

```

from django.conf.urls import url
url(r'^$',views.index,name="index"), //for blank url
url(r'^index/$',views.index,name="index"),
Or
path("",views.Findex, name="index"), //for blank url
path('index/',views.Findex, name="index"),

makemigrations
migrate

@csrf_exempt

—>
STATIC_URL = '/static/'
STATICFILES_DIRS=os.path.join(BASE_DIR, 'static/'),
<—

—>
<link rel="stylesheet" type="text/css" href="/static/css1.css" />
<script src="/static/LoginValidation.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.2/jquery.min.js"></script>

<<—————

—-Css other example (socialmedia)————>

<!doctype html>
{% load static %}

<link rel="stylesheet" type="text/css" href="{% static 'blog/style.css' %}" />

```

<—————

httpresponse - views.py

```

from django.http import HttpResponse

def index(request):

    return HttpResponse("hello i am running")

```

To open a new Page from views.py

```
from django.shortcuts import render, redirect
```

there are two ways to do that

1. `return render(request, 'index.html')`
2. `return redirect('/index.html')`

pass data from one html page to another one

index.html

```
<form action="{% url 'Nhome'%}">
```

```
<input name="fname" placeholder="enter your first name"/>
<input name="lname" placeholder="enter your last name"/>
<button>Submit</button>
```

```
</form>
```

[url.py](#)

```
url(r'^home/$', views.home, name="Nhome"),
```

OR

```
path('home/', views.home, name="Nhome"),
```

[views.py](#)

```
def home(request):
```

```
    var1 = request.GET['fname']
    var2=request.GET['lname']
    return render(request, 'home.html', {'name':var1, 'lname':var2})
```

OR

```
def home(request):
```

```
    var1 = request.GET['fname']
    var2=request.GET['lname']
```

```
    my_context={
        "name": var1,
        "lname": var2
    }
```

```
    return render(request, 'home.html', my_context)
```

[home.html](#)

```
{{ name }} {{ lname }}
```

models in app (vid -> Blog app -> 2),(socialMedia)

```
class Post(models.Model):
```

```
    STATUS_CHOICES=(
```

```
        ('draft', 'Draft'),
        ('published', 'Published')
    )
```

```
    title = models.CharField(max_length=200)
```

```
    pub_dates = models.DateTimeField() // insert -> timezone.datetime.now()
```

```
    body = models.TextField()
```

```

like = models.IntegerField(default=0)
status = models.CharField(max_length=100 ,choices=STATUS_CHOICES, default='draft')

author = models.ForeignKey(          // ex reddit posts.modal

                                User, //from django.contrib.auth.models import User
                                //insert -> postz.author=request.user ## get id of login user

                                on_delete = models.DO_NOTHING,

)

```

then use **makemigrations** command to detects changes(if **makemigrations** command saying **No changes detected**, then make sure app is added in setting.py file.)

Then use **migrate** command to save changes.

enter , insert , set data into database

First import modal.py file in views

```

opp=TUsers.objects.create(Username=name , Usermail=mail)

opp.save()

```

TUsers is name of table

Username is column in table

name is value that we are inserting

enter, insert , set data into database reddit

```

@login_required(login_url='/')
@csrf_exempt
def newpost(request):

    if request.POST['Ptitle'] and request.POST['Purl']:

        postz = Postz()
        postz.title=request.POST['Ptitle']
        postz.url=request.POST['Purl']
        postz.date=timezone.datetime.now()
        postz.author=request.user    # get id of login or current user
        postz.save()

        return HttpResponse(" data ok")

    else:
        return HttpResponse("enter all values")

```

retrieve or get data from database **login**

```

try:
    name = TUsers.objects.get(Usermail=Uemail,Userpass=Upass)
    return HttpResponse('user found')
except:

```

```
return HttpResponseRedirect('user not found')
```

retrieve or get data from Database (vid -> Blog app -> 4)

First you need to do 'Models In App' code

After that in [view.py](#)

```
from .models import Post
```

```
def index(requests):
    posts = Post.objects.order_by('title')
    return render(requests, 'posts/index.html', {'posts':posts})
```

In [index.html](#)

```
{% for post in posts.all %}
    <h1>{{ post.title }}</h1>
    <p>{{ post.body }}</p>
{% endfor %}
```

on click retrieve or get data, dynamic url

[index.html](#)

```
<a href="{% url 'postdetail' post.id %}">
```

[urls.py](#)

```
url(r'^posts/(?P<post_id>[0-9]+)/$', views.post_details, name='postdetail')
```

[views.py](#)

```
def post_details(requests, post_id):
    post = Post.objects.get(pk=post_id)
    return render(requests, 'posts/post_detail.html',{'post': post})
```

[post_detail.html](#)

```
{% for post in posts.all %}
    <h1>{{ post.title }}</h1>
    <p>{{ post.body }}</p>
{% endfor %}
```

on click retrieve or get data, dynamic url 2 example - on click on user show all post posted by that user only (reddit home.html)

[index.html](#)

```
<a href="{% url 'Posts:userpost' post.author_id %}">
```

[urls.py](#)

```
url(r'^userpost/(?P<pk>[0-9]+)', Posts.views.userpostV, name='userpost')
```

[views.py](#)

```
def userpostV(request, pk):
    home_posts=Postz.objects.filter(author_id=pk).order_by('-date')
    return render(request, 'Postz/Home.html', {'home_posts': home_posts})
```

[Home.html](#)

```
{% for post in home_posts.all %}

{{ post.author.username }}

{% endfor %}
```

dynamic url 3 example - show url with slug (www.some.com/1/post-title-slug)

[index.html](#)

```
<a href="{% url 'post_detail' i.id i.slug%}">
```

[urls.py](#)

```
url(r'^blog/(?P<id>\d+)/(?P<slug>[\w-]+)/$', views.post_details, name="post_detail"),
```

[views.py](#)

```
def post_details(request,id,slug):
    data = Post.objects.get(id=id)
    context = {
        'post': data,
    }
    return render(request, 'blog/post_details.html', context)
```

[post_details.html](#)

```
{{ post.title }}
```

or statement in database db

```
from django.db.models import Q
```

```
MyFriends.objects.filter( Q(Friend1 = loginUser) | Q(Friend2 = loginUser))
```

handle 404

[views.py](#)

```
from django.shortcuts import render , get_object_or_404
```

```
post = get_object_or_404(Post, pk=post_id)
return render(requests, 'posts/post_detail.html',{'post':post})
```

ajax form submit, pass, submit multiple form from single, same page ex (twitter)

index.html

```
<form onsubmit="Login(event)" action="{% url 'Nlogin' %}" method="post" id="FormId1" >
  <input id="email" name="email" placeholder="email">
  <input id="password" name="password" placeholder="password">
  <button>Login</button>
</form>
```

js

```
function Login(event) {

    // /* stop form from submitting normally */
    event.preventDefault();

    var email = $("#email").val();
    var password = $("#password").val();

    var frm = $("#FormId1");

    $.ajax({
        type: frm.attr('method'),
        url: frm.attr('action'),
        data: {email:email,password:password},
        success:function(response) {
            alert(response)
        }
    });
}
```

urls.py

```
url(r'^Login/$', views.login, name="Nlogin"),
```

view.py

```
@csrf_exempt
def login(request):

    print('Login')

    print(request.method)

    Uemail = (request.POST.get('email'))
    Upass =(request.POST.get('password'))

    print(Upass)

    try:
        name = TUsers.objects.get(Usermail=Uemail,Userpass=Upass)
        return HttpResponse("success")
    except:
        return HttpResponse("fail")
```

exception handling

how to show which exception occur

except Exception as e: