

Django

static files

static files:

Static files are files that don't change when your application is running. Aside from the HTML generated by the server, web applications generally need to serve additional files such as **images, JavaScript, or CSS** necessary to render the complete web page.

Creating static files:

Static files can be created using following commands from root directory or GUI can also be used to create static files.

```
(my_env) C:\django\np>mkdir static  
(my_env) C:\django\np>mkdir static\css  
(my_env) C:\django\np>mkdir static\img  
(my_env) C:\django\np>mkdir static\js
```

Configuring static files:

To use static files in your project, you need the **'django.contrib.staticfiles'** app in your project's **'INSTALLED_APPS'** list. This manages the static files across entire project during development and production.

```
INSTALLED_APPS = [  
    'django.contrib.admin',  
    'django.contrib.auth',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
]
```

Step 1:

First open settings.py file and go to '**STATIC_URL**' setting .This static URL will append to base URL to serve static files.All static files will be served to the webpage using url.Basically all static files collected in STATICFILES_DIRS or STATIC_ROOT are accessed by webpage using this url.

```
STATIC_URL = 'static/'
```

Step 2:

We need to declare STATICFILES_DIRS.STATICFILES_DIRS is responsible for mentioning path at which static files are stored.STATICFILES_DIRS being a list indicates that having multiple static directories is possible.

```
STATIC_URL = 'static/'

STATICFILES_DIRS= [
|   os.path.join(BASE_DIR,"static"),
]
```

Step 3:

Next step is to set STATIC_ROOT.

```
STATIC_URL = 'static/'

STATICFILES_DIRS= [
|   os.path.join(BASE_DIR,"static"),
]

STATIC_ROOT=os.path.join(BASE_DIR,'assets')
```

STATIC_ROOT is the single root directory from where the django application will serve static files in production. The command 'manage.py collectstatic' will automatically

compile all static files throughout the project and dump it into single root directory, which is declared in STATIC_ROOT.

Loading static files in templates:

In HTML file, we can't write python code because the code is only interpreted by python interpreter not the browser. We know that HTML is a static markup language, while Python is a dynamic programming language.

Django template engine is used to separate the design from the python code and allows us to build dynamic web pages.

Open html page and at very top add {% load static %}

```
{% load static %}

<!DOCTYPE html>
<html>
<head>
  <title>Django Central</title>
```

To load css, Javascript, images in template.

```
<link rel="stylesheet" type="text/css" href="{% static 'styles/responsive.css' %}">
```

```
<script src="{% static 'js/custom.js' %}"></script>
```

```

```

