

Django Blog Tutorials

- Step 1: `django-admin startproject blog`
Step 2: `cd blog`
Step 3: <http://127.0.0.1:8000/>
Step 4: `python manage.py startapp blog_app`
Step 5: `Blog\blog_app\views.py`

```
#Step 5:
from django.http import HttpResponse
#Step 5:
def index(request):
    return HttpResponse("Hello, world. You're at the blog index.")
```

- Step 6: Create a new .py file (urls.py)
`Blog\blog_app\urls.py`

```
from django.urls import path
#Step 6:
from . import views
urlpatterns = [
    path('', views.index),
]
```

- Step 7: `Blog\urls.py`

```
#Step 7
from django.urls import include
urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('blog_app.urls')),
]
```

Step 8: Blog\settings.py

```
INSTALLED_APPS = [  
    'django.contrib.admin',  
    'django.contrib.auth',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
    #Step 8:  
    'blog_app.apps.BlogAppConfig',  
]
```

Step 9: a)Create a new folder :templates [Blog\templates] b)

```
TEMPLATES = [  
    {  
        'BACKEND': 'django.template.backends.django.DjangoTemplates',  
        #Step 9:  
        'DIRS': [os.path.join(BASE_DIR, 'templates')],  
        'APP_DIRS': True,  
        'OPTIONS': {  
            'context_processors': [  
                'django.template.context_processors.debug',  
                'django.template.context_processors.request',  
                'django.contrib.auth.context_processors.auth',  
                'django.contrib.messages.context_processors.messages',  
            ],  
        },  
    ],  
]
```

Step 10: a)Create a new .html file :index.html [Blog\templates\index.html]

b)<h1>Welcome to Django Tutorials</h1>

c) Blog\blog_app\views.py

```
#Step 10:
def index(request):
    return render(request, 'index.html')
```

d) Pass parameters from view to .html

Blog\blog_app\views.py

```
#Step 10:
def index(request):
    argument={
        'name': 'Pranay',
    }
    return render(request, 'index.html', argument)
```

Blog\templates\index.html

```
<h1>Welcome to Django Tutorials</h1>
<h1>{{ name }}</h1>
```

Step 11: ORM(Object Relational Mapper)

a) python manage.py makemigrations

b) python manage.py migrate

c) python manage.py createsuperuser

Username (leave blank to use 'prana'): admin

Email address: pranaykprince@gmail.com

Password:

Password (again):

The password is too similar to the username.

This password is too short. It must contain at least 8 characters.

This password is too common.

Bypass password validation and create user anyway? [y/N]: y

Superuser created successfully.

d) python manage.py runserver

<http://127.0.0.1:8000/admin>

e) Blog\blog_app\models.py

```
from django.db import models

# Step 11 e):
class Blog(models.Model):
    title = models.CharField(max_length=30)
    body = models.TextField()
    date = models.DateTimeField(auto_now_add=True)
    slug = models.SlugField()

    def __str__(self):
        return self.title

    def snippet(self):
        return self.body[:50]+'...'
```

f) Blog\blog_app\models.py

```
#Step 11 f):
from . models import Blog
admin.site.register(Blog)
```

g) python manage.py makemigrations

h) python manage.py migrate

Step 12: Index Page

index.html

```
<div>

    {% for blog in blog %}
        <div class="card" style="width: 18rem;">
            <div class="card-body">
                <h5 class="card-title">{{ blog.title }}</h5>
                <br>
                <p>{{blog.date}}</p>
                <p class="card-text">{{ blog.snippet }}</p>
                <br>
                <a href="/blog/{{blog.slug}}/" class="card-link">
                    View more..
                </a>
            </div>
        </div>
    {% endfor %}

</div>
```

Views.py

```
def index(request):
    blog = Blog.objects.all()
    argument={
        'name':'Pranay',
        'blog':blog,
    }
    return render(request, 'index.html', argument)
```

Step 13:

Urls.py

```
path('blog/<slug>/', views.slug),
```

Views.py

```
def slug(request, slug):
    blog = Blog.objects.filter(slug=slug)
    argument={
        'blog':blog,
    }
    return render(request, 'blogtemplate.html', argument)
```

Blogtemplate.html

```
<div>
    {% for blog in blog %}
        <h1>{{ blog.title }}</h1>
        <br>
        <p class="card-text">{{ blog.body }}</p>
        <br>
        <a href="/" class="card-link">Go back</a>
    </div>
</div>
{% endfor %}
</div>
```

DAY 2

Step 14:

Urls.py

```
path('login', views.login),
```

Login.html

```
<div class="col-md-4">
    <form method="POST" actions="login">
        {% csrf_token %}
        <div class="form-group">
            <label for="exampleInputEmail1">Email address</label>
            <input type="text" name="userid" class="form-control"
id="exampleInputEmail1" aria-describedby="emailHelp" placeholder="Enter
email">
        </div>
        <div class="form-group">
            <label for="exampleInputPassword1">Password</label>
            <input type="password" name="password" class="form-control"
id="exampleInputPassword1" placeholder="Password">
        </div>
        <button type="submit" class="btn btn-primary">Submit</button>
    </form>
</div>
```

```
<div>
    {% for messages in messages %}
        <div class="alert alert-danger" role="alert">
            Invalid username or password
        </div>
    {% endfor %}
</div>
```

Views.py

```
from django.contrib.auth.models import User,auth
```

```
from django.shortcuts import redirect
from django.contrib import messages
```

```
def login(request):
    if request.method == 'POST':
        username=request.POST['userid']
        password=request.POST['password']
        user=auth.authenticate(username=username,password=password)
        if user is not None:
            auth.login(request,user)
            print("Logged In")
            return redirect("/dashboard")
        else:
            messages.error(request,'invalid')
            print("Failed")
            return redirect("/login")
    else:
        return render(request,"login.html")
```


Step 15:

Urls.py

```
path('dashboard', views.dashboard),
```

Views.py

```
def dashboard(request):
    blog = Blog.objects.all()
    argument={
        'name':'Pranay',
        'blog':blog,
    }
    return render(request, 'dashboard.html', argument)
```

Dashboard.html

```
<div>
    <a href="/dashboard/add">Add</a>
</div>
<div>
    <a href="/dashboard/logout">Logout</a>
</div>
<div>
    {% for blog in blog %}
        <h1>{{ blog.title }}</h1>
        <br>
        <p class="card-text">{{ blog.body }}</p>
        <br>
        <form action="dashboard/delete" method="POST">
            {% csrf_token %}
            <input type="hidden" name="id"
value="{{blog.id}}">
            <button type="submit">Delete</button>
        </form>
        <a href="/dashboard/update/{{ blog.slug
}}/">Update</a>
```

```
        </div>
    </div>
    {% endfor %}
</div>
```

Step 16:

Urls.py

```
path('dashboard/add', views.add),
```

Add.html

```
<div class="col-md-4">
    <form action="add" method="POST">
        {% csrf_token %}
        <div class="form-group">
            <label for="exampleFormControlInput1">Title</label>
            <input type="text" name="title" class="form-control"
id="exampleFormControlInput1">
        </div>
        <div class="form-group">
            <label for="exampleFormControlTextarea1">Body</label>
            <textarea class="form-control" name="body"
id="exampleFormControlTextarea1" rows="3"></textarea>
        </div>
        <div class="form-group">
            <label for="exampleFormControlInput1">Slug</label>
            <input type="text" name="slug" class="form-control"
id="exampleFormControlInput1">
        </div>
        <button type="submit">Add</button>
    </form>
</div>
```

Views.py

```
def add(request):  
    if request.method == 'POST':  
        title=request.POST['title']  
        body=request.POST['body']  
        slug=request.POST['slug']  
        blog_add= Blog(title=title,body=body,slug=slug)  
        blog_add.save()  
        return redirect("/dashboard")  
    else:  
        return render(request,"add.html")
```

Step 17:

Urls.py

```
path('dashboard/update/<slug>', views.update),
```

```
path('dashboard/update/<slug>/update', views.update),
```

Views.py

```
def update(request,slug):  
    if request.method == 'POST':  
        blog_id=request.POST['id']  
        title=request.POST['title']  
        body=request.POST['body']  
        slug=request.POST['slug']  
  
        Blog.objects.filter(id=blog_id).update(title=title,body=body,slug=slug)  
        return redirect("/dashboard")  
    else:  
        print(slug)  
        blog = Blog.objects.filter(slug=slug)  
        argument={  
            'blog':blog,  
        }  
    return render(request,"update.html",argument)
```

```
return render(request, "update.html", argument)
```

Update.html

```
<form action="update" method="POST">
    {% csrf_token %}
    {% for blog in blog %}
        <input type="hidden" name="id" value="{{blog.id}}">
        <div class="form-group">
            <label
for="exampleFormControlInput1">Title</label>
            <input type="text" name="title" value="{{
blog.title }}" class="form-control" id="exampleFormControlInput1">
            </div>
            <div class="form-group">
                <label
for="exampleFormControlTextarea1">Body</label>
                <input class="form-control" type="text-area"
name="body" value="{{ blog.body }}" id="exampleFormControlTextarea1">
            </div>
            <div class="form-group">
                <label for="exampleFormControlInput1">Slug</label>
                <input type="text" name="slug" value="{{ blog.slug
}}"class="form-control" id="exampleFormControlInput1">
            </div>
            <button type="submit">Update</button>
        {% endfor %}
    </form>
```

Step 18:

Urls.py

```
path('dashboard/delete', views.delete),
```

Views.py

```
def delete(request):  
    if request.method == 'POST':  
        blog_id=request.POST['id']  
        Blog.objects.filter(id=blog_id).delete()  
        return redirect("/dashboard")  
    else:  
        return redirect("/dashboard")
```

Step 19:

Urls.py

```
path('dashboard/logout', views.logout),
```

Views.py

```
def delete(request):  
    if request.method == 'POST':  
        blog_id=request.POST['id']  
        Blog.objects.filter(id=blog_id).delete()  
        return redirect("/dashboard")  
    else:  
        return redirect("/dashboard")
```