Django list view

Django list view¶

- List view is used to show the list data to the user.
- For example: Show list of contacts to the user.

django contact Model¶

my_app/models.py

```
from django.db import models

class Contact(models.Model):
    first_name = models.CharField(max_length=30)
    last_name = models.CharField(max_length=30)
    email = models.EmailField(max_length=255)
    phone = models.CharField(max_length=10, null=True)

    class Meta:
        db_table = "contact"
```

django list view template¶

tempaltes/list_template.html

- Below template recieves the `contacts` from the context and renders the info.

```
<h1>Contacts</h1>

{% for contact in contacts %}
<div>
    <p><strong>Name:</strong> {{ contact.first_name }} </p>
    <p><strong>Email:</strong> {{ contact.email }} </p>
</div>
{% endfor %}
```

function based list view¶

- It's easy to write and use.
- If we have complex functionality then go for generic `ListView`

- Let's look at the code

my_app/views.py

```
from django.shortcuts import render
from my_app.models import Contact

def fbv_list_view(request):
    contacts = Contact.objects.all()
    context = {'contacts': contacts}
    return render(request, 'list_template.html', context)
```

class based list view¶

- Django provides generic `ListView` for list functionality.
- We can modify it as needed.
- Let's look at the code below.

```
from django.views.generic.list import ListView
from my_app.models import Contact

class CBVListView(ListView):
    template_name = 'list_template.html'
    context_object_name = 'contacts'
    queryset = Contact.objects.all()
```

- It either requires attribute `queryset` or method `get_queryset` to return the query results.
- We can have pagination by using attribute `paginate_by`
- In template we can access the query results either with `context_object_name` value or with name `object_list`

configure the urls¶

- open `my_app/urls.py` and add below code to it.

```
from django.urls import path
from my_app import views

urlpatterns = [
    # ...
    path('fbv_list_view', views.fbv_list_view, name='fbv_list_view'),
    path('cbv_list_view', views.CBVListView.as_view(), name='cbv_list_view'),
    # ...
]
```

- Access the url http://localhost:8000/my_app/fbv_list_view to see the function based list view
- Access the url http://localhost:8000/my_app/cbv_list_view to see the class based list view

References¶

- https://docs.djangoproject.com/en/dev/ref/class-based-views/generic-display/#listview
- https://ccbv.co.uk/projects/Django/4.1/django.views.generic.list/ListView/