

# 장고 차근차근 시작하기 2/E

당신의 파이썬/장고 페이스메이커가 되겠습니다. ☺

EP-16. django-debug-toolbar를 통한 SQL 디버깅

# django-debug-toolbar

- 현재 request/response에 대한 다양한 디버깅 정보를 보여줌.
- 다양한 Panel 지원
  - SQLPanel을 통해, 각 요청 처리 시에 발생한 SQL 내역 확인 가능
  - Ajax 요청에 대한 내역은 미지원

# django-debug-toolbar 설치

- 설치 공식문서
  - <https://django-debug-toolbar.readthedocs.io/en/latest/installation.html>
- 주의사항
  - 웹사이트의 템플릿에 필히 "<body>" 태그가 있어야만, dbt가 동작
  - 이유 : dbt의 html/script 디폴트 주입 타겟이 </body> 태그

# 코드를 통한 SQL 내역 확인

- QuerySet의 query 속성 참조
  - ex) `print(Post.objects.all().query)` → 실제 문자열 참조 시에 SQL 생성
- `settings.DEBUG = True` 시에만 쿼리 실행내역을 메모리에 누적
- 쿼리 확인 `from django.db import connection, connections`

```
for row_dict in connection.queries:  
    print('{time} {sql}'.format(**row_dict))
```

```
connections['default'].queries
```

- 쿼리 초기화
  - 메모리에 누적되기에, 프로세스가 재시작되면 초기화
  - `django.db.reset_queries()` 통해서 수동 초기화도 가능

# 그 외 : django-querycount ([github](#))

- SQL 실행내역을 개발서버 콘솔 표준출력
- Ajax 내역도 출력 가능

```
[20/Nov/2015 19:52:12] "GET / HTTP/1.1" 200 417
http://127.0.0.1:8000/thing/4/


| Type | Database | Reads | Writes | Totals | Duplicates |
|------|----------|-------|--------|--------|------------|
| RESP | default  | 101   | 0      | 101    | 101        |


Total queries: 101 in 0.0433s

Repeated 101 times.
SELECT "fortests_thing"."id", "fortests_thing"."title",
"fortests_thing"."description", "fortests_thing"."day_of",
"fortests_thing"."timestamp" FROM "fortests_thing"

Repeated 1 times.
SELECT "fortests_thing"."id", "fortests_thing"."title",
"fortests_thing"."description", "fortests_thing"."day_of",
"fortests_thing"."timestamp" FROM "fortests_thing" WHERE
"fortests_thing"."id" = 4
```

인생은 짧습니다.  
파이썬/장고를 쓰세요.