

④

マイグレーション

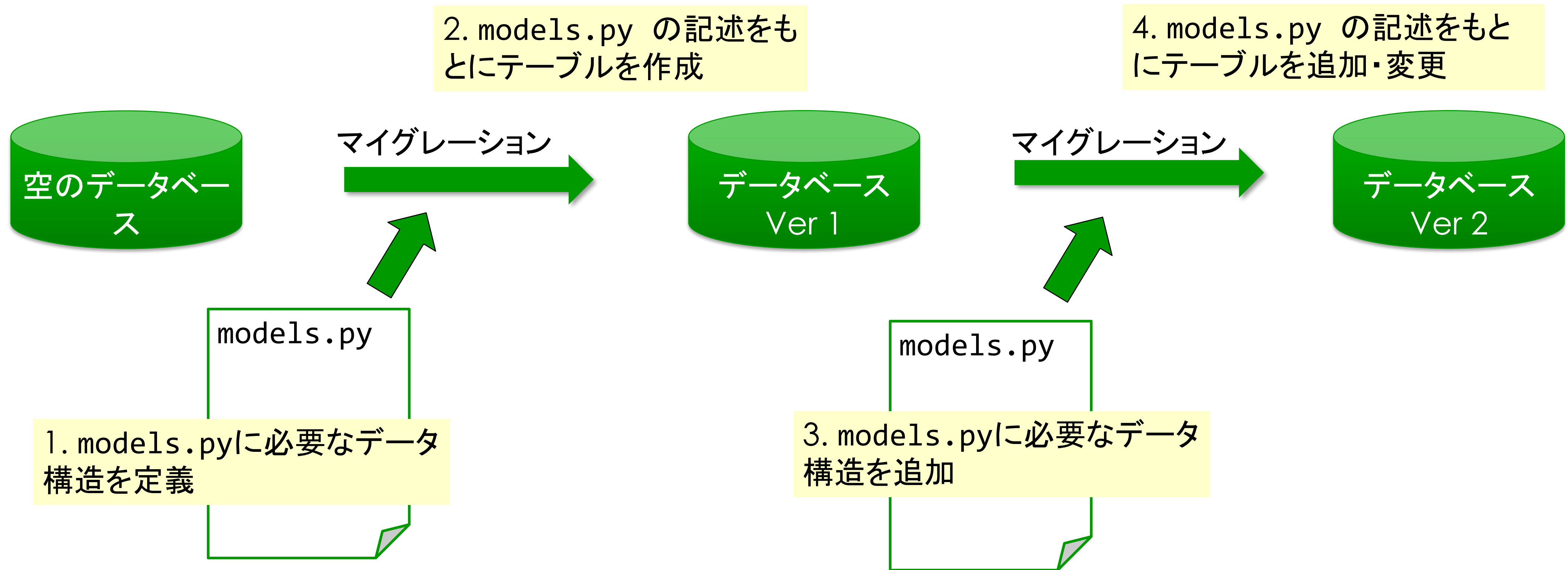
Modelで定義した内容をデータベースのテーブルに反映する操作であるマイグレーションについて学習します。

マイグレーションとは

- 「移行」という意味合い
- Djangoでは、Webアプリケーションの機能をバージョンアップする際に、データベースのテーブルを更新する仕組みが提供されており、これを「マイグレーション」と呼ぶ
- 具体的には、以下のような機能が提供されている
 1. プログラマーは、まずModelのファイルを変更する
 2. 変更の終了後、コマンドを実行することで、Modelの変更に応じた、データベースの更新が行われる

Djangoにおけるマイグレーション

- Djangoでは、機能拡張の度に`models.py`に必要なデータ構造を追記し、マイグレーションによりデータベースを更新する



マイグレーションの手順

- まず、Windows Power Shell上でプロジェクトのフォルダに移動し、仮想環境を有効化します
 - Djangoサーバを動かしている場合は、Ctrl+Cで停止して、そのウィンドウで実行すれば大丈夫です
- 以下の手順でマイグレーションを実行する
 1. マイグレーションを作成する

```
python manage.py makemigrations
```
 2. マイグレーションを実行する

```
python manage.py migrate
```

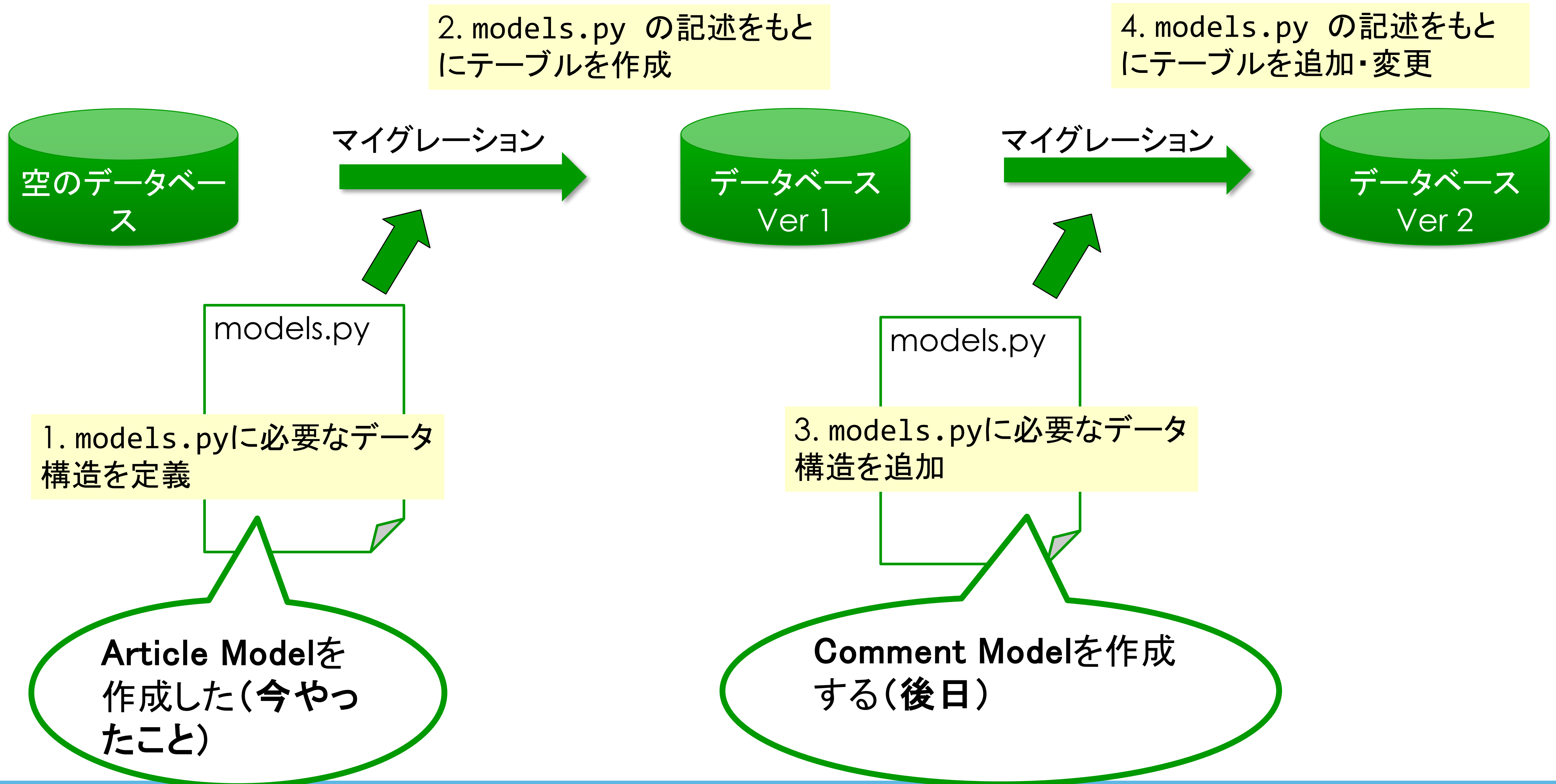
マイグレーションの手順

```
(venv) C:\Users\iniad\Documents\cs2020_web\blog_sample>python manage.py makemigrations
Migrations for 'blog':
  blog\migrations\0001_initial.py
    - Create model Article

(venv) C:\Users\iniad\Documents\cs2020_web\blog_sample>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, blog, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  ...(中略)...
  Applying sessions.0001_initial... OK

(venv) C:\Users\iniad\Documents\cs2020_web\blog_sample>
```

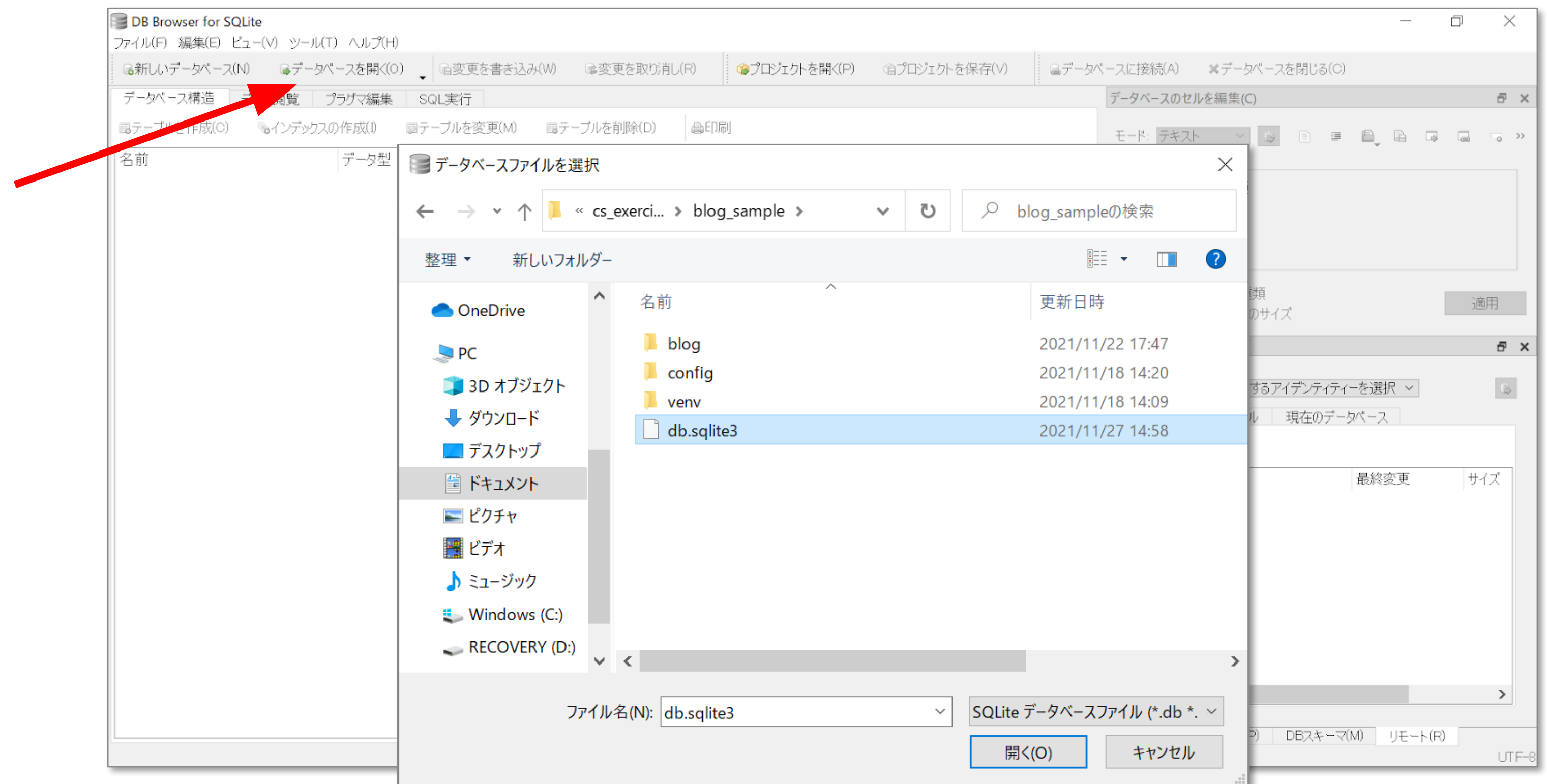
Djangoにおけるマイグレーション



DB Browserでテーブルができていることを確認しよう (1/2)

- DBBrowser for SQLite で db.sqlite3を開きます
 - ※Djangoでは、開発用にSQLiteのデータベースが生成されます

「データベースを開く」から
blog_sample直下の
db.sqlite3を指定し、
「開く」ボタンを押す。



DB Browserでテーブルができていることを確認しよう (2/2)

- Database Structureタブから、右図のようにDBのカラム定義が確認できます
- テーブル blog_article ができていることを確認しよう
- テーブル blog_article には、以下が格納
 - id : 記事ID
 - title : 記事のタイトル
 - body : 記事の本文
 - posted_at : 記事の投稿日時
 - published_at : 記事の公開日時
 - like : イイね！の数

