1. 프로젝트에 앱 추가

생성: Django-admin startapp appname

url 매핑 등록:

Django.urls의 include기능 활용, path(‘pybo/’, include(‘pybo.urls’)) pybo.urls의 내용에 pybo/로 시작하는 url모두 이동

app디렉토리에 urls.py 만들고

from django.urls import path

from . import views

urlpatterns = [

path('', views.index),

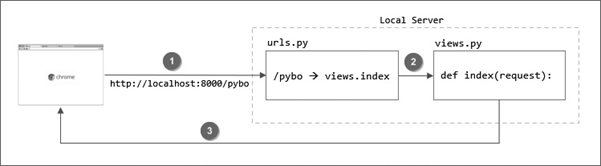
]

형식으로 내용 추가

그 안에 있는 함수(index등)를 path이름과 같이 등록

path(‘pathname/’, views.functionname)

views.py에 해당 경로에서 실행될 함수 정의



1. 각 app의 models.py

[docs.djangoproject.com/en/3.0/ref/models/fields/#field-types](https://docs.djangoproject.com/en/3.0/ref/models/fields/#field-types)

app/models.py에 객체 만들기 -> 데이터베이스에 저장될 형식

데이터베이스 쓰기 전에 config/settings.py에 앱 자체 등록 필요 (app/apps.py)

INSTALLED APPS리스트에 'pybo.apps.PyboConfig' 내용 추가

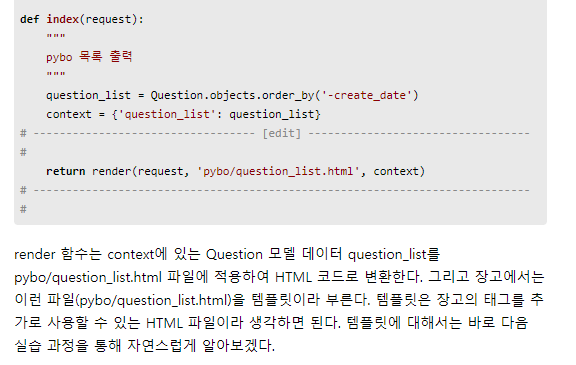
모델 정의, 앱 등록 끝났으면 makemigrations, migrate실행

Foreignkey 설정하면 설정받는 객체에 클래스이름\_set 속성이 생겨서 연결된 데이터베이스 내용조회 가능

1. 각 app의 admin.py

Admin.site.register(modelname)으로 생성 관리할 모델 입력가능

1. Views.py에 정의한 함수 (특정 path에 연결된 함수)의 기능 + html 템플릿



1. 템플릿 디렉터리

템플릿 디렉터리 생성 후 settings.py의 TEMPLATES의 ‘DIRS’에 등록

mysite/templates/pybo와 같은 형태로 각 앱의 템플릿을 하나의 디렉터리 안에서 분류해서 관리



{% if question\_list %}

<ul>

{% for question in question\_list %}

<li><a href="/pybo/{{ question.id }}/">{{ question.subject }}</a></li>

{% endfor %}

</ul>

{% else %}

<p>질문이 없습니다.</p>

{% endif %}

이 템플릿 추가하면 리스트 클릭 기능이 (a herf = url)클릭해보면 pybo/questiondata\_id 경로로 이동함 따라서 해당 경로에 연결된 함수 정의로 질문 상세내용을 보여주는 것 만들기

(views.detail 정의)

1. url namespace…

path 에 name = 함수이름 인수 추가해서 템플릿 파일에 name 활용가능

1. 답변 등록

Detail에 답변 등록 기능 추가

<**form** action="{% url 'pybo:answer\_create' question.id %}" method="post"> {% csrf\_token %} <**textarea** name="content" id="content" rows="15"></**textarea**> <**input** type="submit" value="답변등록"> </**form**>

Question\_detail.html에 템플릿 내용 추가하고 views.py에

Answer\_create함수 추가하고 같은 링크 안에서 리디렉션(버튼 누른 후에 갈 url) 리턴해서 활용

1. 표준 템플릿 상속

Base.html 쓰고 block영역 만들기

1. 등록

Index 관련 html에 버튼 기능 끝에 추가

Views.py에 등록 함수 정의하고 url 매핑 추가

Model\_form.html에서

App/forms.py에 폼 정의

Question\_form.html에 질문 등록 페이지 만들고 위에서 정의한 함수와 같은 url매핑

New Data request이면 페이지 리디렉션 반환, 저장하기 request(POST)이면 질문 데이터베이스에 등록하고 index페이지로 리디렉션