

Ask Django

여러분의 파이썬/장고 페이스메이커가 되겠습니다.

EP 18. HTML Form

HTML Form #ref

웹페이지에서는 form 태그를 통해, 데이터를 전송
ex) 로그인 폼, 댓글 폼

하나의 form 태그는 하나 이상의 위젯 (Widget) 을 가진다.

```
<form action="" method="POST">  
  <input type="text" /> : 1줄 문자열 입력  
  <textarea></textarea> : 1줄 이상의 문자열 입력  
  <select></select> : 항목 중 택일  
  <input type="checkbox" /> : 체크박스 (한 그룹 내 여러 항목을 다수 선택 가능)  
  <input type="radio" /> : 라디오박스 (한 그룹 내 여러 항목 중에 하나만 선택 가능)  
  그 외 다수 위젯  
</form>
```

HTML Form 태그 필수 속성

- action : 요청을 보낼 주소
- method
 - "GET" : 주로 데이터 조회 요청 시
 - "POST" : 파괴적인 액션 (생성/수정/삭제) 에 대한 요청 시

- enctype : request.POST 요청 시에만 유효
 - application/x-www-form-urlencoded (디폴트)
 - 파일 업로드 불가
 - multipart/form-data : 파일 업로드 가능
 - text/plain : 스펙에는 정의되어있으나, 실제로 쓰이지는 않음.

url encoded

key=value 값의 쌍이 & 문자로 이어진 형태

공백은 +로 인코딩하며, Special 문자들은 ASCII 16진수로 변경

```
from urllib.parse import urlencode
print(urlencode({'key1': 'value1', 'key2': 10, 'name': '공유'}))
print('공유'.encode('utf8'))
print(' '.join('%{:X}'.format(ch) for ch in '공유'.encode('utf8')))
```

key2=10&key1=value1&name=%EA%B3%B5%EC%9C%A0

b' \xea\x b3\x b5\x ec\x 9c\x a0 '

%EA%B3%B5%EC%9C%A0

Form Method

Form Method

- GET 방식 : 엽서에 비유. 물건을 보낼 수 없다.
 - application/x-www-form-urlencoded 방식으로만 인코딩하여, GET인자로 전달
- POST 방식 : 택배에 비유. 다양한 물건을 보낼 수 있다.
 - GET인자/POST인자 가능
 - 지정된 enctype으로 인코딩하여, body에 포함시켜 처리

<form method="GET">

- enctype 지정 불가. enctype 지정을 무시하고 urlencoded된 key/value쌍을 URL뒤에 붙여, GET인자로서 전달
- 주로 검색폼에서 인자를 넘길 때, 씁니다.

간단 글쓰기 폼

```
<form method="GET" action="">
  <input type="text" name="title" />
  <textarea name="content"></textarea>
  <input type="file" name="photo" />
  <input type="submit" value="저장" />
</form>
```


요청 패킷

```
GET http://localhost:8000/blog/new/?title=title&content=content&photo=filename.jpg HTTP/1.1
Host: localhost:8000
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.10; rv:39.0) Gecko/20100101 Firefox/39.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ko-KR,ko;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
DNT: 1
Referer: http://localhost:8000/blog/new/
Cookie: csrftoken=bDaUcTBk3CxBH3FJ4X9zmQpjWD6swBG7; _ga=GA1.1.2131557215.1432514035; sessionid=fk7101xhd9nbwlncoidyxb1amzff9yom
Connection: keep-alive
```

(header 만 있고, body 부분이 없음, header/body구분은 첫 빈줄 하나로 구분)

GET인자

title=title&content=content&photo=filename.jpg

장고 View 호출 시, 인자 현황

```
request.GET : <QueryDict: {'title': ['title'], 'content': ['content'], 'photo': ['filename.jpg']}>
request.POST : <QueryDict: {}>
request.FILES : <MultiValueDict: {}>
```

<form method="POST" enctype="application/x-www-form-urlencoded">

- 디폴트 enctype : application/x-www-form-urlencoded
- urlencoded된 key/value쌍을 request BODY에 담아서 요청

간단 글쓰기 폼

```
<form method="POST" action="">
  <input type="text" name="title" />
  <textarea name="content"></textarea>
  <input type="file" name="photo" />
  <input type="submit" value="저장" />
</form>
```

요청 패킷

```
POST http://localhost:8000/diary/new/ HTTP/1.1
Host: localhost:8000
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.10; rv:39.0) Gecko/20100101 Firefox/39.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ko-KR,ko;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
DNT: 1
Referer: http://localhost:8000/diary/new/
Cookie: csrftoken=bDaUcTBk3CxBH3FJ4X9zmQpjWD6swBG7; _ga=GA1.1.2131557215.1432514035; sessionid=fk7101xhd9nbwlncoidyxb1amzff9yom
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 46
```

```
title=title&content=content&photo=filename.jpg
```

POST 인자

```
title=title&content=content&photo=filename.jpg
```

장고 View 호출 시, 인자 현황

```
request.GET : <QueryDict: {}>
request.POST : <QueryDict: {'title': ['title'], 'content': ['content'], 'photo': ['filename.jpg']}>
request.FILES : <MultiValueDict: {}>
```

<form method="POST" enctype="multipart/form-data">

- 파일 업로드 지원

간단 글쓰기 폼

```
<form method="POST" action="" enctype="multipart/form-data">
  <input type="text" name="title" />
  <textarea name="content"></textarea>
  <input type="file" name="photo" />
  <input type="submit" value="저장" />
</form>
```

```
POST http://localhost:8000/diary/new/ HTTP/1.1
Host: localhost:8000
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.10; rv:39.0) Gecko/20100101 Firefox/39.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ko-KR,ko;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
DNT: 1
Referer: http://localhost:8000/diary/new/
Cookie: csrftoken=bDaUcTBk3CxBH3FJ4X9zmQpjWD6swBG7; _ga=GA1.1.2131557215.1432514035; sessionid=fk7l01xhd9nbwlncoidyxb1amzff9yom
Connection: keep-alive
Content-Type: multipart/form-data; boundary=-----14973275531370807725960869059
Content-Length: 296
```

```
-----14973275531370807725960869059
Content-Disposition: form-data; name="title"
```

title

```
-----14973275531370807725960869059
```

```
Content-Disposition: form-data; name="content"
```

(생략 ...)

장고 View 호출 시, 인자 현황

```
request.GET : <QueryDict: {}>
request.POST : <QueryDict: {'title': ['title'], 'content': ['content']}>
request.FILES : <MultiValueDict: {'photo': [<InMemoryUploadedFile: 사진.png (image/png)>]}>
```


*Life is short,
use Python3/Django.*