

## model

DB の構造や操作手順を記述するファイル。  
クラス変数としての Field クラスは DB テーブルのフィールドを表す。  
フィールドの変更を行った後は `manage.py makemigrations` で `migrate` ファイルを作成し、`manage.py migrate` で変更を適用する必要がある。  
Meta クラスを定義することでテーブル名やデフォルトのソートオーダなどを指定可能。

Listing 1 models.py 記述例

```
1 from django.db import models
2
3
4 class Question(models.Model):
5     class Meta:
6         verbose_name = '質問',
7         verbose_name_plural = '質問の複数形',
8
9         question_text = models.CharField(max_length=200)
```

## フィールド例

| フィールド名            | 内容         | 補足                        |
|-------------------|------------|---------------------------|
| CharField         | 文字列        | 文字数の指定が必要                 |
| TextField         | 文字列        | こっちは何文字で OK               |
| IntegerField      | 数字         |                           |
| SmallIntegerField | 数字         | ステータス等の桁数の短い数字用           |
| BooleanField      | Boolean    | フラグなど                     |
| NullBooleanField  | Boolean    | None(選択なし) も許す Boolean    |
| .DateField        | 日付         | 出版日や発売日など時間情報がいない場合       |
| DateTimeField     | 日時         | 作成日など時間まで取りたい場合           |
| ForeignKey        | 数字 (pk)    | 1 対多のリレーションに使用            |
| OneToOneField     | 数字 (pk)    | 1 対 1 のリレーションに使用          |
| ManyToManyField   | -          | 多対多のリレーション。中間テーブルが自動生成される |
| FileField         | 文字列 (path) | ファイルを扱うためのフィールド           |
| ImageField        | 文字列 (path) | 画像ファイルを扱うためのフィールド         |

## フィールドの引数

| 引数名          | 設定値                       | 補足                         |
|--------------|---------------------------|----------------------------|
| verbose_name | 文字列                       | 管理画面等に自動表示される名前            |
| max_length   | 数字                        | 最大長。CharField などで文字数の指定に使用 |
| null         | True/False                | DB で null を許すかどうか          |
| blank        | True/False                | Form で空の入力を許すかどうか          |
| help_text    | 文字列                       | Form 等で使用する補足文字            |
| default      | 任意の値                      | DB のデフォルト値 / Form の初期値     |
| choices      | (key, value) の tuple のリスト | 選択肢                        |
| editable     | True/False                | 管理画面で編集可能かどうか              |
| auto_now     | True/False                | 自動で現在時刻を挿入する               |
| auto_now_add | True/False                | 新規作成時のみ自動で現在時刻を挿入する        |

## クエリセットと instance

モデルクラスは `objects` という名前で Manager クラスを持っており、Manager クラスを通して DB 上から必要な行を取得する。

Manager クラスからはクエリセットを得ることもできる。クエリセットは iterable であり、インスタンスを返す。  
`instance` は DB テーブルのレコードを示しており、クラス定義で行った Field と同名のメンバ変数を持ち、そのフィールド値を取得できる。

Listing 2 queryset の取得と instance の取得

```
1 qs = Question.objects.all()
2 for obj in qs:
3     print(obj.question_text)
```