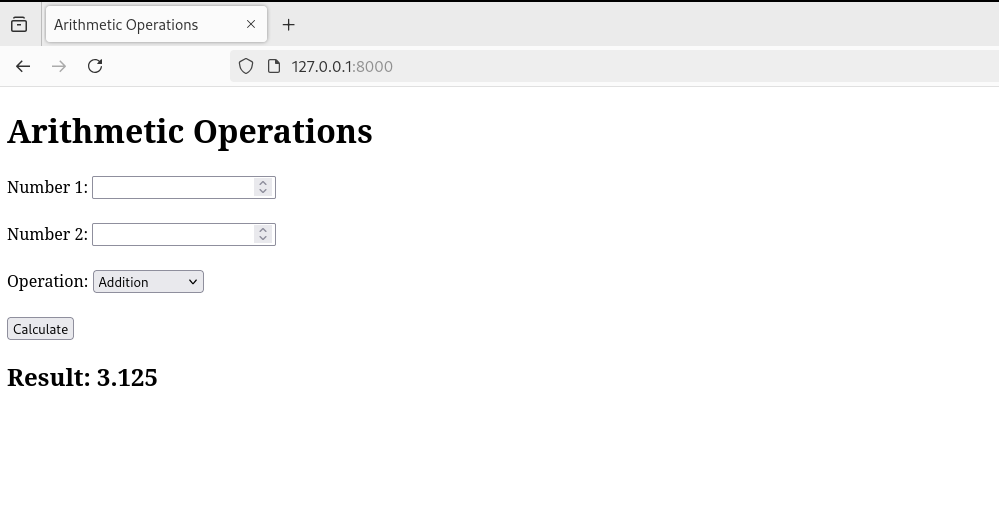
1) Develop a simple web page to perform basic arithmetic operations. Take two

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.contrib import admin

from django.urls import path

from arithmetic\_app.views import calculate

urlpatterns = [

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

path('', calculate, name='calculate'),

]

templates

<!DOCTYPE html>

<html>

<head>

<title>Arithmetic Operations</title>

</head>

<body>

<h1>Arithmetic Operations</h1>

<form method="post">

{% csrf\_token %}

<label for="num1">Number 1:</label>

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

<label for="num2">Number 2:</label>

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

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

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

<option value="add">Addition</option>

<option value="subtract">Subtraction</option>

<option value="multiply">Multiplication</option>

<option value="divide">Division</option>

</select><br><br>

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

</form>

{% if result is not None %}

<h2>Result: {{ result }}</h2>

{% endif %}

</body>

</html>

views.py

from django.shortcuts import render

# Create your views here.

def calculate(request):

result = None

if request.method == "POST":

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":

result = "Cannot divide by zero" if num2 == 0 else num1 / num2

return render(request, "arithmetic.html", {"result": result})

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

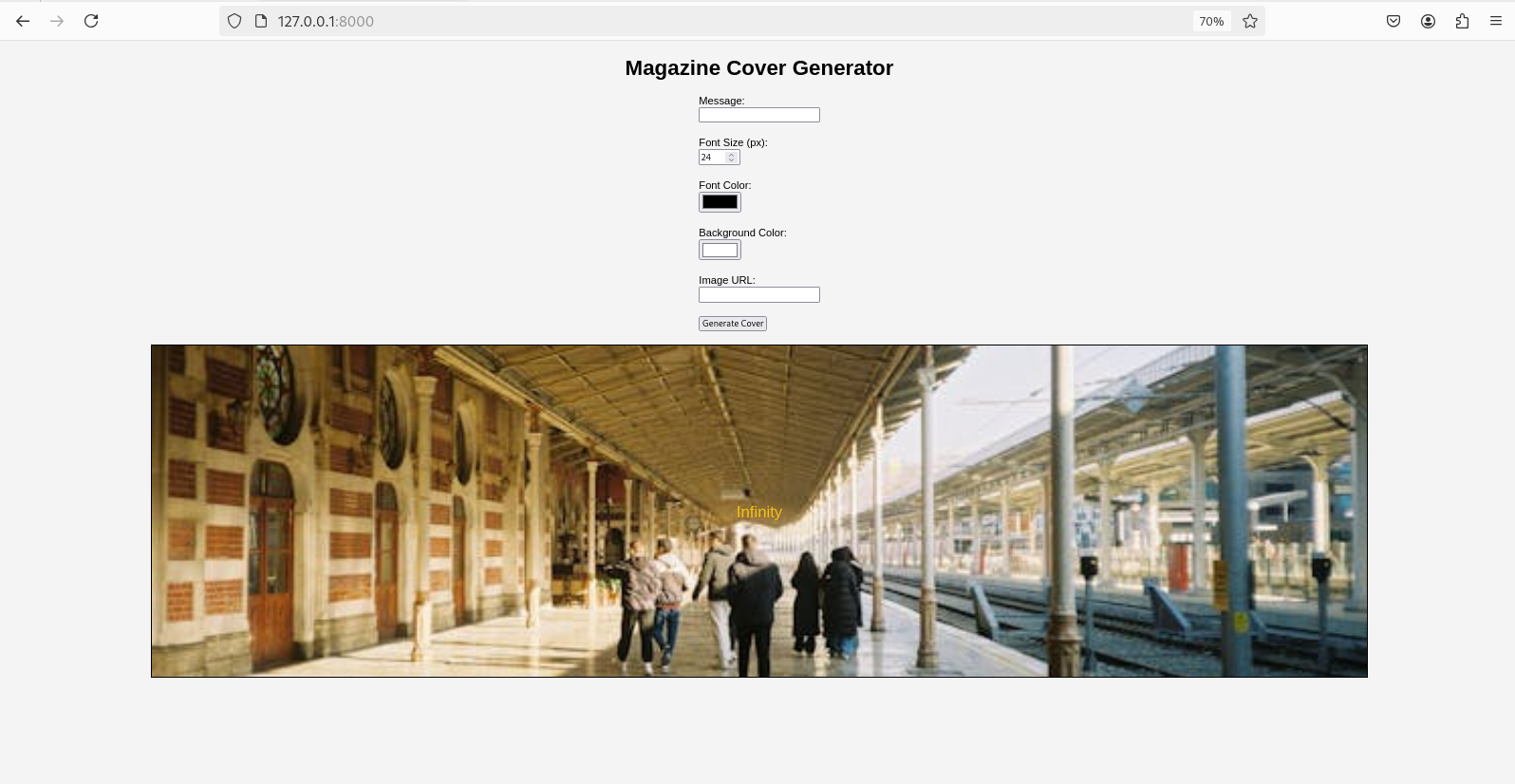
form should provide the options for selecting the image, background color,

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.

Urls.py

from django.contrib import admin

from django.urls import path

from magazine\_app.views import generate\_cover

urlpatterns = [

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

path('', generate\_cover, name='generate\_cover'),

]

views.py

from django.shortcuts import render

# Create your views here.

def generate\_cover(request):

context = {}

if request.method == "POST":

# Get user inputs

message = request.POST.get("message", "")

font\_size = request.POST.get("font\_size", "24")

font\_color = request.POST.get("font\_color", "#000000")

bg\_color = request.POST.get("bg\_color", "#FFFFFF")

image\_url = request.POST.get("image\_url", "")

# Pass inputs to the template

context = {

"message": message,

"font\_size": font\_size,

"font\_color": font\_color,

"bg\_color": bg\_color,

"image\_url": image\_url,

}

return render(request, "magazine.html", context)

Template

<!DOCTYPE html>

<html>

<head>

<title>Magazine Cover Generator</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

display: flex;

flex-direction: column;

align-items: center;

justify-content: center;

background-color: #f4f4f4;

}

.form-container {

margin-bottom: 20px;

}

.cover {

width: 80%;

height: 500px;

display: flex;

align-items: center;

justify-content: center;

position: relative;

border: 2px solid #000;

}

.cover img {

position: absolute;

width: 100%;

height: 100%;

object-fit: cover;

}

.cover .message {

position: relative;

text-align: center;

}

</style>

</head>

<body>

<h1>Magazine Cover Generator</h1>

<div class="form-container">

<form method="post">

{% csrf\_token %}

<label for="message">Message:</label><br>

<input type="text" id="message" name="message" required><br><br>

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

<input type="number" id="font\_size" name="font\_size" min="10" max="100" value="24"><br><br>

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

<input type="color" id="font\_color" name="font\_color" value="#000000"><br><br>

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

<input type="color" id="bg\_color" name="bg\_color" value="#FFFFFF"><br><br>

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

<input type="url" id="image\_url" name="image\_url"><br><br>

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

</form>

</div>

{% if message %}

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

{% if image\_url %}

<img src="{{ image\_url }}" alt="Background Image">

{% endif %}

<div class="message" style="

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

color: {{ font\_color }};">

{{ message }}

</div>

</div>

{% endif %}

</body>

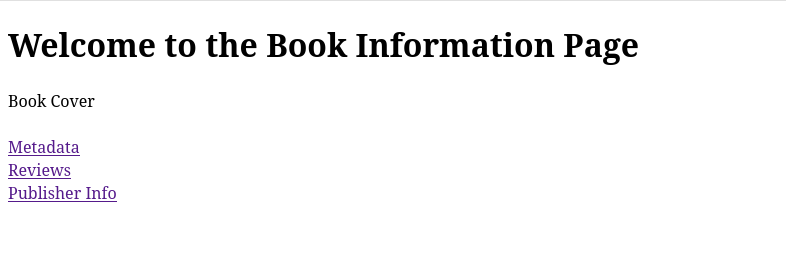
</html>

3) Design a simple web application to provide information about a book. The

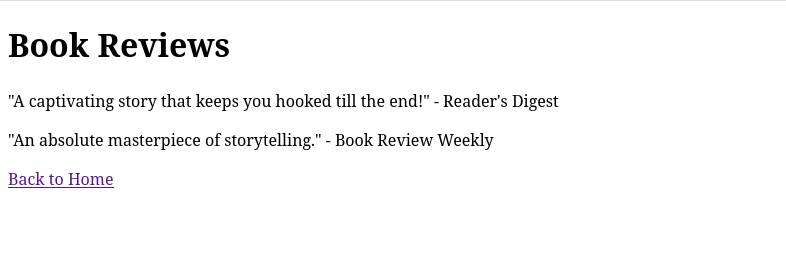
home page of the application should display the cover page of the book along

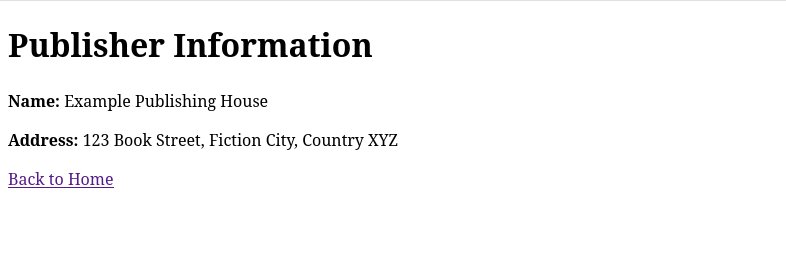
with three hyperlinks: Metadata, Reviews, Publisher info. Give provision to

revert to home page from any other page.









Urls.py

from django.contrib import admin

from django.urls import path

from book\_app import views

urlpatterns = [

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

path('', views.home, name='home'),

path('metadata/', views.metadata, name='metadata'),

path('reviews/', views.reviews, name='reviews'),

path('publisher-info/', views.publisher\_info, name='publisher\_info'),

]

views.py

from django.shortcuts import render

# Create your views here.

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\_info(request):

return render(request, "publisher\_info.html")

Templates

home.html

<!DOCTYPE html>

<html>

<head>

<title>Book Information</title>

</head>

<body>

<h1>Welcome to the Book Information Page</h1>

<img src="https://via.placeholder.com/200x300" alt="Book Cover"><br><br>

<a href="{% url 'metadata' %}">Metadata</a><br>

<a href="{% url 'reviews' %}">Reviews</a><br>

<a href="{% url 'publisher\_info' %}">Publisher Info</a><br>

</body>

</html>

metadata.html

<!DOCTYPE html>

<html>

<head>

<title>Book Metadata</title>

</head>

<body>

<h1>Book Metadata</h1>

<ul>

<li>Title: Example Book</li>

<li>Author: John Doe</li>

<li>Genre: Fiction</li>

<li>Published Year: 2025</li>

<li>ISBN: 123-4567890123</li>

</ul>

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

</body>

</html>

publisher\_info.html

<!DOCTYPE html>

<html>

<head>

<title>Publisher Info</title>

</head>

<body>

<h1>Publisher Information</h1>

<p><strong>Name:</strong> Example Publishing House</p>

<p><strong>Address:</strong> 123 Book Street, Fiction City, Country XYZ</p>

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

</body>

</html>

reviews.html

<!DOCTYPE html>

<html>

<head>

<title>Book Reviews</title>

</head>

<body>

<h1>Book Reviews</h1>

<p>"A captivating story that keeps you hooked till the end!" - Reader's Digest</p>

<p>"An absolute masterpiece of storytelling." - Book Review Weekly</p>

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

</body>

</html>