Installation/ Création des app

1. Etape 1 : Créer un projet dans pycharm
2. Etape 2 : pip install django dans le terminal
3. Etape 3 : django-admin startproject <nom du projet>
4. Etape 4 : django-admin startapp <nom de app>

Paramétrage app

1. Ajout de l’app dans setting.py => MY\_INSTALLED\_APP = [<nom app>.apps.<nom app>Confg (référence à la class config se trouvant dans le répertoire app de l’application
2. Création du modèle : création des tables en dérivant la class models.Model

Exemple :

class Promoteur(models.Model):  
 nom = models.CharField(max\_length=40)  
 description = models.CharField(max\_length=1000)

1. Installation des app

Saisir dans le terminal la commande suivante

python manage.py migrate

1. Migration du modèle

Saisir dans le terminal la commande suivante :

python manage.py makemigrations <nom de l’app>

1. Installation du modèle

Saisir dans le terminal la commande suivante :

python manage.py migrate

1. Rajout des modèles dans l’espace d’administration

Dans le répertoire de l’application, ouvrir le fichier admin.

Importer les modèles que l’on désire rendre configurable dans l’espace d’administration

from .models import <nom du modèle>

Pour une configuration par defaut renseigner la ligne suivante :

admin.site.register(<nom du modèle>)

from django.contrib import admin  
from .models import Promoteur  
  
admin.site.register(Promoteur)

1. Création d’un superuser pour accéder à l’espace admin

Dans le terminal taper : python manage.py createsuperuser

Renseigner les éléments attendus.

1. Ajout de l’url de l’application

Afin de pouvoir construire puis consulter les urls d’une application, il est nécessaire de configurer le fichier urls se trouvant à la racine du projet.

Il faut importer la bibliothèque include : from django.urls import include

Puis modifier urlpatterns :

urlpatterns = [  
 path('promoteur/', include('monimmo.urls')),  
 path('admin/', admin.site.urls),  
]

Ici l’url promoteur renvoie aux urls de l’application monimmo.

L’application lira les urls de l’application monimmo lorsque l’url débutera par « promoteur ».

1. Personnalisation de l’espace d’administration

Créer une class dérivant de admin.ModelAdmin

Saisir la valeur des paramètres.

Ajouter la class dans admin.site.register()

Exemple :

from django.contrib import admin  
from .models import Promoteur  
  
class PromoteurAdmin(admin.ModelAdmin):  
 list\_filter = ('nom',)  
  
admin.site.register(Promoteur, PromoteurAdmin)

1. Ajout des templates

Les templates sont des modèles de page html qui peuvent être réutilisables.

Dans l’emplacement d’installation créer un répertoire templates/<nom de l’app>/<nom template>.html

La structure ne doit pas être différente de celle proposée à l’exception qu’il est possible de rajouter des niveaux de profondeur.

Il est possible d’ajouter des variables qui seront renseignées dans les vues. Les variables sont définies comme ceci : {{<nom de variable>}}

Il est possible d’ajouter des filtres aux variables : {{<nom variable>|filter}}

1. Création des vues

Importer la bibliothèque render

from django.shortcuts import render

L’objet render est utile, car il permet d’exploiter les templates et de passer des arguments.

Il convient également d’importer les class des modèles que l’on veut passer aux vues.

Les vues se présentent sous forme de méthode ou de classe. Il existe des classes par défaut.

Exemple :

from django.shortcuts import render  
from django.http import HttpResponse  
from .models import Promoteur  
  
def detail(request, prom\_name):  
 output = Promoteur.objects.get(pk=prom\_name)  
 print(output)  
 return render(request, 'monimmo/base.html', {'Constructeur': output})

La vue suivante va récupérer le paramètre prom\_name (récupéré dans l’url) et l’utiliser pour récupérer le promoteur.

Le résultat est passé dans le template base.html, et la variable Constructeur est remplacée par le résultat de la requête output.

1. Création des urls de l’application

Les urls permettent d’appeler des vues et d’y passer d’éventuels paramètres.

from django.urls import path  
from . import views  
  
  
urlpatterns = [  
 path('<int:prom\_name>/', views.detail, name='post\_list'),  
]

l’url ci-dessus va charger la vue détail, et récupérer la valeur numérique du début de l’url et la passer à la variable prom\_name de la vue detail.