

Seeds

FIRST STEP

-5 week-

INDEX

01	Django 소개
02	Django 핵심 기능
03	Django의 특징
04	Django의 장점 및 단점
05	Django 프로젝트 생성

01 Django 소개

- Django는 파이썬으로 제작된 오픈소스 웹 프레임워크이다.
- 비교적 입문이 쉬운 언어인 파이썬으로 제작되어 진입장벽이 그렇게 높은 편이 아니고 특유의 높은 생산성으로 특히 스타트업에게 사랑받는 프레임워크
- 광고로 제작된 대표적인 사이트: 인스타그램, 핀터레스트 등



01 Django의 핵심 기능

- 데이터베이스 관리
 - 데이터베이스(Database), 줄여서 DB라고도 하는 용어를 들어보았을 것이다. 컴퓨터에서 정보들을 저장하고 불러오며, 저장된 정보들을 찾기 위한 저장소이다.
- 강력한 관리자 기능
 - Django는 처음부터 이 관리자 기능에 큰 중점을 두고 만들어 졌다. 실제로 전혀 노력하지 않아도 꽤 쓸만한 수준의 데이터베이스 조회/추가/수정/삭제 기능을 제공하며, 약간의 수정만으로도 실제 운영되는 서비스에서도 사용할만한 관리자 페이지를 자동으로 생성해준다.
- 보안
 - django는 다양한 보안 장치를 기본적으로 탑재하고 있으며, 숙련자의 경우 자신이 원하는 대로 보안 요소들을 쉽게 추가할 수 있게 설계되어 있다. 실제로 django가 제공하는 기능을 굳이 비활성화하지 않는다면, django가 관리하는 영역의 보안은 크게 신경쓰지 않아도 될 정도이다.

01 Django의 특징

- MVT 패턴
 - django는 MVC(Model - View - Controller)을 기반으로 한 프레임워크, 하지만 django에서는 View를 Template, Controller를 View라고 부른다. django에서 View는 데이터를 가져오고 변형하는 컴포넌트인 반면에 Template은 데이터를 사용자에게 보여주는 컴포넌트이다. 그래서 django를 흔히 MVT(Mode - View - Template)프레임워크라고 부름.
- 자동으로 구성되는 관리자 화면
 - django는 웹서비스 콘텐츠 즉 데이터베이스에 대한 관리기능을 위하여 프로젝트를 시작하는 시점에 기본기능으로 관리자 화면을 제공.
- 다국어 지원
 - django는 동일한 소스코드를 다른나라에게도 사용 가능하도록 텍스트의 번역, 날짜/시간/숫자의 포맷, 타임존의 지정과 같은 다국어 환경을 제공.

01 Django의 특징

- 강력한 ORM기능을 지원
 - ORM(Object Relational Mapping, 객체 관계 매핑)은 객체와 관계형 DB를 연결해주는 개념을 의미한다. 쉽게 설명하면, 생성한 클래스(객체)를 SQL문으로 자동변환하여 데이터를 다루는 것. 이 기능 덕분에 SQL문을 사용하지 않고도 DB데이터를 관리할 수 있다.
- 자체적인 템플릿을 지원
 - Django Template을 지원하며, 템플릿 태그를 사용해 HTML 템플릿에 context로 전송된 데이터를 사용할 수 있음. 이는 동적인 페이지를 구현하는데 도움을 준다.
- 소스코드의 변경 사항을 자동으로 반영
 - django는 자동으로 .py파일의 변동을 감시한다. 변경이 감지될 경우 변경 내역을 자동으로 반영해준다. 따라서 코드를 변경할 때 마다 수동으로 웹서버를 재시작 할 필요가 없다.

01 Django와 Flask의 비교

- 둘의 범주에서 차이가 있다.
 - django는 Full-stack Framework 이며 , Flask는 MicroFramework이다.
 - Flask는 가벼운 프레임워크이기 때문에, 개발자가 거의 모든 것을 설정하고 세팅해야 하는 번거러움이 있지만, 그만큼 내가 원하는 기능을 편하게 확장할 수 있으면, 보다 유연하다.
 - 반면에 django는 flask의 10배나 무거운 프레임워크로 거의 모든 기능을 다 지원하지만, 그만큼 자유도가 낮다.
- 프로젝트 레이아웃이 다르다.
 - Flask는 프로젝트마다 1개의 어플리케이션을 개발하도록 되어 있지만,
 - django는 한 프로젝트 내에 다양한 어플리케이션이 존재할 수 있다.
- DB접근에 차이가 있다.
 - Flask는 DB ORM기능이 제공되지 않기 때문에, SQLAlchemy와 같은 ORM 지원 패키지를 사용하는 반면, Django는 RDBMS와의 상호작용을 완전히 지원하는 기본 내장되어 있는 강력한 ORM이 제공된다.

01 Django의 장점

- 파이썬 기반 웹 프레임워크다.
 - 비교적 익히기 쉬운 파이썬을 기반으로 익히기 쉬우며 파이썬에서 가능한 모든 동작이 가능하며, 강력한 라이브러리들이 많다.
- 반복적으로 구현해야 하는 부분들이 이미 완성되어 있다.
 - 로그인, 회원가입 인증 등 라이브러리를 통해 간단히 구현이 가능하다.
- 개발시간이 단축된다.
 - 스크립트 언어이므로 비교적 쉽고 간결하며 그 결과 개발시간의 단축으로 이어진다.

01 Django의 장점

- URL 디자인이 매우 편리하다.(정규표현식 사용 가능, 함수와 URL을 1:1 매칭이 가능하다)
- 파이썬 기반의 머신러닝, 데이터 분석 라이브러리와 융합하여 사용하기 편하다.
- 출시된지 오랜 시간이 흘렀고, Instagram처럼 성공적으로 도입된 사례가 많은 만큼 안정성과 기능이 충분히 검증됨.
- 커뮤니티 활동이 활발하고, 관련 문서(자료)가 많다.

01 Django의 단점

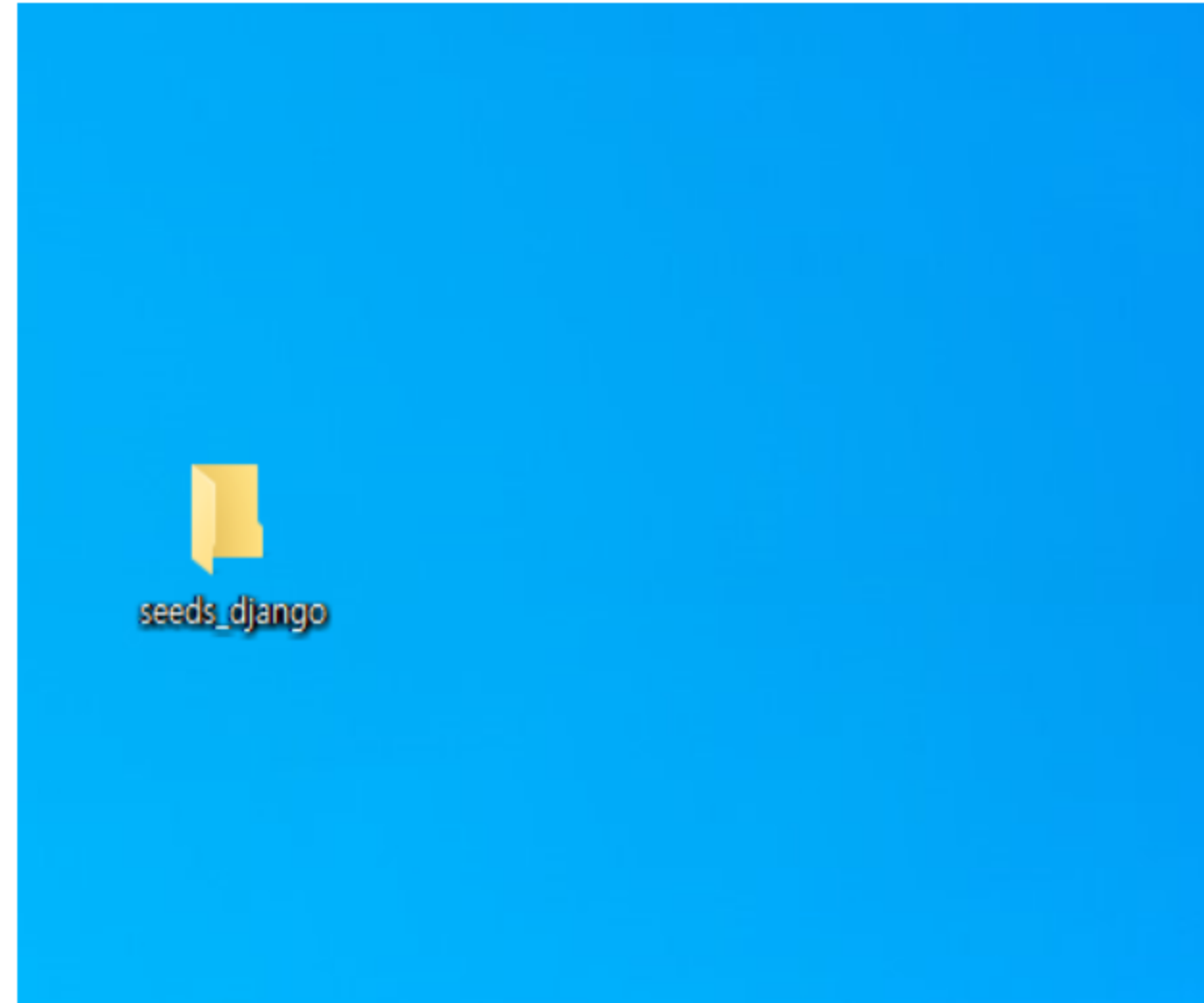
- 파이썬이 가지는 단점을 그대로 가져온다.
 - 인터프리터 언어이므로 생기는 문제점이 존재한다.
- 한글화된 자료가 부족하다.
 - 한국에 비해 외국에서 많이 사용되게 대부분 영어로 된 자료가 많다.
- 이미 만들어진 기능을 사용하는 경우가 많기 때문에 원하대로 커스텀을 하기가 힘들다.
 - 이미 라이브러리를 사용하고 있는 상황에서 수정을 해야하는 경우가 많아 해석해야한다.

01 Django의 단점

- 다른 웹 프레임워크에 비해 비교적 속도가 느리다.
- 숙련되기 위해선 django 자체에 대한 이해도 뿐만 아니라 파이썬에 대해서도 높은 이해도를 요구한다.

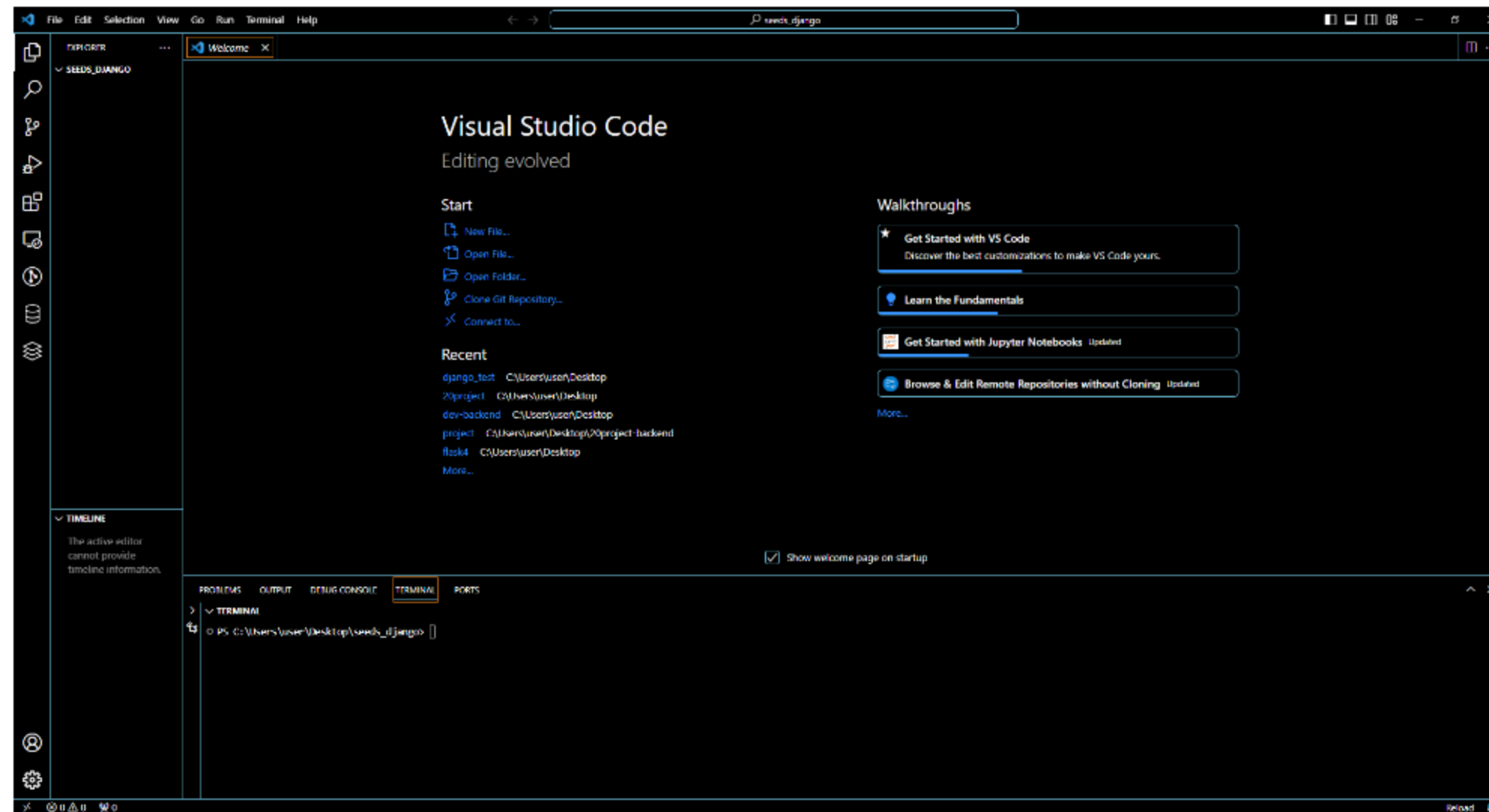
01 Django 프로젝트 생성

- 1. 바탕화면 폴더 생성



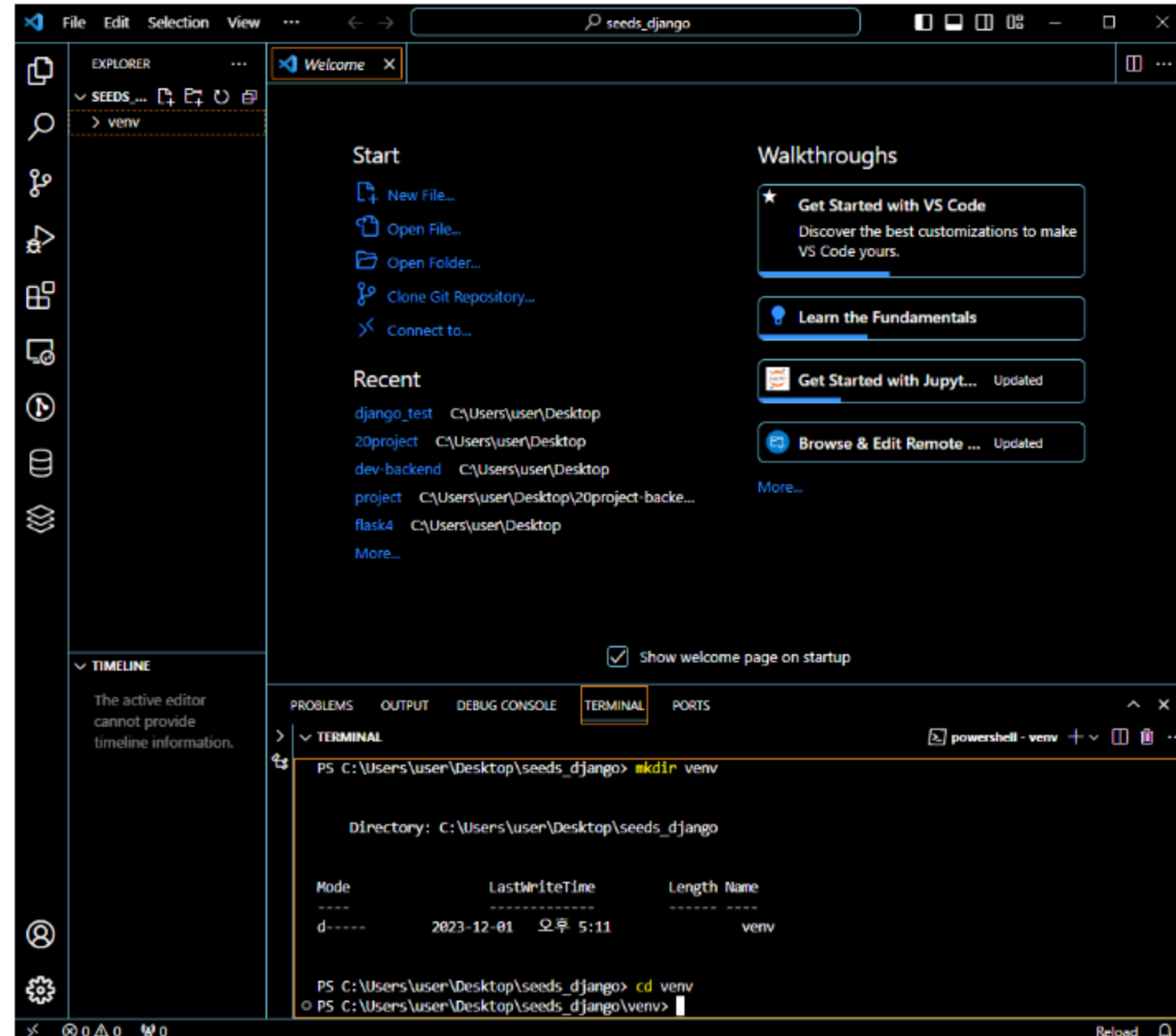
02 Django 프로젝트 생성

- 2. Visual Studio Code로 폴더 열기



02 Django 프로젝트 생성

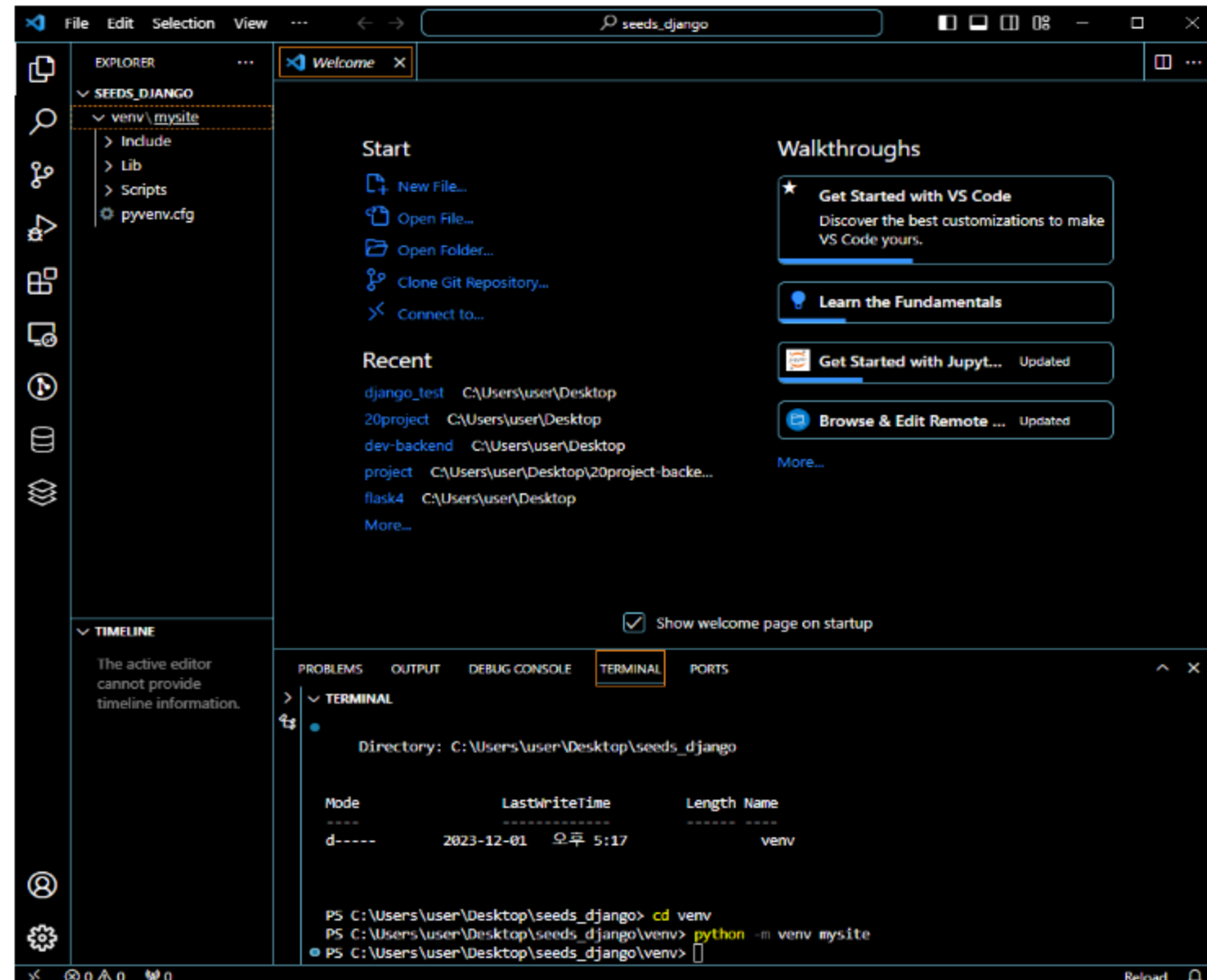
- 3. 가상환경 폴더 생성
 - 터미널 : `mkdir venv`



- 현재 경로: `C:\Users\user\Desktop\seeds_django`

02 Django 프로젝트 생성

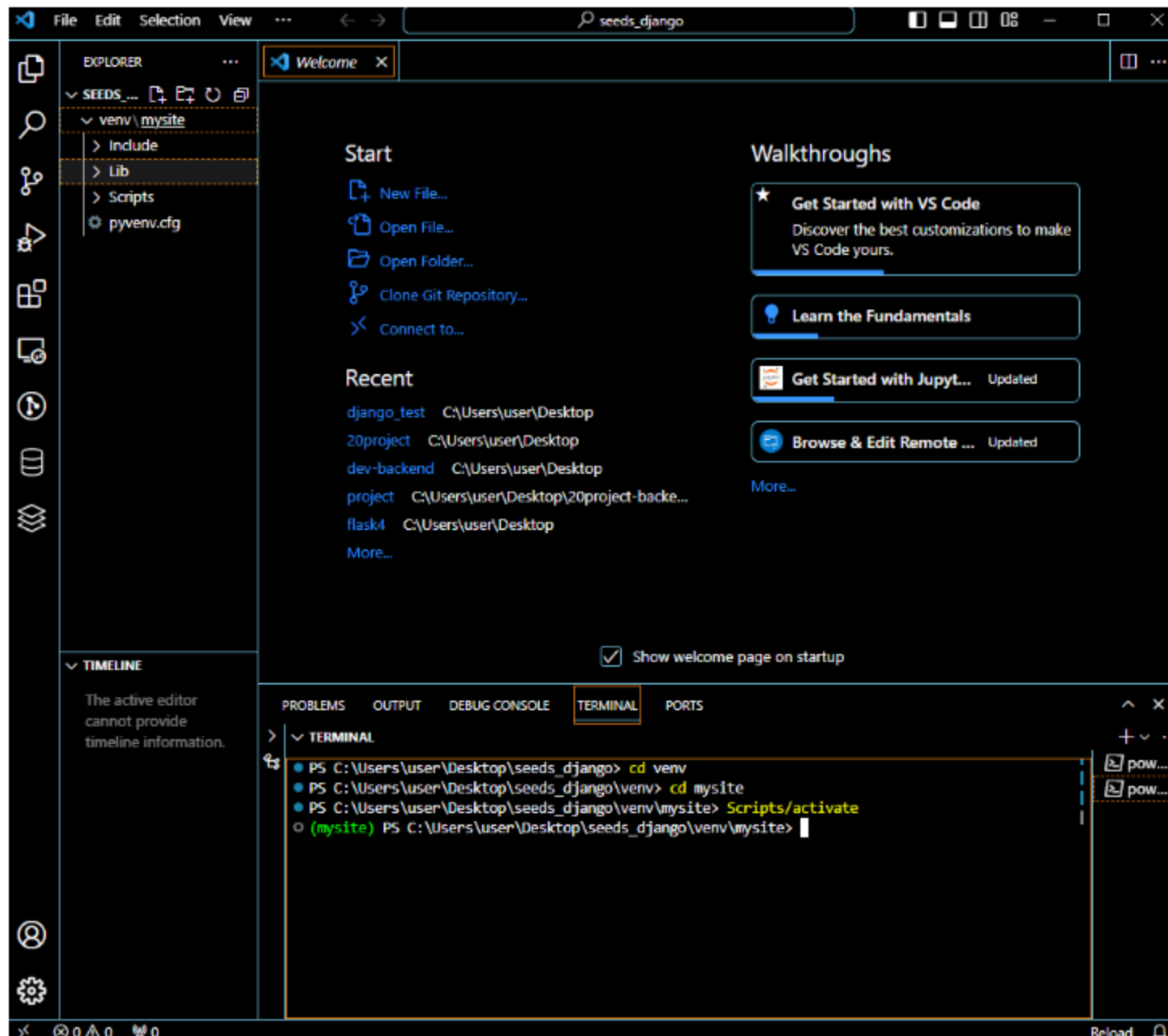
- 4. 가상환경 생성
 - 터미널 : `python -m venv mysite`



- 현재 경로: `C:\Users\user\Desktop\seeds_django\venv`

02 Django 프로젝트 생성

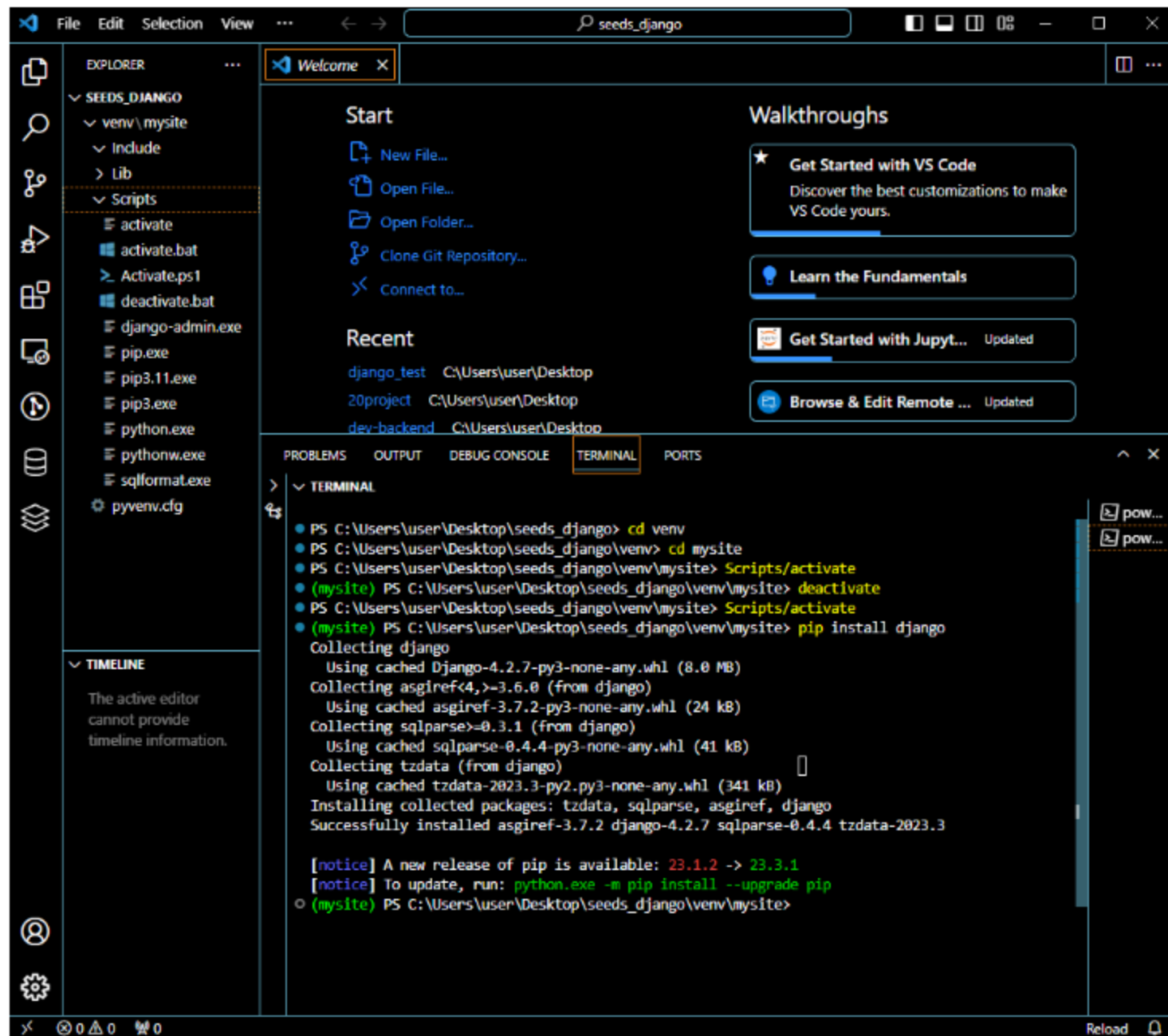
- 5. 가상환경 진입
 - 터미널 : **Scripts/activate**



- 현재 경로: C:\Users\user\Desktop\seeds_django\venv\mysite
- *가상환경 나가는 명령어 : deactivate

02 Django 프로젝트 생성

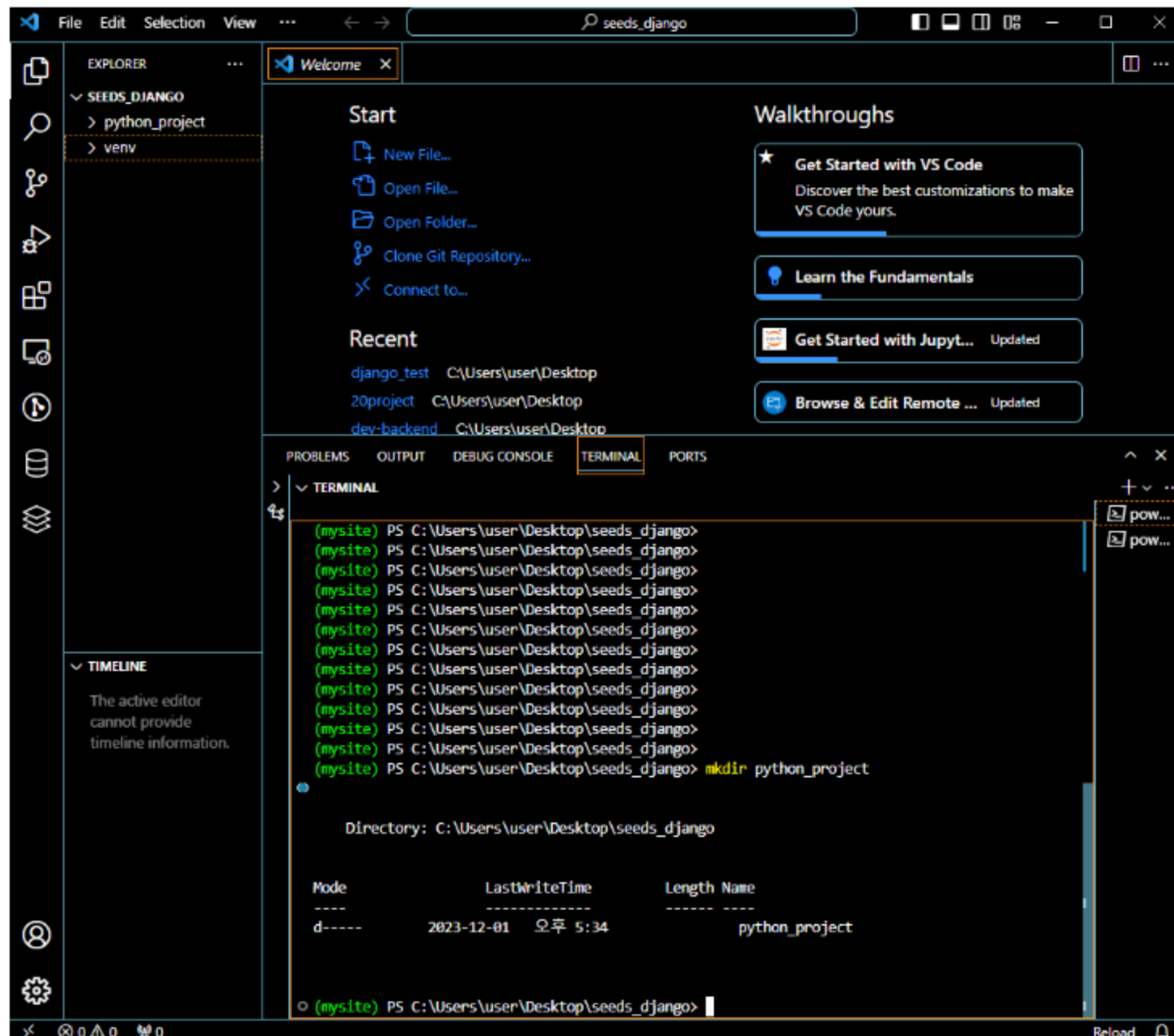
- 6. Django 설치
 - 터미널 : **pip install django**



- 현재 경로: C:\Users\user\Desktop\seeds_django\venv\mysite
- You should consider upgrading via the 'python -m pip install --upgrade pip' command.
-> 이런 메시지가 나온다면 pip를 업그레이드 시켜 줄 것.

02 Django 프로젝트 생성

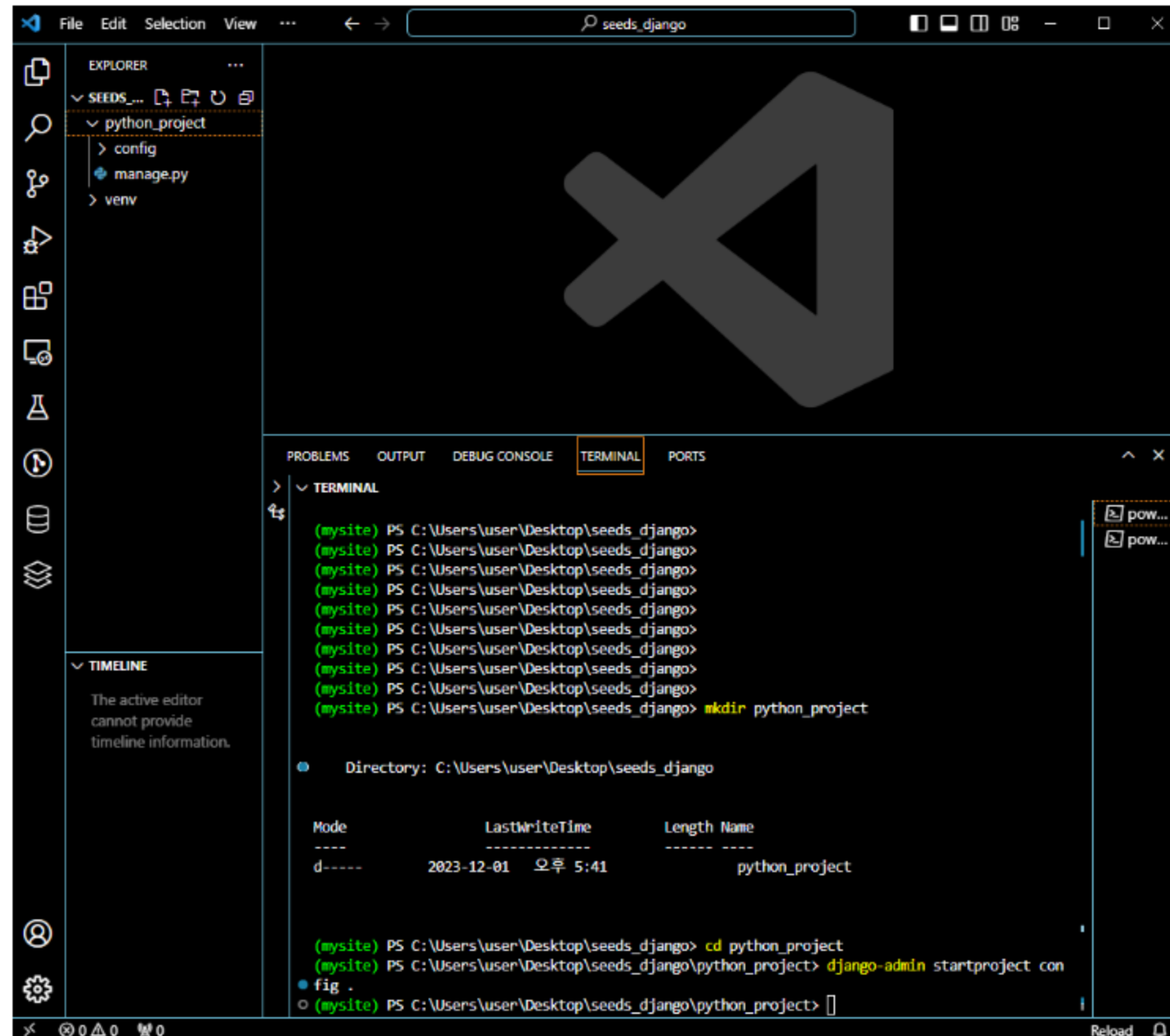
- 7. 가상환경 밖 폴더 생성
 - 터미널 : `mkdir python_project`



- 현재 경로: `C:\Users\user\Desktop\seeds_django`

02 Django 프로젝트 생성

- 8. django 프로젝트 생성
 - 터미널 : `django-admin startproject config .`

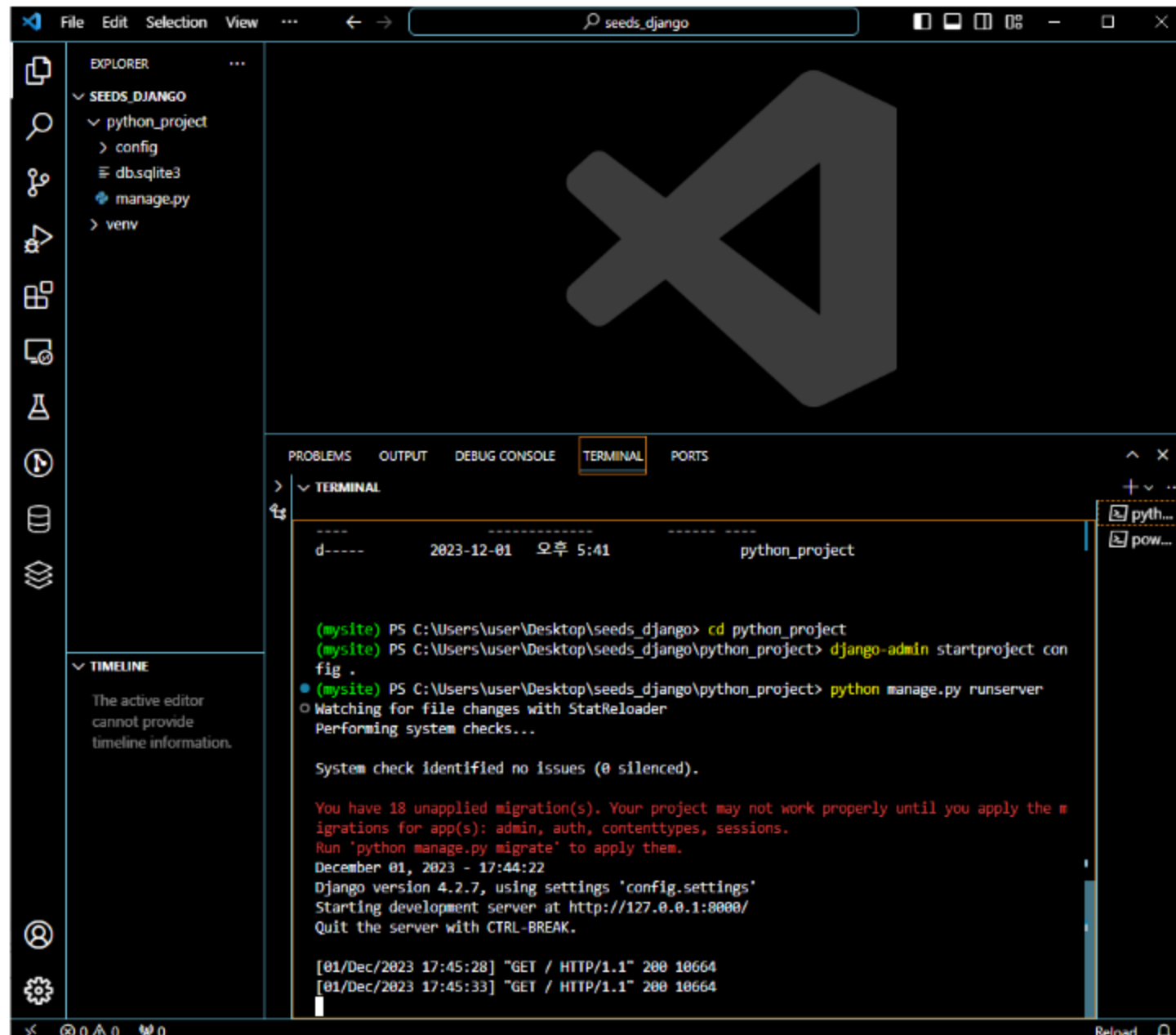


- 현재 경로: C:\Users\user\Desktop\seeds_django\python_project

02 Django 프로젝트 생성

- 9. 개발 서버 구동
 - 터미널 : `python manage.py runserver`

- 현재 경로: `C:\Users\user\Desktop\seeds_django\python_project`



과제

django 기본구조에 대해서 공부해 오세요.