**WB LAB-6**

**ADITYA AGARWAL**

**ROLL NO. 14**

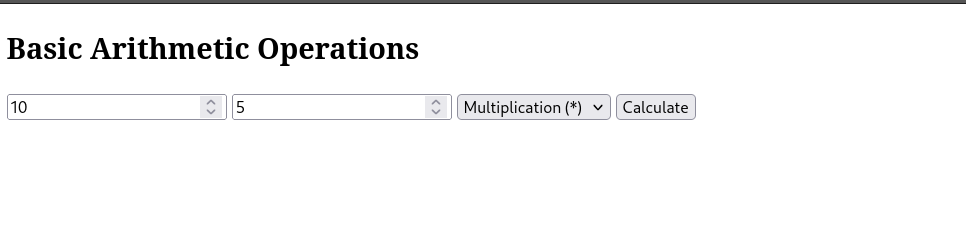
**220905106**

**1.Develop a simple web page to perform basic arithmetic operations. Take two**

**integer inputs from the user, select the operation to be performed using a**

**dropdown. Include a button “Calculate” to perform the selected operation, and**

**then display the result in the same web page.**



urls.py

from django.urls import path

from . import views

urlpatterns = [

path('', views.calculator\_view, name='calculator'),

]

views.py

from django.shortcuts import render

def calculator\_view(request):

result = None

if request.method == 'POST':

try:

num1 = int(request.POST.get('num1', 0))

num2 = int(request.POST.get('num2', 0))

operation = request.POST.get('operation')

if operation == 'add':

result = num1 + num2

elif operation == 'subtract':

result = num1 - num2

elif operation == 'multiply':

result = num1 \* num2

elif operation == 'divide':

if num2 != 0:

result = num1 / num2

else:

result = "Cannot divide by zero"

except ValueError:

result = "Please enter valid numbers"

return render(request, 'calculator/calculator.html', {'result': result})

Calculator.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Basic Calculator</title>

<style>

body {

font-family: Arial, sans-serif;

max-width: 600px;

margin: 50px auto;

padding: 20px;

}

.calculator-form {

background-color: #f5f5f5;

padding: 20px;

border-radius: 8px;

box-shadow: 0 2px 4px rgba(0,0,0,0.1);

}

.form-group {

margin-bottom: 15px;

}

label {

display: block;

margin-bottom: 5px;

font-weight: bold;

}

input[type="number"], select {

width: 100%;

padding: 8px;

border: 1px solid #ddd;

border-radius: 4px;

box-sizing: border-box;

}

button {

background-color: #4CAF50;

color: white;

padding: 10px 20px;

border: none;

border-radius: 4px;

cursor: pointer;

width: 100%;

}

button:hover {

background-color: #45a049;

}

.result {

margin-top: 20px;

padding: 15px;

background-color: #e9f7ef;

border-radius: 4px;

text-align: center;

font-size: 18px;

}

</style>

</head>

<body>

<div class="calculator-form">

<h2 style="text-align: center;">Basic Calculator</h2>

<form method="post">

{% csrf\_token %}

<div class="form-group">

<label for="num1">First Number:</label>

<input type="number" id="num1" name="num1" required>

</div>

<div class="form-group">

<label for="num2">Second Number:</label>

<input type="number" id="num2" name="num2" required>

</div>

<div class="form-group">

<label for="operation">Operation:</label>

<select id="operation" name="operation" required>

<option value="add">Addition (+)</option>

<option value="subtract">Subtraction (-)</option>

<option value="multiply">Multiplication (×)</option>

<option value="divide">Division (÷)</option>

</select>

</div>

<button type="submit">Calculate</button>

</form>

{% if result is not None %}

<div class="result">

Result: {{ result }}

</div>

{% endif %}

</div>

</body>

</html>

manage.py

#!/usr/bin/env python

"""Django's command-line utility for administrative tasks."""

import os

import sys

def main():

"""Run administrative tasks."""

os.environ.setdefault('DJANGO\_SETTINGS\_MODULE', 'lab6.settings')

try:

from django.core.management import execute\_from\_command\_line

except ImportError as exc:

raise ImportError(

"Couldn't import Django. Are you sure it's installed and "

"available on your PYTHONPATH environment variable? Did you "

"forget to activate a virtual environment?"

) from exc

execute\_from\_command\_line(sys.argv)

if \_\_name\_\_ == '\_\_main\_\_':

main()

settings.py

from pathlib import Path

import os

BASE\_DIR = Path(\_\_file\_\_).resolve().parent.parent

SECRET\_KEY = 'django-insecure-your-secret-key'

DEBUG = True

ALLOWED\_HOSTS = []

INSTALLED\_APPS = [

'django.contrib.admin',

'django.contrib.auth',

'django.contrib.contenttypes',

'django.contrib.sessions',

'django.contrib.messages',

'django.contrib.staticfiles',

'calculator',

'magazine',

]

MIDDLEWARE = [

'django.middleware.security.SecurityMiddleware',

'django.contrib.sessions.middleware.SessionMiddleware',

'django.middleware.common.CommonMiddleware',

'django.middleware.csrf.CsrfViewMiddleware',

'django.contrib.auth.middleware.AuthenticationMiddleware',

'django.contrib.messages.middleware.MessageMiddleware',

'django.middleware.clickjacking.XFrameOptionsMiddleware',

]

ROOT\_URLCONF = 'lab6.urls'

TEMPLATES = [

{

'BACKEND': 'django.template.backends.django.DjangoTemplates',

'DIRS': [],

'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',

],

},

},

]

WSGI\_APPLICATION = 'lab6.wsgi.application'

DATABASES = {

'default': {

'ENGINE': 'django.db.backends.sqlite3',

'NAME': BASE\_DIR / 'db.sqlite3',

}

}

AUTH\_PASSWORD\_VALIDATORS = [

{

'NAME': 'django.contrib.auth.password\_validation.UserAttributeSimilarityValidator',

},

{

'NAME': 'django.contrib.auth.password\_validation.MinimumLengthValidator',

},

{

'NAME': 'django.contrib.auth.password\_validation.CommonPasswordValidator',

},

{

'NAME': 'django.contrib.auth.password\_validation.NumericPasswordValidator',

},

]

LANGUAGE\_CODE = 'en-us'

TIME\_ZONE = 'UTC'

USE\_I18N = True

USE\_TZ = True

STATIC\_URL = 'static/'

STATICFILES\_DIRS = [os.path.join(BASE\_DIR, 'static')]

DEFAULT\_AUTO\_FIELD = 'django.db.models.BigAutoField'

lab6/urls.py

from django.contrib import admin

from django.urls import path, include

urlpatterns = [

path('admin/', admin.site.urls),

path('calculator/', include('calculator.urls')),

path('magazine/', include('magazine.urls')),

]

**2.Develop a simple web form that generates the front cover for a magazine. The**

**form should provide the options for selecting the image, background color,**

**changing font size, color etc. Input messages must be taken from the user so**

**as to display it on the front cover with legible font family and font size. The**

front cover developed should be proportionate to the web page size. Place the

css files inside static folder.

Magazine.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Magazine Cover Generator</title>

{% load static %}

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

</head>

<body>

<div class="magazine-container">

<div class="magazine-form">

<h2 style="text-align: center;">Magazine Cover Generator</h2>

<form method="post">

{% csrf\_token %}

<div class="form-group">

<label for="text">Magazine Text:</label>

<input type="text" id="text" name="text" value="{{ text|default:'' }}" required>

</div>

<div class="form-group">

<label for="image\_url">Image URL:</label>

<input type="url" id="image\_url" name="image\_url" value="{{ image\_url|default:'' }}" required

placeholder="https://example.com/image.jpg">

</div>

<div class="form-group">

<label for="font\_size">Font Size (px):</label>

<input type="number" id="font\_size" name="font\_size" value="{{ font\_size|default:'48' }}"

min="12" max="72">

</div>

<div class="form-group">

<label for="font\_family">Font Family:</label>

<select id="font\_family" name="font\_family">

<option value="Arial" {% if font\_family == 'Arial' %}selected{% endif %}>Arial</option>

<option value="Times New Roman" {% if font\_family == 'Times New Roman' %}selected{% endif %}>Times New Roman</option>

<option value="Georgia" {% if font\_family == 'Georgia' %}selected{% endif %}>Georgia</option>

<option value="Verdana" {% if font\_family == 'Verdana' %}selected{% endif %}>Verdana</option>

<option value="Impact" {% if font\_family == 'Impact' %}selected{% endif %}>Impact</option>

</select>

</div>

<div class="form-group">

<label for="font\_color">Font Color:</label>

<input type="color" id="font\_color" name="font\_color" value="{{ font\_color|default:'#000000' }}">

</div>

<div class="form-group">

<label for="bg\_color">Background Color:</label>

<input type="color" id="bg\_color" name="bg\_color" value="{{ bg\_color|default:'#ffffff' }}">

</div>

<button type="submit">Generate Cover</button>

</form>

</div>

{% if submitted %}

<div class="magazine-cover" style="background-color: {{ bg\_color }};">

{% if image\_url %}

<img src="{{ image\_url }}" alt="Magazine Cover" class="magazine-image">

{% endif %}

<div class="preview-text" style="

font-family: '{{ font\_family }}', sans-serif;

font-size: {{ font\_size }}px;

color: {{ font\_color }};

text-shadow: 2px 2px 4px rgba(0,0,0,0.5);">

{{ text }}

</div>

</div>

{% endif %}

</div>

</body>

</html>

urls.py

from django.urls import path

from . import views

urlpatterns = [

path('', views.magazine\_view, name='magazine'),

]

views.py

from django.shortcuts import render

def magazine\_view(request):

context = {}

if request.method == 'POST':

context = {

'text': request.POST.get('text', ''),

'font\_size': request.POST.get('font\_size', '48'),

'font\_family': request.POST.get('font\_family', 'Arial'),

'font\_color': request.POST.get('font\_color', '#000000'),

'bg\_color': request.POST.get('bg\_color', '#ffffff'),

'image\_url': request.POST.get('image\_url', ''),

'submitted': True

}

return render(request, 'magazine/magazine.html', context)

settings.py

from pathlib import Path

import os

BASE\_DIR = Path(\_\_file\_\_).resolve().parent.parent

SECRET\_KEY = 'django-insecure-your-secret-key'

DEBUG = True

ALLOWED\_HOSTS = []

INSTALLED\_APPS = [

'django.contrib.admin',

'django.contrib.auth',

'django.contrib.contenttypes',

'django.contrib.sessions',

'django.contrib.messages',

'django.contrib.staticfiles',

'calculator',

'magazine',

]

MIDDLEWARE = [

'django.middleware.security.SecurityMiddleware',

'django.contrib.sessions.middleware.SessionMiddleware',

'django.middleware.common.CommonMiddleware',

'django.middleware.csrf.CsrfViewMiddleware',

'django.contrib.auth.middleware.AuthenticationMiddleware',

'django.contrib.messages.middleware.MessageMiddleware',

'django.middleware.clickjacking.XFrameOptionsMiddleware',

]

ROOT\_URLCONF = 'lab6.urls'

TEMPLATES = [

{

'BACKEND': 'django.template.backends.django.DjangoTemplates',

'DIRS': [],

'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',

],

},

},

]

WSGI\_APPLICATION = 'lab6.wsgi.application'

DATABASES = {

'default': {

'ENGINE': 'django.db.backends.sqlite3',

'NAME': BASE\_DIR / 'db.sqlite3',

}

}

AUTH\_PASSWORD\_VALIDATORS = [

{

'NAME': 'django.contrib.auth.password\_validation.UserAttributeSimilarityValidator',

},

{

'NAME': 'django.contrib.auth.password\_validation.MinimumLengthValidator',

},

{

'NAME': 'django.contrib.auth.password\_validation.CommonPasswordValidator',

},

{

'NAME': 'django.contrib.auth.password\_validation.NumericPasswordValidator',

},

]

LANGUAGE\_CODE = 'en-us'

TIME\_ZONE = 'UTC'

USE\_I18N = True

USE\_TZ = True

STATIC\_URL = 'static/'

STATICFILES\_DIRS = [os.path.join(BASE\_DIR, 'static')]

DEFAULT\_AUTO\_FIELD = 'django.db.models.BigAutoField'

lab6/urls.py

from django.contrib import admin

from django.urls import path, include

urlpatterns = [

path('admin/', admin.site.urls),

path('calculator/', include('calculator.urls')),

path('magazine/', include('magazine.urls')),

]

