|  |  |
| --- | --- |
| python manage.py runserver | サーバー起動 |
| ctrt + C | サーバー停止 |
| py -3.8 -m venv 仮想環境名 | python3.8の仮想環境を作成する |
| 仮想環境名\Scripts\activate | 仮想環境起動 |
| deactivate | 仮想環境終了 |
| python manage.py test | testファイルに書いたユニットテストコードを実行する際に使用 |
| python manage.py createsuperuser | superuser(システム管理者)のレコードを作成するためのコマンド。実行後、username、メールアドレス、パスワードが求められる |
| python manage.py makemigrations | models.pyに記入した内容を反映し、マイグレーションファイルを自動生成するためのコマンド |
| python manage.py migrate | マイグレーションファイルに作成された内容を基にデータベースのテーブルの作成と変更をします |
| python manage.py makemigrations アプリ名 | Djangoにモデルに変更があったこと(この場合、新しいものを作成しました)を伝え、そして変更をマイグレーションの形で保存する |
| django-admin startproject プロジェクト名 | Djangoのプロジェクトを構成するコードの自動生成 |
| python manage.py startapp アプリ名 | プロジェクトに対してアプリケーション(MVCの説明：オンラインショップに該当)を作成 |
| python manage.py shell | pythonの対話型シェルを起動 |
| exit() | シェル終了 |
| python -m django --version | djangoがインストールされていればdjangoのバージョンを返答 |

マイグレーションは Django がモデル (そしてデータベーススキーマでもあります) の変更を保存する方法です。マイグレーションは、ディスク上のただのファイルです。

＊補足

python manage.py test　クラス単位での実行方法など細かく指定できる

<https://docs.djangoproject.com/ja/3.1/intro/tutorial05/>

また何かよく使いそうだなと感じましたら、追加致します。