

WEEK5 I.T LAB Submission

Name: Sagnik Chatterjee

Reg: 180905478

Roll No :61

Sec : VI B

Base Template File(Using for Q1 ,Q2 AND Q3 and would later for q4 and additional exercises):
base.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <title>Week 5 Programs</title>
  </head>
  <body>
    <header>
      <a href="{% url 'q1' %}">Q1 </a> | <a href="{% url 'q2'
%}">Q2 </a> |
      <a href="{% url 'q3' %}">Q3 </a>
    </header>
    <body>
      {% block content %} {% endblock content %}
    </body>
    <footer>
      <p>Author: Sagnik Chatterjee</p>
      Github:
      <p>
        <a
href="https://github.com/sagnik-chatterjee">sagnik-chatterjee
</a>
```

```
</p>
</footer>
</body>
</html>
```

Project level settings:

lab5 settings.py:

```
"""
Django settings for lab5 project.

Generated by 'django-admin startproject' using Django 3.2.

For more information on this file, see
https://docs.djangoproject.com/en/3.2/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/3.2/ref/settings/
"""

import os
from pathlib import Path

# Build paths inside the project like this: BASE_DIR /
'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent

# Quick-start development settings - unsuitable for
production
```

```
# See
https://docs.djangoproject.com/en/3.2/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production
secret!
SECRET_KEY = "#"

# SECURITY WARNING: don't run with debug turned on in
production!
DEBUG = True

ALLOWED_HOSTS = []

# Application definition

INSTALLED_APPS = [
    "django.contrib.admin",
    "django.contrib.auth",
    "django.contrib.contenttypes",
    "django.contrib.sessions",
    "django.contrib.messages",
    "django.contrib.staticfiles",
    "q1",
    "q2",
    "q3",
]

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 = "lab5.urls"

TEMPLATES = [
    {
        "BACKEND":
"django.template.backends.django.DjangoTemplates",
        "DIRS": [os.path.join(BASE_DIR, "templates")],
        "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 = "lab5.wsgi.application"

# Database
#
https://docs.djangoproject.com/en/3.2/ref/settings/#database
s
```

```
DATABASES = {
    "default": {
        "ENGINE": "django.db.backends.sqlite3",
        "NAME": BASE_DIR / "db.sqlite3",
    }
}

# Password validation
#
https://docs.djangoproject.com/en/3.2/ref/settings/#auth-password-validators

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.NumericPasswordValid
ator",
    },
]

# Internationalization
# https://docs.djangoproject.com/en/3.2/topics/i18n/

LANGUAGE_CODE = "en-us"

TIME_ZONE = "UTC"

USE_I18N = True

USE_L10N = True

USE_TZ = True


# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/3.2/howto/static-files/

STATIC_URL = "/static/"


# Default primary key field type
#
https://docs.djangoproject.com/en/3.2/ref/settings/#default-
auto-field

DEFAULT_AUTO_FIELD = "django.db.models.BigAutoField"
```

Lab5 urls.py:

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

urlpatterns = [
    path("admin/", admin.site.urls),
    path("q1/", include("q1.urls")),
    path("q2/", include("q2.urls")),
    path("q3/", include("q3.urls")),
    #path("q4/", include("q4.urls")),
    # path("/addn1", include("addn1.urls")),
]
```

Q1

Urls.py :-

```
from django.conf.urls import url
from .views import TeacherFormView

urlpatterns = [
    url("", TeacherFormView.as_view(), name="q1"),
]
```

Views.py :-

```
from django.shortcuts import render
from django.views.generic import TemplateView

# Create your views here.

class TeacherFormView(TemplateView):
```

```
template_name = "q1.html"
```

Template :

(q1.html)

```
{% extends 'base.html' %} {% block content %}  
<h2>Q1 Teacher Form</h2>  
<form>  
  <label>Name:</label><br />  
  <input type="text" id="nm" /><br />  
  <label>Date Of Birth:</label><br />  
  <input type="date" id="dob" /><br />  
  <label>Contact Number: </label><br />  
  <input type="number" id="cnum" /><br />  
  <label>E-mail: </label><br />  
  <input type="email" id="smail" /><br />  
  <label>-----MARKS-----</label>  
  <br /><label>English Marks : </label><br />  
  <input type="number" id="emark" /><br />  
  <label>Physics Marks: </label><br />  
  <input type="number" id="pmark" /><br />  
  <label>Chemistry Marks : </label><br />  
  <input type="number" id="cmark" /><br />  
  <button type="button" onclick="checkMarks()">Submit</button>  
</form>  
<br />  
<pre id="det" style="display: none"></pre>  
<label id="avg_marks"></label>  
<script>  
  function checkMarks() {  
    document.getElementById("det").style.display = "none";  
    document.getElementById("det").innerHTML = " ";  
    document.getElementById("avg_marks").innerHTML = " ";  
  }  
</script>
```



```

var s_name = document.getElementById("nm").value;
var s_dob = document.getElementById("dob").value;
var s_contact = document.getElementById("cnum").value;
var s_email = document.getElementById("smail").value;
var e_marks =
parseInt(document.getElementById("emark").value);
var p_marks =
parseInt(document.getElementById("pmark").value);
var c_marks =
parseInt(document.getElementById("cmark").value);
var avg = ((e_marks + p_marks + c_marks) * 100) / 300;
var data = `

                ***** RESULTS *****

                Name of the student: ${s_name}\n
                Date of Birth: ${s_dob}\n
                Contact Number: ${s_contact}\n
                E-mail ID: ${s_email}\n
                -----Marks-----

                English Marks :${e_marks}
                Physics Marks :${p_marks}
                Chemistry Marks :${c_marks}

                The average marks of the Subjects is:-

                ${avg} %

                `;

document.getElementById("det").style.display = "block";
document.getElementById("det").innerHTML = data;
//document.getElementById('avg_marks').innerHTML = avg
}
</script>
{% endblock content %}

```

Screenshot :

Activities Firefox Web Browser May 3 09:23 Week 5 Programs — Mozilla Firefox

[Q1](#) | [Q2](#) | [Q3](#)

Q1 Teacher Form

Name:

Date Of Birth:

Contact Number:

E-mail:

-----MARKS-----

English Marks :

Physics Marks:

Chemistry Marks :

***** RESULTS *****
Name of the student: sagnik

Date of Birth: 2000-07-25

Contact Number: 123456789

E-mail ID: sagnik@gmail.com

-----Marks-----
English Marks :99
Physics Marks :97
Chemistry Marks :98

The average marks of the Subjects is:- 98 %

Author: Sagnik Chatterjee
Github:
[sagnik-chatterjee](#)

127.0.0.1:9000/q1/ Week 5 Programs

Q2:

Urls.py

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

urlpatterns = [
    path("", EmployeeForm.as_view(), name="q2"),
]
```

Views.py

```

from django.shortcuts import render
from django.views.generic import TemplateView

# Create your views here.

class EmployeeForm(TemplateView):
    template_name = "q2.html"

```

Template(Q2.html):

```

{% extends 'base.html' %} {% block content %}
<h1>Employee Operations Q2</h1>
<label for="employee ">Choose a employee id:</label>
<select id="employee_id">
  <option value="180905478">180905478</option>
  <option value="180906478">180906478</option>
  <option value="180907478">180907478</option>
  <option value="180953086">180953086</option>
</select>
<br />
<h2>Date of Joining :</h2>
<br />
<input type="date" id="doj" />
<button type="button" onclick="checkEligible()">Is Eligible
?</button>
<h1 id="control"></h1>
<script>
  function checkEligible() {
    var doj = document.getElementById("doj").value;
    var emp = document.getElementById("employee_id").value;
    var curr_dt = new Date();
    var doj_new = new Date(doj);

```

```
var diff = (curr_dt.getTime() - doj_new.getTime()) / 1000;
diff /= 60 * 60 * 24;
var ans = Math.abs(Math.round(diff / 365.25));
if (ans > 5) {
    document.getElementById(
        "control"
    ).innerHTML = `${emp} is Eligible For Promotion`;
} else {
    document.getElementById(
        "control"
    ).innerHTML = `${emp} is not Eligible For Promotion`;
}
}
</script>
{% endblock content %}
```

Screenshot:

[Q1](#) | [Q2](#) | [Q3](#)

Employee Operations Q2

Choose a employee id: 180906478 ▾

Date of Joining :

24 / 06 / 2000

Is Eligible ?

180906478 is Eligible For Promotion

Author: Sagnik Chatterjee

Github:

[sagnik-chatterjee](#)

Week 5 Programs

Q3

Views.py

```
from django.shortcuts import render

# Create your views here.
from django.views.generic import TemplateView

class BasicArithmetic(TemplateView):
    template_name = "q3.html"
```

Urls.py

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

```
from .views import BasicArithmetic

urlpatterns = [
    path("", BasicArithmetic.as_view(), name="q3"),
]
```

templates(q3.html):

```
{% extends 'base.html' %} {% block content %}
<h1>Calculator Operations Q3</h1>

<label>Enter Number1:</label>
<input type="number" id="num_1" /><br /><br />
<label>Enter Number2:</label>
<input type="number" id="num_2" /><br /><br />
<label>Operation</label>

<select id="operation_id">
  <option selected value="+">+</option>
  <option value="-">-</option>
  <option value="*">*</option>
  <option value="/">/</option>
</select>
<br />
<button type="button" onclick="checkOperations()">Calculate
Result :</button>
<h1 id="control"></h1>
<script>
  function checkOperations() {
    var num1 =
parseInt(document.getElementById("num_1").value);
    var num2 =
parseInt(document.getElementById("num_2").value);
```

```

var op = document.getElementById("operation_id").value;
if (op === "+") {
    var ans = num1 + num2;
    document.getElementById(
        "control"
    ).innerHTML = ` The result is :- ${num1} ${op} ${num2}
:= ${ans}`;
} else if (op === "-") {
    var ans = num1 - num2;
    document.getElementById(
        "control"
    ).innerHTML = ` The result is :- ${num1} ${op} ${num2}
:= ${ans}`;
} else if (op === "*") {
    var ans = num1 * num2;
    document.getElementById(
        "control"
    ).innerHTML = ` The result is :- ${num1} ${op} ${num2}
:= ${ans}`;
} else {
    var ans = num1 / num2;
    document.getElementById(
        "control"
    ).innerHTML = ` The result is :- ${num1} ${op} ${num2}
:= ${ans}`;
}
}
</script>
{% endblock content %}

```

Screenshot:

[Q1](#) | [Q2](#) | [Q3](#)

Calculator Operations Q3

Enter Number1: Enter Number2: Operation

The result is :- $123123 * 12312312313 := 1515928828913499$

Author: Sagnik Chatterjee

Github:

[sagnik-chatterjee](#)