

**Django Python Shell**

# Django Python Shell

- 장고는 파이썬 셸을 이용하여 데이터를 관리할 수 있는 API를 제공한다.
- 복잡한 데이터 처리 또는 별도로 웹 브라우저로 접속할 필요가 없는 경우 셸로 데이터를 처리한다.
- `python manage.py shell`
- `manage.py` 모듈에서 정의한 `DJANGO_SETTINGS_MODULE` 속성을 이용하여 미리 `mysite/settings.py` 모듈을 임포트한다.
- 장고의 ORM은 테이블을 클래스로 표현한다. 클래스의 객체를 생성하는 것은 테이블의 레코드를 생성하는 것을 의미한다.

# Create - 데이터 생성/입력

- 테이블에 레코드를 생성하기 위해서는 필드값을 지정하여 객체를 생성한 후에 `save()` 메소드를 호출한다.
- `save()` 명령이 실행되기 전까지는 메모리에서만 변경된다.
- `save()` 명령을 통해 변경사항을 데이터베이스에 반영한다.
- 내부적으로 SQL 용어의 INSERT 문장을 실행한다.

# Create - 데이터 생성/입력

```
>>> from polls.models import Question, Choice
```

```
>>> from django.utils import timezone
```

```
>>> q = Question(question_text="What's new?",  
pub_date=timezone.now())
```

```
>>> q.save()
```

# Read - 데이터 조회

- 데이터베이스로부터 데이터를 조회하기 위해서는 QuerySet 객체를 사용한다.
- QuerySet은 데이터베이스 테이블로부터 꺼내 온 객체들의 컬렉션이다.
- QuerySet은 필터를 가질 수 있으며 필터를 사용하여 QuerySet 내에서 조건에 맞는 레코드만 다시 추출한다.
- QuerySet은 SQL 용어로 SELECT 문장에 하며 필터는 WHERE 절에 해당한다.

## **NOTE** 컬렉션이란?

다수의 객체를 한 곳에 모아 동일한 방식으로 다룰 수 있도록 하는 데이터 구조

# Read - 데이터 조회

- 조회 결과를 담는 QuerySet을 얻으려면 objects 객체를 사용한다.
- objects 객체는 테이블 정보를 담고 있는 객체이다.

```
>>> Question.objects.all()
```

```
>>> Question.objects.all()[:10:2] # 실행 결과 리스트를 반환한다.
```

- 한 개의 요소만 있는 것이 확실한 경우 get() 메소드를 호출한다. 호출 결과는 QuerySet이 아닌 하나의 객체이다.

```
>>> one_entry = Question.objects.get(pk=1)
```

# Read - 데이터 조회

- 조건에 맞는 일부 레코드만 검색할 때는 filter()와 exclude() 메소드를 사용한다.
  - filter(): 주어진 조건에 맞는 객체를 담고 있는 QuerySet 반환
  - exclude(): 주어진 조건에 맞는 객체를 제외한 QuerySet 반환
- QuerySet 메소드는 실행 결과 QuerySet 컬렉션을 반환하므로 체인식 호출이 가능하다.

```
>>> Question.objects.filter(question_text_startswith='What')  
.exclude(pub_date_gte=datetime.date.today())  
.filter(pub_date_gte=datetime(2005, 1, 30))
```

# Update - 데이터 수정

- 이미 존재하는 객체에 대한 필드값을 수정하는 경우 필드 속성값을 변경한 후 `save()` 메소드를 호출한다.
- SQL 용어로 UPDATE에 해당한다.

```
>>> q.question_text = 'What is your favorite hobby?'
```

```
>>> q.save()
```

```
>>> Question.objects.filter(pub_date__year=2007  
)update(question_text='Everything is the same')
```



# Delete - 데이터 삭제

- 객체를 삭제하기 위해서는 delete() 메소드를 사용한다.
- SQL 용어로 DELETE 절에 해당한다.

```
>>> Question.objects.filter(pub_date__year).delete()
```

```
>>> Question.objects.all().delete()
```