

# Django - blog

## models.py

```
from django.db import models

class MyModel(models.Model):
    name = models.CharField(max_length=50)
    age = models.IntegerField()
    email = models.EmailField()
```

1. django.db의 models 가져오기
2. class 모델 생성
  - 테이블 속성

데이터베이스에 모델(테이블) 생성

models.py 변경 -> migrations -> migrate

## Django - blog

데이터베이스 값 생성

```
my_model = MyModel(name='a', age=10, email="aaaa@gmail.com")
```

```
my_model.save()
```

데이터베이스에 값을 저장  
변수에 클래스 할당 -> 저장(save)

# Django - blog

데이터베이스 값 조회

**MyModel.objects.all()**

**MyModel.objects.get((조건)name='a')**

데이터베이스에 값 조회

**MyModel 테이블 -> objects -> all, get**

## Django - blog

데이터베이스 값 수정

```
my_model.age = 20  
my_model.save()
```

데이터베이스에 값 수정

object(변수) -> . 찍고 속성 접근 -> 수정 -> 저장

## Django - blog

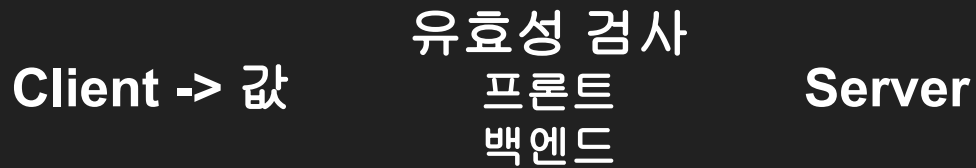
데이터베이스 값 삭제

```
my_model.delete()
```

데이터베이스에 값 삭제

`object(변수) -> .` 찍고 메서드 삭제

# 유효성 검사



## URL 구성

<http://www.domain.co.kr/blog>

<http://www.domain.co.kr/blog/detail/1>

<http://www.domain.co.kr/blog/detail/2>

<http://www.domain.co.kr/blog/detail/3>

Detail 페이지



글 수정  
페이지

두 기능의 원리는 같지만 화면을 다르게 구성하면 구현 가능

## 각 객체(Object)의 고유값

- 객체 -> id: post.pk
- url: detail/<int:pk>/
- html: href="{% url 'blog:detail' pk=post.pk %}"