

올인원 패키지 Online.

장고 세션 다루기

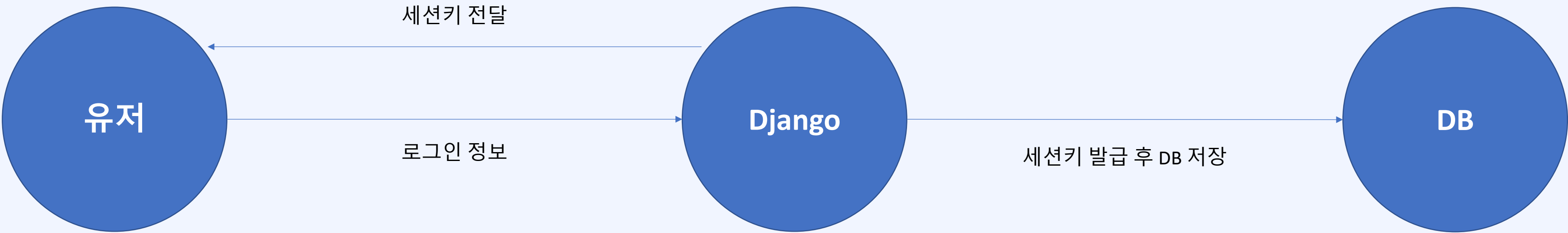
PART1 | Django Login 방식

PART2 | Django 세션 id 방식 vs JWT 방식

PART3 | Django 세션 다루기

1. Django 로그인 방식

1.
장고 세션 다루기



Enter a part of table name			
MySQL - localhost			
SQLite - db.sqlite3			
테이블			
auth_group			
auth_group_perm			
auth_permission			
django_admin_log			
django_content_type			
django_migration			
django_session			
shortener_paypla			
shortener_users			
shortener_users_			
shortener_users_			

	session_key	session_data	expire_date
1	p6o49rnsypcslo129jkgcbir	.eJxVjDEOAiEQRe9CbcgglGBp7xnlAD0yaiBZdivj3XV	2021-06-12 07:45:55.797784
2	zbb2rmy77n0ztfx5jcc6xfa	.eJxVjMsOwiAQRf-FtSG8KS7d-w1kGAapGkhKuzL-u	2021-08-09 14:17:35.737367
3	iw8bjowuirgbm88l3j9paso	.eJxVjMsOwiAQRf-FtSG8KS7d-w1kGAapGkhKuzL-u	2021-08-13 16:35:21.129020
4	5be5lk7vfroyb9dqz16fjh7	.eJxVjMsOwiAQRf-FtSFQHi1duvcbyDAMFjVgSptojP-	2021-08-13 16:35:52.181171
5	qqf2rdi6fm4vkaphimlt7pb	.eJxVjMsOwiAQRf-FtSG8KS7d-w1kGAapGkhKuzL-u	2021-08-13 16:48:28.456707
6	1ywhz7bepokb4m4yr4nn9	.eJxVjDsOgzAQBe_iOrJs_AFTps8Z0O56HUgCSDauc	2021-08-13 17:27:30.115657

2. Django 세션 방식 활성화

```
MIDDLEWARE = [  
    "django.middleware.security.SecurityMiddleware",  
    "django.contrib.sessions.middleware.SessionMiddleware",  
    "django.middleware.common.CommonMiddleware",  
    "django.middleware.csrf.CsrfViewMiddleware",  
    "django.contrib.auth.middleware.AuthenticationMiddleware",  
    "django.contrib.messages.middleware.MessageMiddleware",  
    "django.middleware.clickjacking.XFrameOptionsMiddleware",  
    "debug_toolbar.middleware.DebugToolbarMiddleware", # Django De  
]
```

settings.py

3. Django 세션 엔진 종류

SESSION_ENGINE

- 'django.contrib.sessions.backends.db'
- 'django.contrib.sessions.backends.file'
- 'django.contrib.sessions.backends.cache'
- 'django.contrib.sessions.backends.cached_db'
- 'django.contrib.sessions.backends.signed_cookies'

<https://docs.djangoproject.com/en/3.2/ref/settings/#session-engine>

4. JWT와는 뭐가 다른가?

4.

장고 세션 다루기

Encoded

PASTE A TOKEN HERE

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MzkwMjQyLCJpc29udCI6ImFkbG9uZS51c2VudCJ9
```

Decoded

EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{  "alg": "HS256",  "typ": "JWT"}
```

PAYLOAD: DATA

```
{  "sub": "1234567890",  "name": "John Doe",  "iat": 1516239022}
```

5. JWT와는 뭐가 다른가?

	JWT	장고 세션
요청마다 DB 조회	Optional	Must
정보 변경	가능	가능
저장 방식	모든 방법 가능	쿠키
보안	우수	매우 우수(HTTPS 사용)

6. 세션 다루기

```
>>> # initial assignment
>>> request.session[0] = 'bar'
>>> # subsequent requests following serialization & deserialization
>>> # of session data
>>> request.session[0] # KeyError
>>> request.session['0']
'bar'
```

SESSION_COOKIE_AGE = 1209600
SESSION_COOKIE_DOMAIN = None (example.com)
SESSION_COOKIE_HTTPONLY = True
SESSION_COOKIE_NAME = True
SESSION_COOKIE_SAMESITE = Lax
SESSION_COOKIE_SECURE = False
SESSION_SAVE_EVERY_REQUEST = False

<https://docs.djangoproject.com/en/3.2/topics/http/sessions/#using-sessions-in-views>