을 `/var/www/cgi-bin` 디렉터리로 이동한다.

```
$ sudo mv ~/multiple.py /var/www/cgi-bin
```

위의 `mv` 명령은 `multiple.py` 파일이 홈 디렉터리에 있다고 가정한 경우이다.

## CGI 파일 권한

작성한 CGI 파일(예: `multiple.py`)은 아파치가 실행할 수 있도록 다음과 같이 실행권한을 주어야 한다.

```
$ cd /var/www/cgi-bin  
$ chmod a+x multiple.py
```

## 아파치 재시작

아파치 설정이 바뀌었으므로 다음과 같이 아파치를 재시작해 주어야 한다.

```
$ sudo systemctl restart apache2.service
```

## 참고

---

<https://docs.python.org/ko/3/library/cgi.html> (<https://docs.python.org/ko/3/library/cgi.html>)

마지막 편집일시 : 2021년 7월 16일 10:48 오후

댓글 0

피드백

- **이전글** : 16-01 webbrowser - 편리한 웹 브라우저 제어기
- **다음글** : 16-03 cgiitb - CGI 오류 확인하기

↑ TOP