

WEB PROGRAMMING LAB-6

LAB EXERCISES

Date @February 11, 2026

Name: Devadathan N R

Registration No: 230905010

Class: CSE - A1

Roll No: 04

Creating the python environment & installing Django

```
cd ~/Desktop/230905010_DevadathanNR/06_LAB
python3 -m venv djangoenv
source djangoenv/bin/activate
(djangoenv) > python3 -m pip install django
```

Question 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 drop-down. Include a button “Calculate” to perform the selected operation, and then display the result in the same web page.

Create project & App

```
# Create Django project
django-admin startproject calculator-project
cd calculator-project

# Create app
python manage.py startapp calcapp
```

calculator-project/settings.py

```
import os # added at the top

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

STATIC_URL = '/static/'
STATICFILES_DIRS = [BASE_DIR / 'static']
```

Project URLs: calculator-project/urls.py

```
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('calcapp.urls')),
]
```

App URLs: calcapp/urls.py

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

```
urlpatterns = [
    path('', views.calculator, name='calculator'),
]
```

View: [calcapp/views.py](#)

```
from django.shortcuts import render

def calculator(request):
    result = None

    if request.method == "POST":
        num1 = int(request.POST.get('num1'))
        num2 = int(request.POST.get('num2'))
        operation = request.POST.get('operation')

        if operation == 'add':
            result = num1 + num2
        elif operation == 'sub':
            result = num1 - num2
        elif operation == 'mul':
            result = num1 * num2
        elif operation == 'div':
            result = num1 / num2 if num2 != 0 else "Cannot divide by zero"

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

Create template/ in calcapp/. Inside template/

calculator.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Arithmetict Calculator</title>
</head>
<body>
    <h2>Basic Arithmetict Operations</h2>

    <form method="post">
        {%- csrf_token %}

        <label>Number 1:</label>
        <input type="number" name="num1" required><br><br>

        <label>Number 2:</label>
        <input type="number" name="num2" required><br><br>

        <label>Operation:</label>
        <select name="operation">
            <option value="add">Addition</option>
            <option value="sub">Subtraction</option>
            <option value="mul">Multiplication</option>
            <option value="div">Division</option>
        </select><br><br>

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

    {%- if result is not None %}<br>
        <h3>Result: {{ result }}</h3>
    {%- endif %}<br>
</body>
</html>
```

Run

```
python manage.py runserver 8080
```

Output

Basic Arithmetic Operations

Number 1:

Number 2:

Operation:

Result: 100

Question 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.

Create project & App

```
# Create Django project  
django-admin startproject magazine-project  
cd magazine-project
```

```
# Create app
python manage.py startapp coverapp
```

magazine_project/settings.py

```
import os # added at the top

INSTALLED_APPS = [
    'coverapp', # added this line
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]

STATIC_URL = '/static/'
STATICFILES_DIRS = [BASE_DIR / 'static']
```

Project URLs: magazine_project/urls.py

```
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('coverapp.urls')),
]
```

App URLs: coverapp/urls.py

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

urlpatterns = [
    path('', views.cover, name='cover'),
]
```

View: `coverapp/views.py`

```
from django.shortcuts import render

def cover(request):
    context = {}

    if request.method == "POST":
        context = {
            'title': request.POST.get('title'),
            'subtitle': request.POST.get('subtitle'),
            'bgcolor': request.POST.get('bgcolor'),
            'fontcolor': request.POST.get('fontcolor'),
            'fontsize': request.POST.get('fontsize'),
            'image': request.POST.get('image'),
        }

    return render(request, 'cover.html', context)
```

Create template/ in coverapp/. Inside template/

`cover.html`

```

<!DOCTYPE html>
<html>
<head>
    <title>Magazine Cover Generator</title>
    {% load static %}
    <link rel="stylesheet" href="{% static 'style.css' %}">
</head>
<body>

<h2>Magazine Front Cover Generator</h2>

<form method="post">
    {% csrf_token %}

    <label>Title:</label>
    <input type="text" name="title" required><br><br>

    <label>Subtitle:</label>
    <input type="text" name="subtitle"><br><br>

    <label>Background Color:</label>
    <input type="color" name="bgcolor" value="#ffffff"><br><b
r>

    <label>Font Color:</label>
    <input type="color" name="fontcolor" value="#000000"><br>
<br>

    <label>Font Size:</label>
    <input type="number" name="fontsize" value="36"><br><br>

    <label>Select Image:</label>
    <select name="image">
        <option value="img1.jpg">Image 1</option>
        <option value="img2.jpg">Image 2</option>

```

```

</select><br><br>

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

<hr>

{%
  if title %}
<div class="cover"
    data-bgcolor="{{ bgcolor }}"
    data-fontcolor="{{ fontcolor }}"
    data-fontsize="{{ fontsize }}>

    

    <h1 class="title">{{ title }}</h1>
    <h3 class="subtitle">{{ subtitle }}</h3>
</div>
{%
  endif %}

<script>
const cover = document.querySelector('.cover');
if (cover) {
  cover.style.backgroundColor = cover.dataset.bgcolor;
  cover.style.setProperty('--fontcolor', cover.dataset.fontcolor);
  cover.style.setProperty('--fontsize', cover.dataset.fontSize + 'px');
}
</script>

</body>
</html>

```

Run

```
python manage.py runserver 8080
```

Output

Magazine Front Cover Generator

Title:

Subtitle:

Background Color:

Font Color:

Font Size: ^ ▼

Select Image:



New York Times

The Best in town

Question 3

Design a simple web application to provide information about a book. The home page of the application should display the cover page of the book along with three hyperlinks: Metadata, Reviews, Publisher info. Give provision to revert to home page from any other page.

Create project & App

```
# Create Django project
django-admin startproject book-project
cd book-project

# Create app
python manage.py startapp bookapp
```

book-project/settings.py

```
import os # added at the top

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

```
STATIC_URL = '/static/'  
STATICFILES_DIRS = [BASE_DIR / 'static']
```

Project URLs: `book-project/urls.py`

```
from django.contrib import admin  
from django.urls import path, include  
  
urlpatterns = [  
    path('admin/', admin.site.urls),  
    path('', include('bookapp.urls')),  
]
```

App URLs: `bookapp/urls.py`

```
from django.urls import path  
from . import views  
  
urlpatterns = [  
    path('', views.home, name='home'),  
    path('metadata/', views.metadata, name='metadata'),  
    path('reviews/', views.reviews, name='reviews'),  
    path('publisher/', views.publisher, name='publisher'),  
]
```

View: `bookapp/views.py`

```
from django.shortcuts import render  
  
def home(request):
```

```

        return render(request, 'home.html')

def metadata(request):
    return render(request, 'metadata.html')

def reviews(request):
    return render(request, 'reviews.html')

def publisher(request):
    return render(request, 'publisher.html')

```

Create template/ in bookapp/. Inside template/

home.html

```

<!DOCTYPE html>
<html>
<head>
    <title>Book Home</title>
    {% load static %}
    <link rel="stylesheet" href="{% static 'style.css' %}">
</head>
<body>

<div class="container">
    <h1>Book Information System</h1>

    

    <div class="nav">
        <a href="{% url 'metadata' %}">Metadata</a>
        <a href="{% url 'reviews' %}">Reviews</a>
        <a href="{% url 'publisher' %}">Publisher Info</a>
    </div>
</div>

```

```
    </div>
</div>

</body>
</html>
```

metadata.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Metadata</title>
    {% load static %}
    <link rel="stylesheet" href="{% static 'style.css' %}">
</head>
<body>

<div class="container">
    <h2>Book Metadata</h2>

    <div class="content">
        <p><b>Title:</b> The Art of Almost</p>
        <p><b>Author:</b> John Doe</p>
        <p><b>ISBN:</b> 978-1234567890</p>
        <p><b>Genre:</b> Fiction</p>
    </div>

    <div class="back">
        <a href="{% url 'home' %}">Back to Home</a>
    </div>
</div>

</body>
</html>
```

publisher.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Publisher Info</title>
    {% load static %}
    <link rel="stylesheet" href="{% static 'style.css' %}">
</head>
<body>

<div class="container">
    <h2>Publisher Information</h2>

    <div class="content">
        <p><b>Publisher:</b> Open Pages Publishing</p>
        <p><b>Location:</b> New Delhi, India</p>
        <p><b>Year:</b> 2024</p>
    </div>

    <div class="back">
        <a href="{% url 'home' %}">Back to Home</a>
    </div>
</div>

</body>
</html>
```

reviews.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Reviews</title>
    {% load static %}
```

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

<div class="container">
    <h2>Book Reviews</h2>

    <div class="content">
        <p>★★★★★☆ – A thoughtful and engaging read.</p>
        <p>★★★★★★ – Beautifully written and emotionally r
ich.</p>
    </div>

    <div class="back">
        <a href="{% url 'home' %}">Back to Home</a>
    </div>
</div>

</body>
</html>
```

Run

```
python manage.py runserver 8080
```

Output

home.html

Book Information System



[Metadata](#) [Reviews](#) [Publisher Info](#)

[metadata.html](#)

Book Metadata

Title: The Art of Almost

Author: John Doe

ISBN: 978-1234567890

Genre: Fiction

[Back to Home](#)

reviews.html

Book Reviews

★★★★★☆ — A thoughtful and engaging read.

★★★★★ — Beautifully written and emotionally rich.

[Back to Home](#)

publisher.html

Publisher Information

Publisher: Open Pages Publishing

Location: New Delhi, India

Year: 2024

[Back to Home](#)