Web Programming Lab – 7

220905390 CSE – D2 46 RISHIT MANDAL

Lab 6 – Additional Questions

1. 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.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'),
]
```

views.py

```
from django.shortcuts import render

book_info = {
'title': "Atomic Habits",
'author': "James Clear",
'cover_image': "/static/images/book_cover.jpg",
'publisher': "Penguin co. Publisher",
'description': "A supremely practical and useful book. James Clear distils the most fundamental information about habit formation, so you can accomplish more by focusing on less."
}

def home(request):
return render(request, 'book/home.html', {'book': book_info})
```

```
def metadata(request):
return render(request, 'book/metadata.html', {'book': book_info})

def reviews(request):
reviews = [
"Great book! Really enjoyed the storytelling.",
"An amazing read with deep characters and plot.",
"A bit slow in the middle, but the ending is fantastic!"
]
return render(request, 'book/reviews.html', {'book': book_info, 'reviews': reviews})

def publisher(request):
return render(request, 'book/publisher.html', {'book': book_info})
```

html file(s)

home.html

```
<!DOCTYPE html>
<html lang="en">

<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>{{ book.title }} - Home</title>
</head>

<body>
<h1>{{ book.title }}</h1>
<img src="{{ book.cover_image }}" alt="Book Cover" width="200">
<p><strong>Author:</strong> {{ book.author }}</p>
<p><strong>Publisher:</strong> {{ book.publisher }}</p>
<p><strong>Description:</strong> {{ book.description }}</p>
<nav>
<a href="/">Home</a> |
<a href="/metadata/">Metadata</a> |
<a href="/reviews/">Reviews</a> |
<a href="/publisher/">Publisher Info</a>
</nav>
</body>

</html>
```

metadata.html

```
<!DOCTYPE html>
<html lang="en">
```

```
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>{{ book.title }} - Metadata</title>
</head>

<body>
<h1>Metadata for {{ book.title }}</h1>
<p><strong>Title:</strong> {{ book.title }}</p>
<p><strong>Author:</strong> {{ book.author }}</p>
<p><strong>Publisher:</strong> {{ book.publisher }}</p>
<p><strong>Description:</strong> {{ book.description }}</p>
<nav>
<a href="/">Home</a> |
<a href="/reviews/">Reviews</a> |
<a href="/publisher/">Publisher Info</a>
</nav>
</body>

</html>
```

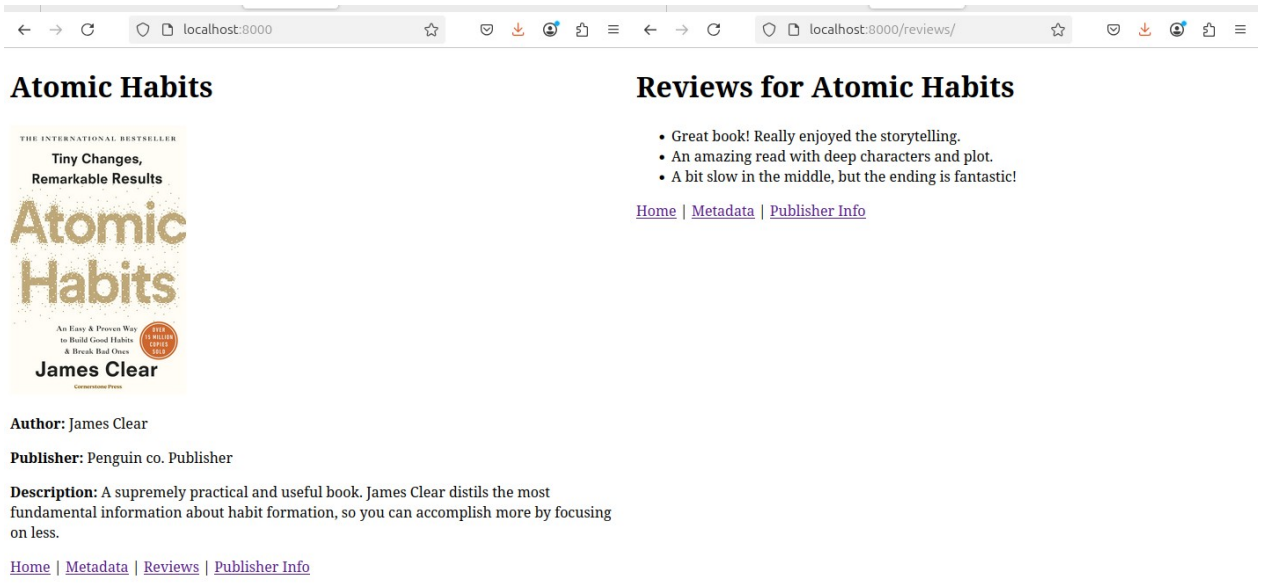
publisher.html

```
<!DOCTYPE html>
<html lang="en">

<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>{{ book.title }} - Publisher Info</title>
</head>

<body>
<h1>Publisher Information for {{ book.title }}</h1>
<p><strong>Publisher:</strong> {{ book.publisher }}</p>
<p><strong>Publisher Description:</strong> Reknowned Publisher for many popular books.
</p>
<nav>
<a href="/">Home</a> |
<a href="/metadata/">Metadata</a> |
<a href="/reviews/">Reviews</a>
</nav>
</body>

</html>
```

review.html

```
<!DOCTYPE html>
<html lang="en">
```

```
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>{{ book.title }} - Reviews</title>
</head>

<body>
<h1>Reviews for {{ book.title }}</h1>
<ul>
{% for review in reviews %}
<li>{{ review }}</li>
{% endfor %}
</ul>
<nav>
<a href="/">Home</a> |
<a href="/metadata/">Metadata</a> |
<a href="/publisher/">Publisher Info</a>
</nav>
</body>

</html>
```



2. Design a simple web application which will ask the user to input his name and a message, display the two items concatenated in a label, and change the format of the label using radio buttons and check boxes for selection, the user can make the label text bold, underlined or italic and change its color. include buttons to display the message in the label, clear the text boxes and label and exit

urls.py

```
from django.contrib import admin
from django.urls import path
from message import views

urlpatterns = [
path('admin/', admin.site.urls),
path('', views.home, name='home'),
]
```

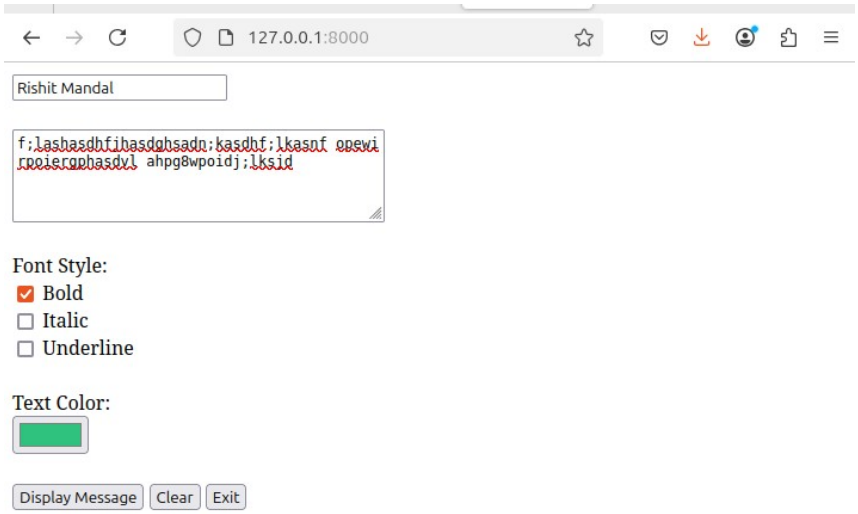
views.py

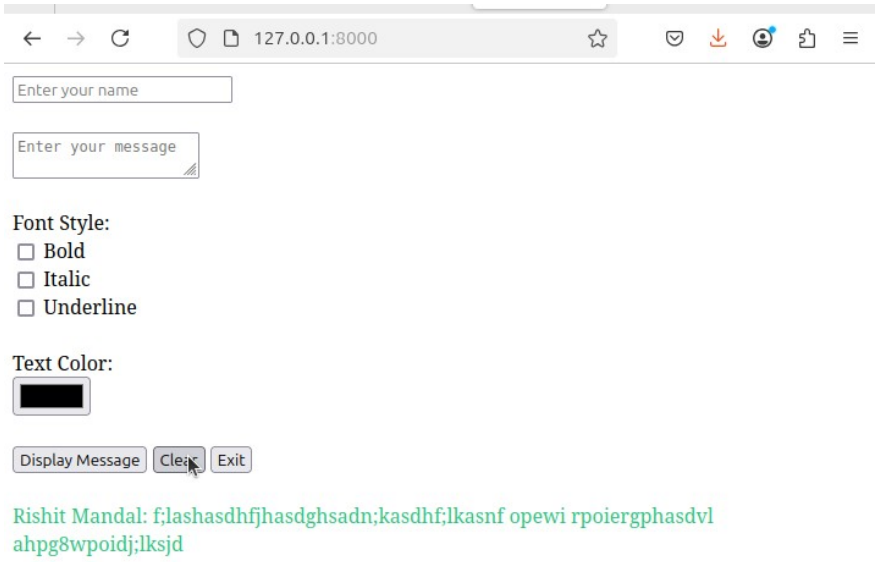
```
from django.shortcuts import render

def home(request):
if request.method == "POST":
name = request.POST.get('name')
message = request.POST.get('message')
font_style = request.POST.get('font_style', '')
text_color = request.POST.get('text_color', 'black')

if font_style == 'bold':
label_style = 'font-weight: bold; color: ' + text_color + ';'
if font_style == 'italic':
label_style = 'font-style: italic; color: ' + text_color + ';'
if font_style == 'underline':
label_style = 'text-decoration: underline; color: ' + text_color + ';'
else:
label_style = 'color: ' + text_color + ';'

label_text = f'{name}: \n {message}'
return render(request, 'home.html', {'label_text': label_text, 'label_style': label_style})

return render(request, 'home.html')
```

html file(s) :

home.html

```
<!DOCTYPE html>
<html lang="en">

<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Message App</title>
</head>

<body>
<form method="POST">
{% csrf_token %}
<input type="text" name="name" placeholder="Enter your name" required><br><br>
<textarea name="message" placeholder="Enter your message" required></textarea><br><br>

<label>Font Style:</label><br>
<input type="checkbox" name="font_style" value="bold"> Bold<br>
<input type="checkbox" name="font_style" value="italic"> Italic<br>
<input type="checkbox" name="font_style" value="underline"> Underline<br><br>

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

<input type="submit" value="Display Message">
<input type="reset" value="Clear">
<input type="button" value="Exit" onclick="window.close();"><br><br>
</form>

{% if label_text %}
<label style="{{ label_style }}">{{ label_text }}</label>
{% endif %}
</body>

</html>
```





Lab Exercises – Lab 7

1) Develop a web application using Django framework to demonstrate the transfer of multiple parameters between web pages. User should be presented with a dropdown list containing car manufacturers, a text box which takes model name of the manufacturer and a submit button. On submitting the web page, the user is forwarded

to a new page. This new page should display the selected car manufacturer name and the model name.

urls.py

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

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

from django.urls import path
from . import views

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

views.py

```
from django.shortcuts import render
from .forms import CarForm

def car_form(request):
if request.method == 'POST':
form = CarForm(request.POST)
if form.is_valid():
manufacturer = form.cleaned_data['manufacturer']
model_name = form.cleaned_data['model_name']
return render(request, 'car/car_result.html', {'manufacturer': manufacturer, 'model_name':
model_name})
else:
form = CarForm()

return render(request, 'car/car_form.html', {'form': form})
```

form.py

```
from django import forms
```

```
class CarForm(forms.Form):
manufacturer_choices = [
('BMW', 'BMW'),
('Mercedes', 'Mercedes'),
('Audi', 'Audi'),
]
manufacturer = forms.ChoiceField(choices=manufacturer_choices)
model_name = forms.CharField(max_length=100)
```

html file(s):

car_form.html

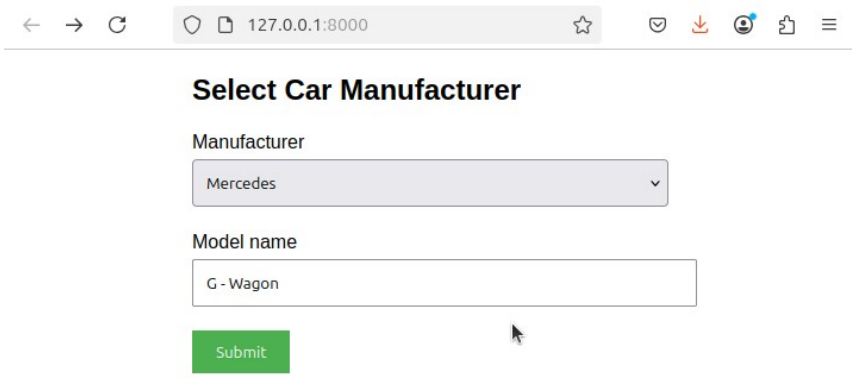
```
<!DOCTYPE html>
<html>

<head>
<title>Car Manufacturer Form</title>
<style>
body {
font-family: Arial, sans-serif;
}

.container {
max-width: 400px;
margin: auto;
}

.form-group {
margin-bottom: 15px;
}

label {
font-weight: bold;
}

input,
select {
width: 100%;
padding: 10px;
margin: 5px 0;
}

button {
background-color: #4CAF50;
```

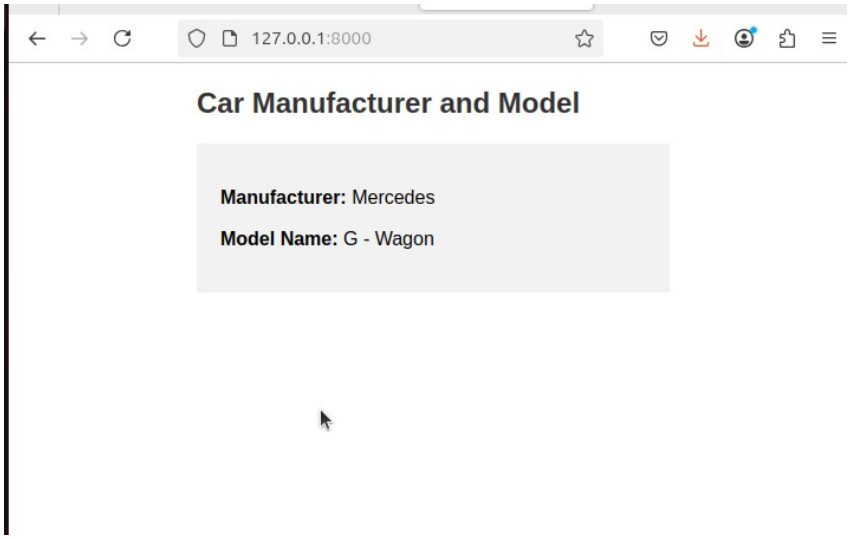
```
color: white;
border: none;
padding: 10px 20px;
cursor: pointer;
}

button:hover {
background-color: #45a049;
}
</style>
</head>

<body>
<div class="container">
<h2>Select Car Manufacturer</h2>
<form method="post">
{% csrf_token %}
<div class="form-group">
{{ form.manufacturer.label }}
{{ form.manufacturer }}
</div>
<div class="form-group">
{{ form.model_name.label }}
{{ form.model_name }}
</div>
<button type="submit">Submit</button>
</form>
</div>
</body>

</html>
```

car_result.html

```
<!DOCTYPE html>
<html>

<head>
<title>Car Model Result</title>
<style>
body {
font-family: Arial, sans-serif;
}

.container {
max-width: 400px;
margin: auto;
```

```
}

.result {
padding: 20px;
background-color: #f2f2f2;
margin-top: 20px;
}

h2 {
color: #333;
}
</style>
</head>

<body>
<div class="container">
<h2>Car Manufacturer and Model</h2>
<div class="result">
<p><strong>Manufacturer:</strong> {{ manufacturer }}</p>
<p><strong>Model Name:</strong> {{ model_name }}</p>
</div>
</div>
</body>

</html>
```





2) Create a page firstPage.html with two TextBoxes [Name, Roll], DropDownList [Subjects], and a button. Create another page secondPage.html with a label and a button. When the user clicks the button in first Page, he should be sent to the second page and display the contents passed from first page in the label. The button in second page should navigate the user back to firstPage. Use Django sessions to transfer information

urls.py

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

urlpatterns = [
```

```
path('', views.first_page, name='first_page'),
path('second/', views.second_page, name='second_page'),
]
```

views.py

```
from django.shortcuts import render, redirect
from .forms import UserInfoForm

def first_page(request):
if request.method == 'POST':
form = UserInfoForm(request.POST)
if form.is_valid():
request.session['name'] = form.cleaned_data['name']
request.session['roll'] = form.cleaned_data['roll']
request.session['subjects'] = form.cleaned_data['subjects']
return redirect('second_page')
else:
form = UserInfoForm()
return render(request, 'firstPage.html', {'form': form})

def second_page(request):
name = request.session.get('name', 'Not Found')
roll = request.session.get('roll', 'Not Found')
subjects = request.session.get('subjects', 'Not Found')
if request.method == 'POST':
return redirect('first_page')
return render(request, 'secondPage.html', {'name': name, 'roll': roll, 'subjects': subjects})
```

forms.py

```
from django import forms

class UserInfoForm(forms.Form):
name = forms.CharField(max_length=100)
roll = forms.CharField(max_length=100)
subjects = forms.ChoiceField(choices=[('Math', 'Math'), ('Science', 'Science'), ('English', 'English')])
```

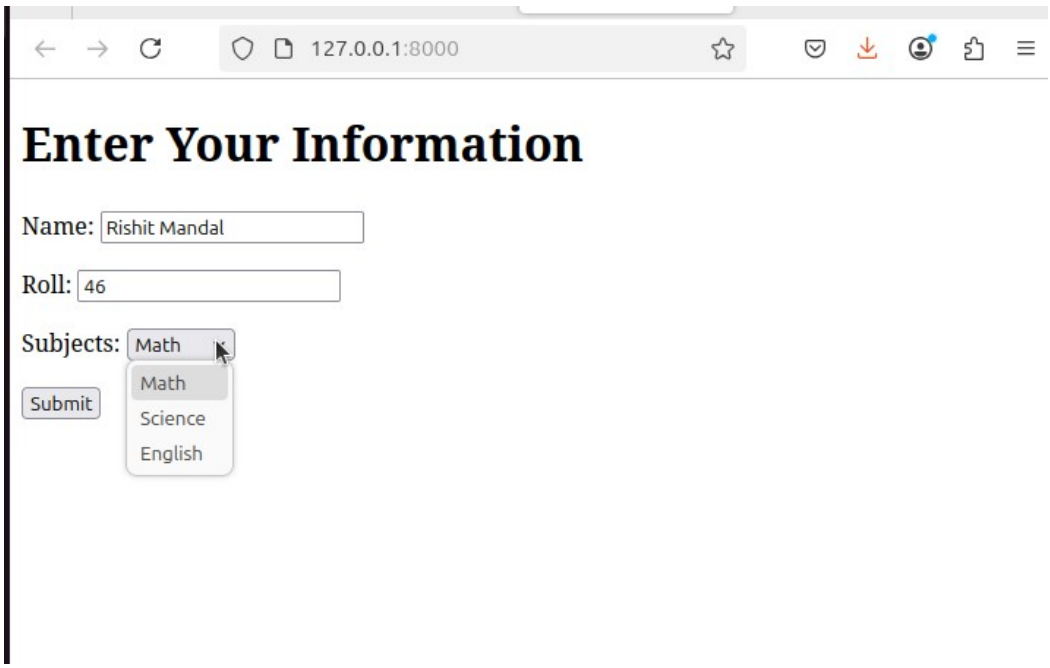
html file(s):

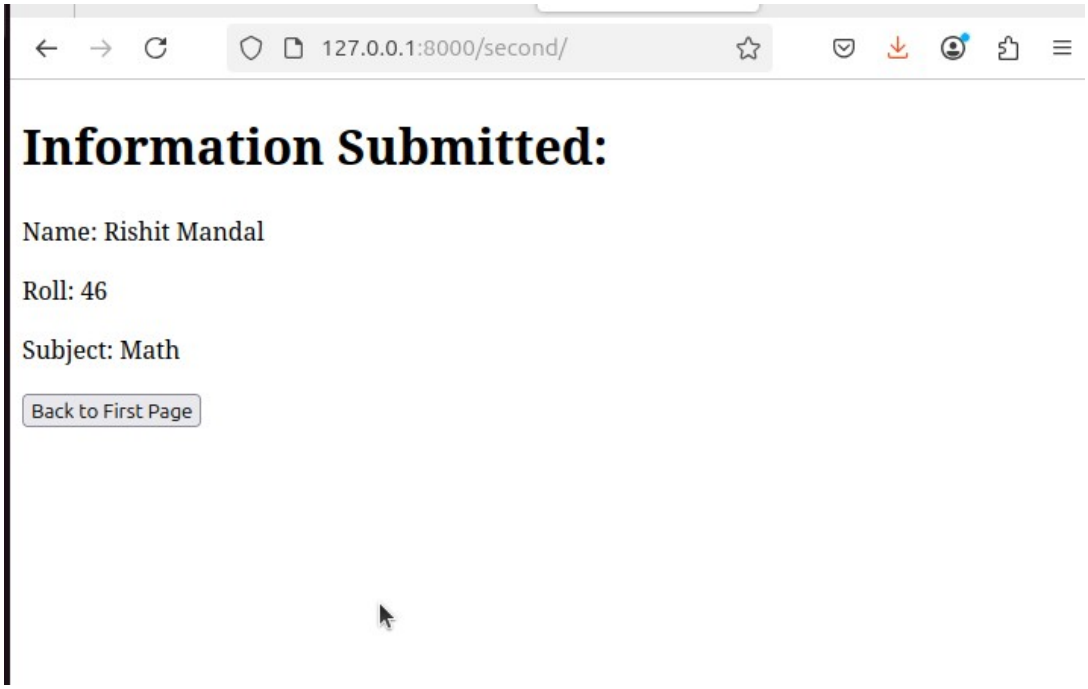
firstpage.html

```
<!DOCTYPE html>
<html lang="en">

<head>
<meta charset="UTF-8">
<title>First Page</title>
</head>

<body>
<h1>Enter Your Information</h1>
<form method="post">
{% csrf_token %}
{{ form.as_p }}
<button type="submit">Submit</button>
</form>
</body>

</html>
```

secondpage.html

```
<!DOCTYPE html>
<html lang="en">

<head>
<meta charset="UTF-8">
<title>Second Page</title>
</head>

<body>
<h1>Information Submitted:</h1>
<p>Name: {{ name }}</p>
<p>Roll: {{ roll }}</p>
<p>Subject: {{ subjects }}</p>
<form method="post">
{% csrf_token %}
<button type="submit">Back to First Page</button>
</form>
</body>

</html>
```


127.0.0.1:8000
Enter Your Information
Name: Rishit Mandal
Roll: 46
Subjects: Math
Math
Science
English
Submit


127.0.0.1:8000/second/
Information Submitted:
Name: Rishit Mandal
Roll: 46
Subject: Math
Back to First Page