Sl. No	Name of Experiment	WEEK
1	Sample programs	Week1
2	Sample programs	Week2
3	Installation of python,Django and VS code, Creation of virtual environment Django Project and App.	Week3
4	Display Current Date and Time in Server.	Week4
5	Display Current Date and Time four hours ahead and four hours before in server.	Week5
6	Django app that displays an unordered list of fruits and ordered list of selected students for an event	Week6
7	Inherit this layout.html and create 3 additional pages: contact us, About Us and Home page of any website.	Week7
8	Django app that performs student registration to a course. It should also display list of students registered for any selected course. Create students and course as models with enrolment as ManyToMany field.	Week8
9	a. For student and course models created in Lab experiment for Module2, register admin interfaces, perform migrations and illustrate data entry through admin forms b. Develop a Model form for student that contains his topic chosen for project, languages used and duration with a model called project.	Week9
10	For students enrolment developed in Module 2, create a generic class view which displays list of students and detailview that displays student details for any selected student in the list.	Week10
11	Develop example Django app that performs CSV and PDF generation for any models created in previous laboratory component.	Week11

4,5).Django Application to display Display Current Date and Time in Server. Also Display Current Date and Time four hours ahead and four hours before in server.

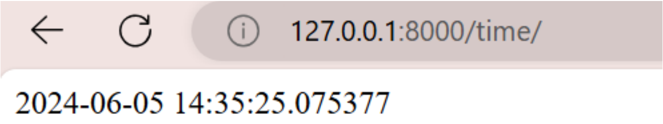
Views.py

```
def atime(request,delta):
    resp=datetime.datetime.now()+datetime.timedelta(hours=delta)
    return HttpResponse(resp)
def ctime(request):
    resp=datetime.datetime.now()
    return HttpResponse(resp)
def btime(request,delta):
    resp=datetime.datetime.now()-datetime.timedelta(hours=delta)
    return HttpResponse(resp)
```

urls.py

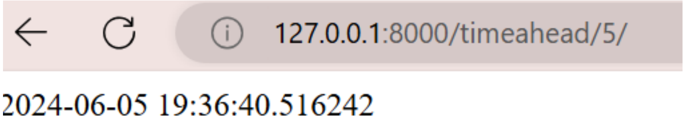
```
    path('time/',views.ctime),
    path('timeahead/<int:delta>/',views.atime),
    path('timebehind/<int:delta>/',views.btime),
```

output:



127.0.0.1:8000/time/

```
2024-06-05 14:35:25.075377
```



127.0.0.1:8000/timeahead/5/

```
2024-06-05 19:36:40.516242
```

6.Django app that displays an unordered list of fruits and ordered list of selected students for an event

Template → fruit_student.html

```
<!DOCTYPE html>

<html>
    <head>
        <title>
            unorderd fruit and ordered student list
        </title>
    </head>
</html>
    <body>
        <h1/> unorded list of fruits </h1>
        <ul>
            {% for fruit in fruitList %}
            <li> {{fruit}}</li>
            {% endfor %}
        </ul>
    <h1/> orded list of students</h1>
    <ol>
        {% for student in studentList %}
        <li> {{student}}</li>
        {% endfor %}
    </ol>
    </body>
</html>
```

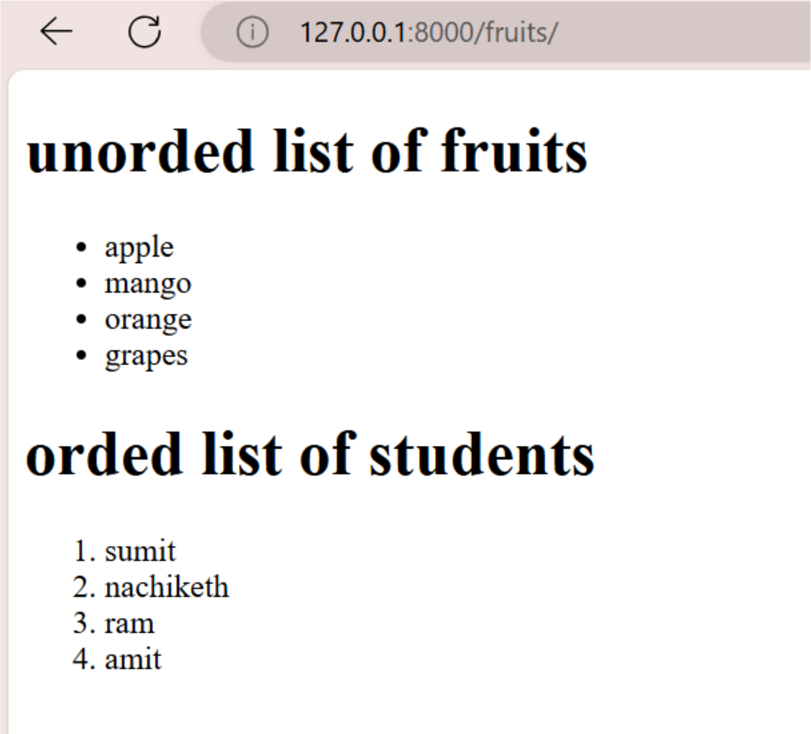
Views.py

```
def fruit_student(request):
    studentList=['sumit','nachiketh','ram','amit']
    fruitList=['apple','mango','orange','grapes']
    return render(request,'fruit_student.html',
           {'fruitList':fruitList,'studentList':studentList})
```

Urls.py

```
 path('fruits/',views.fruit_student),
```

Output:



127.0.0.1:8000/fruits/

unorded list of fruits

- apple
- mango
- orange
- grapes

orded list of students

1. sumit
2. nachiketh
3. ram
4. amit

7.Inherit this layout.html and create 3 additional pages: contact us, About Us and Home page of any website.

Template→layout.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>

        </title>
    </head>

    <body>
        <nav>
            <a href="/home/">HOME </a>
            <a href="/contactus/">CONTACT US </a>
            <a href="/aboutus/">ABOUT US</a>
        </nav>
        <section>
            {% block title %} {% endblock %}
        </section>
        <section>
            {% block content %} {% endblock %}
        </section>
        <footer>
            <hr>
            &copy;full stack development
            </hr>
        </footer>
    </body>
</html>
```

Home.html

```
{% extends 'layout.html' %}
{% block title %} HOME PAGE  {% endblock  %}
{% block content %}
<h1> this is my home page </h1>
{% endblock %}
```

Contactus.html

```
{% extends 'layout.html' %}
{% block title %} CONTACT US  {% endblock  %}
{% block content %}
<h1> CONTACT US </h1>
<P> NAME:FSD </P>
<P> EMAILID:xyz@gmail.com </p>
{% endblock %}
```

Aboutus.html

```
{% extends 'layout.html' %}
{% block title %} ABOUT US  {% endblock  %}
{% block content %}
<h1> ABOUT US </h1>
<P> BLDEACET,vi sem cse,fsd </P>
{% endblock %}
```

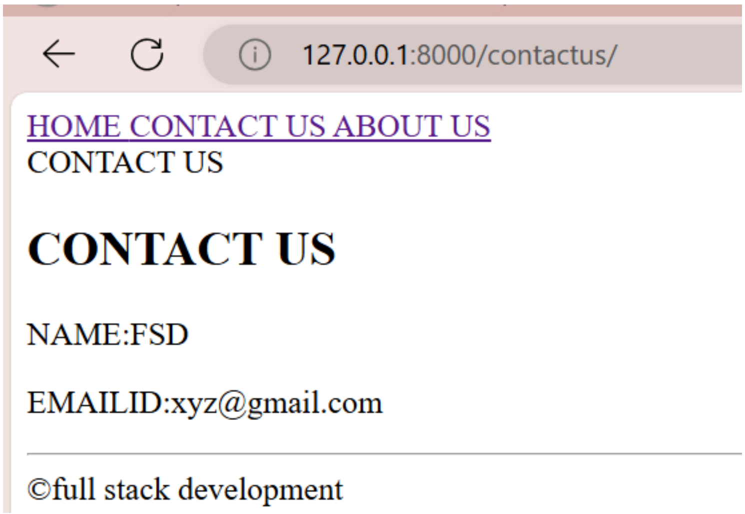
Views.py

```
def home(request):
    return render(request,'home.html')
def contactus(request):
    return render(request,'contactus.html')
def aboutus(request):
    return render(request,'aboutus.html')
```

urls.py

```
    path('contactus/',views.contactus),
    path('aboutus/',views.aboutus),
    path('home/',views.home),
```

output:



127.0.0.1:8000/contactus/

HOME CONTACT US ABOUT US
CONTACT US

CONTACT US

NAME:FSD

EMAILID:xyz@gmail.com

©full stack development

3a.Develop a Django app that performs student registration to a course. It should also display list of students registered for any selected course. Create students and course as models with enrolment as ManyToMany field.

basic.html

```
<!DOCTYPE html>
<html>
    <head>
        <style>
            nav{background-color: lightblue;padding: 15px; }
            nav a {
    color: #fff; /* Text color */
    text-decoration: none; /* Remove underline */
    padding: 10px 20px; /* Padding around each link */
    margin: 0px 10px; /* Spacing between links */
    border-radius: 5px; /* Rounded corners */
    background-color: #555;
    flex-wrap: wrap;
}
nav a:hover {
    background-color:aqua;/* Background color on hover */
}
ul {
    list-style: none; margin: 0;
    padding: 0;
    display: flex; /* Use flexbox */
    flex-wrap: wrap; /* Allow items to wrap to the next line */ flex-
direction: row; /* Display items in a column */
    }
    li {
        margin-right: 20px;
        margin-bottom: 25px;
        }
        </style>
        <title>
            {% block title %} {% endblock %}
        </title>
    </head>
    <body>
        <center> <h1 style="background-color: blue;color:yellow"> STUDENT
COURSE REGISTRATION PORTAL</h1></center>
        <nav>
            <ul>
                <li> <a href="/home/">HOME</a></li>
                <li> <a href="/studentlist/">STUDENT LIST</a></li>
                <li> <a href="/courselist/">COURSE LIST</a> </li>
                <li> <a href="/register/">REGISTER</a></li>
                <li> <a href="/enrolledlist/">ENROLLED LIST</a></li>

            </ul>
        </nav>
        <section>
            {% block content %} {% endblock %}
        </section>
        <footer>
        <hr/>
        <center>
            &copy; Designed and Developeb by Dept. of CSE, BLDEACET.
        </center>
        </footer>
    </body>
</html>
```

courselist.html

```
{% extends 'basic.html' %}
{% block title %} Course List {% endblock %}
{% block content%}
<h1> Course List</h1>
<table border="1">
    <tr>
        <th>
            Sub Code
        </th>
        <th>
            Sub Name
        </th>
        <th>
            Credits
        </th>
    </tr>
    {% for c in course_list %}
    <tr>
        <td>{{c.courseCode}}</td>
        <td>{{c.courseName}}</td>
        <td>{{c.courseCredits}}</td>
    </tr>
    {% endfor %}
</table>
{% endblock %}
```

home.html

```
{% extends 'basic.html' %}
{% block title %} Home Page {% endblock %}
{% block content %}
<li>Click on Student List to get the List of students</li>
<li> Click on Course List to get the list of courses</li>
<li>click on register to enroll student to a course</li>
{% endblock %}
```

register.html

```
{% extends 'basic.html' %}
{% block title %} Course Register Page {% endblock %}
{% block content %}
<h1> Student Course Registration</h1>
<form method="POST" action="">
    {% csrf_token %}
    Select USN:
    <select name="student">
        {% for s in student_list %}
        <option value="{{s.id}}">{{s.usn}}</option>
        {% endfor %}
    </select>
    Select Course:
    <select name="course">
        {% for c in course_list %}
        <option value="{{c.id}}">{{c.courseCode}}</option>
        {% endfor %}
    </select>
    <input type="submit" value="ENROLL"/>
</form>
{% endblock %}
```

studentlist.html

```
{% extends 'basic.html' %}
{% block title %} Student List {% endblock %}
{% block content%}
<h1>Student List</h1>
<table border="1">
    <tr>
        <th>
            USN
        </th>
        <th>
            NAME
        </th>
        <th>
            SEM
        </th>
    </tr>
    {% for s in student_list %}
    <tr>
        <td>{{s.usn}}</td>
        <td>{{s.name}}</td>
        <td>{{s.sem}}</td>
    </tr>
    {% endfor %}
</table>
{% endblock %}
```

Enrolledlist.html

```
{% extends 'basic.html' %}
{% block title %} Course Registration Details {% endblock %}
{% block content %}
<form method="POST" action="">
    {% csrf_token %}
    Select Course:
    <select name="course">
        {% for c in Course_List %}
        <option value="{{c.id}}">{{c.courseCode}}</option>
        {% endfor %}
    </select>
    <input type="submit" value="Search"/>
    {% if student_list %}
    <h1> List of Students registered of the course {{course.courseCode}}</h1>
<table border="1">
    <tr>
        <th>
        USN
        </th>
        <th>
        NAME
        </th>
        <th>
        SEM
        </th>
    </tr>
    {% for s in student_list %}
    <tr>
        <td>{{s.usn}}</td>
```

```
            <td>{{s.name}}</td>
            <td>{{s.sem}}</td>
        </tr>
        {% endfor %}
</table>
{% endif %}
</form>
{% endblock %}
```

models.py

```
from django.db import models

# Create your models here.
from django.db import models
from django.forms import ModelForm # Create your models here.

class course(models.Model):
     courseCode=models.CharField(max_length=10)
     courseName=models.CharField(max_length=50)
     courseCredits=models.IntegerField()

     def __str__ (self):
          return self.courseCode+" "+self.courseName+"
"+str(self.courseCredits)

class student(models.Model):
    usn=models.CharField(max_length=10)
    name=models.CharField(max_length=40)
    sem=models.IntegerField()
    courses=models.ManyToManyField(course,related_name='student_set')

    def __str__ (self):
        return self.usn+" "+self.name+" "+str(self.sem)
```

```
(regenv) PS D:\registration>python manage.py makemigrations
(regenv) PS D:\registration>python manage.py migrate
(regenv) PS D:\registration> python manage.py shell
Python 3.10.0 (tags/v3.10.0:b494f59, Oct  4 2021, 19:00:18) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
(InteractiveConsole)
```

```
>>> from regapp.models import student,course
>>> s1=student(usn='2BL21CS009',name='Harish', sem='6')
>>> s2=student(usn='2BL21CS001',name='kumar1', sem='6')
>>> s3=student(usn='2BL21CS003',name='Pooja', sem='6')
>>> s4=student(usn='2BL21CS004',name='Chetan', sem='6')
>>> s5=student(usn='2BL21CS005',name='Priya', sem='6')
>>> s6=student(usn='2BL21CS006',name='Ram', sem='6')


>>> studlist=[s1,s2,s3,s4,s5,s6]
>>> for i in studlist:
...     i.save()

>>> c1=course(courseCode='21CS61',courseName='SE',courseCredits=3)
>>> c2=course(courseCode='21CS62',courseName='FSD',courseCredits=3)
>>> c3=course(courseCode='21CS63',courseName='CGV',courseCredits=3)
>>> c4=course(courseCode='21CS64',courseName='DBMS',courseCredits=3)
>>> c5=course(courseCode='21CSL62',courseName='FSD Lab',courseCredits=2)


>>> courseList=[c1,c2,c3,c4,c5]
>>> for course in courseList:
...     course.save()
>>>
```

views.py

```
from django.http import HttpResponse
from django.shortcuts import render
from regapp.models import student,course
# Create your views here.

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

def studentlist(request):
    s=student.objects.all()
    return render(request,'studentlist.html',{'student_list':s})

def courselist(request):
    c=course.objects.all()
    return render(request,'courselist.html',{'course_list':c})

def register(request):
    if request.method=="POST":
         sid=request.POST.get("student")
         cid=request.POST.get("course")
         studentobj=student.objects.get(id=sid)
         courseobj=course.objects.get(id=cid)
         res=studentobj.courses.filter(id=cid)
         if res:
             resp="<h1>Student with usn %s has already enrolled for the 
%s<h1>"%(studentobj.usn,courseobj.courseCode)
             return HttpResponse(resp)
         studentobj.courses.add(courseobj)
         resp="<h1>student with usn %s successfully enrolled for the course 
with sub code %s</h1>"%(studentobj.usn,courseobj.courseCode)
         return HttpResponse(resp)
    else:
        studentlist=student.objects.all()
        courselist=course.objects.all()
    return 
render(request,'register.html',{'student_list':studentlist,'course_list':cours
elist})


def enrolledStudents(request):
    if request.method=="POST":
        cid=request.POST.get("course")
        courseobj=course.objects.get(id=cid)
        studentlistobj=courseobj.student_set.all()
        return 
render(request,'enrolledlist.html',{'course':courseobj,'student_list':studentl
istobj})

    else:
        courselist=course.objects.all()
        return render(request,'enrolledlist.html',{'Course_List':courselist})
```

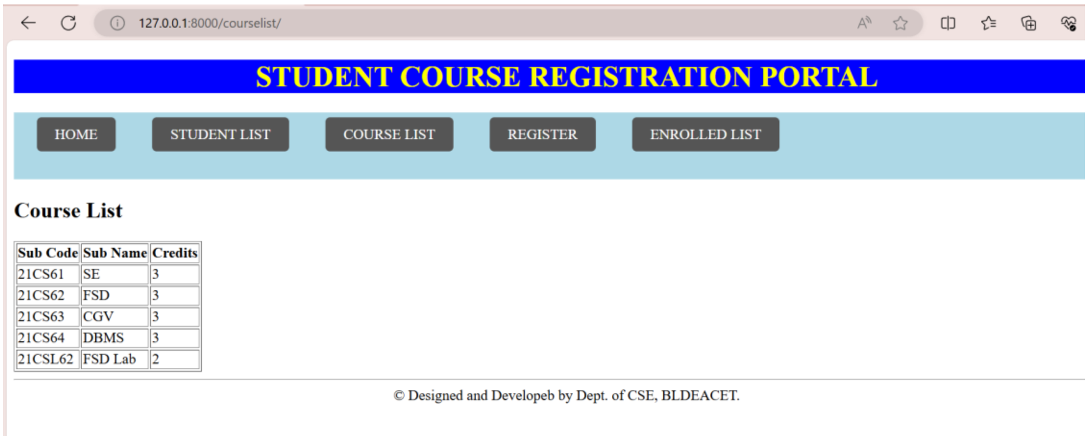
urls.py

```
from django.contrib import admin
from django.urls import path
from regapp.views import home, 
studentlist,courselist,register,enrolledStudents
urlpatterns = [
path('secretadmin/', admin.site.urls),
path('',home),
path('home/',home),
path('studentlist/',studentlist),
path('courselist/',courselist),
path('register/',register),
path('enrolledlist/',enrolledStudents),
]
```

Settings.py

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

ouput

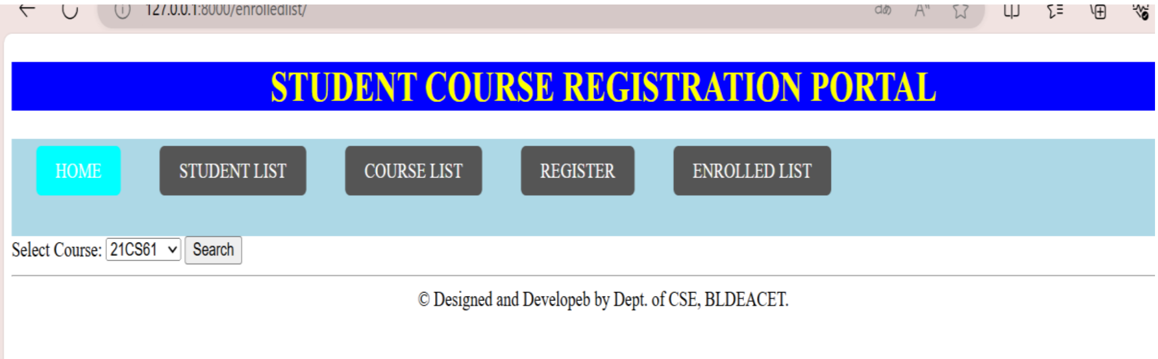


STUDENT COURSE REGISTRATION PORTAL

Course List

Sub Code	Sub Name	Credits
21CS61	SE	3
21CS62	FSD	3
21CS63	CGV	3
21CS64	DBMS	3
21CSL62	FSD Lab	2

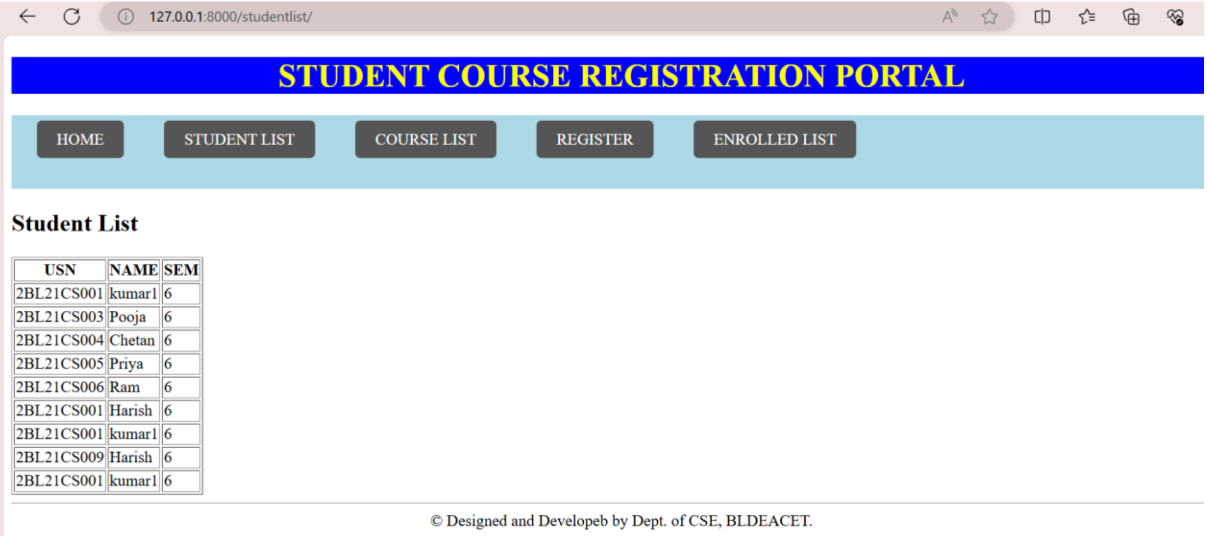
© Designed and Developeb by Dept. of CSE, BLDEACET.



STUDENT COURSE REGISTRATION PORTAL

Select Course: 21CS61

© Designed and Developeb by Dept. of CSE, BLDEACET.

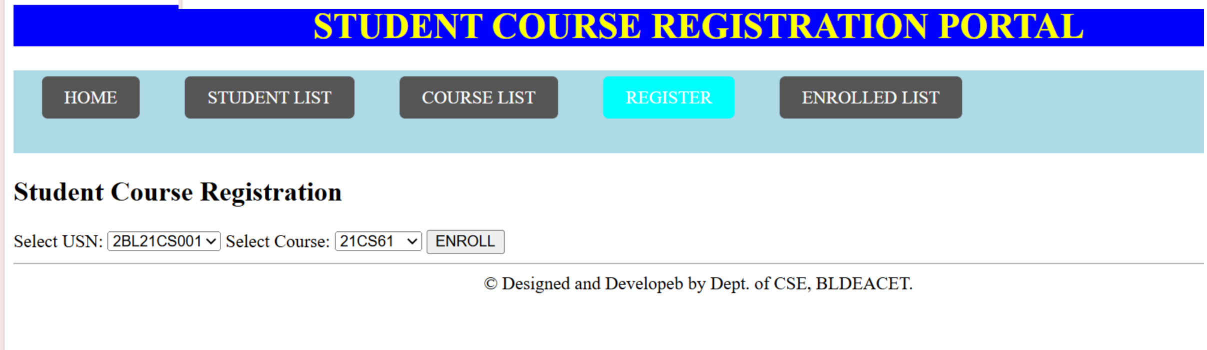


STUDENT COURSE REGISTRATION PORTAL

Student List

USN	NAME	SEM
2BL21CS001	kumar1	6
2BL21CS003	Pooja	6
2BL21CS004	Chetan	6
2BL21CS005	Priya	6
2BL21CS006	Ram	6
2BL21CS001	Harish	6
2BL21CS001	kumar1	6
2BL21CS009	Harish	6
2BL21CS001	kumar1	6

© Designed and Developeb by Dept. of CSE, BLDEACET.



STUDENT COURSE REGISTRATION PORTAL

Student Course Registration

Select USN: 2BL21CS001 Select Course: 21CS61

© Designed and Developeb by Dept. of CSE, BLDEACET.

3b. Develop a Model form for student that contains his topic chosen for project, languages used and duration with a model called project.

Basic.html

Home.html

Studentlist.html

Courselist.html

Register.html

projectreg.html

Models.py → project reg

Views.py→ add_project

url.py→ addproject ->path

admin.py

form.py

projreg.html

```
{% extends 'basic.html' %}
{% block title %} Project Details Registration {% endblock %}
{% block content %}
<form method="POST" action="">
{% csrf_token %}
<table border="1">
{{ form.as_table }}
<tr>
<td>
<input type="submit" value="Add Project"/>
</td>
</tr>
</table>
</form>
{% endblock %}
```

admin.py

```
from django.contrib import admin
from studentapp.models import student,course
# Register your models here.
```

```
#admin.site.register(student)
#admin.site.register(course)
admin.site.site_header='FDP ON Django'
admin.site.site_title='FDP ON Django'
@admin.register(student)
class studentAdmin(admin.ModelAdmin):
    list_display=('usn','name')
    ordering=('usn',)
    search_fields=('name',)
@admin.register(course)
class courseAdmin(admin.ModelAdmin):
    list_display=('courseCode','courseName')
    ordering=('courseCode',)
    search_fields=('courseName',)
```

forms.py

```
from django.forms import ModelForm
from studentapp.models import projectReg


class projectForm(ModelForm):
    required_css_class="required"
    class Meta:
        model=projectReg
        fields=['student','ptitle','planguage','pduration']
```

models.py

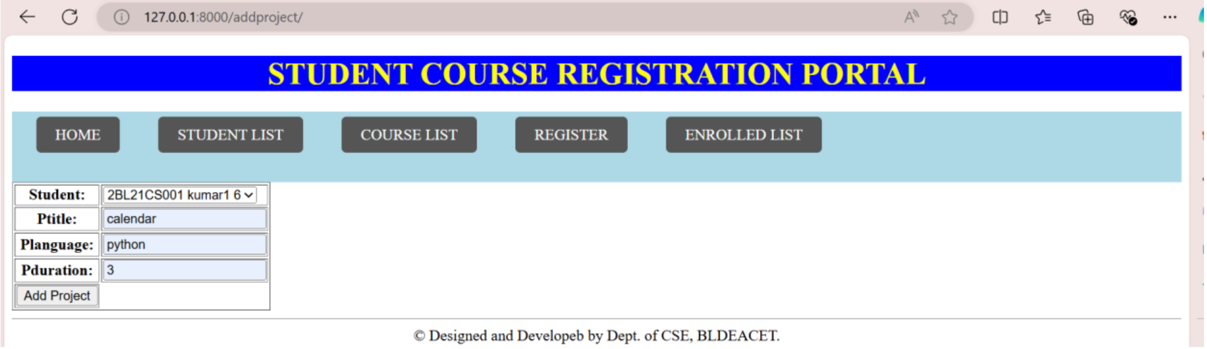
```
class projectReg(models.Model):
    student=models.ForeignKey(student,on_delete=models.CASCADE)
    ptitle=models.CharField(max_length=30)
    planguage=models.CharField(max_length=30)
    pduration=models.IntegerField()
```

views.py

```
def add_project(request):
    if request.method=="POST":
        form=projectForm(request.POST)
        if form.is_valid():
            form.save()
            return HttpResponse("<h1>Project Data Successfully saved</h1>")
        else:
            return HttpResponse("<h1>Project details not saved</h1>")
    else:
        form=projectForm()
        return render(request, "projectReg.html",{'form':form})
```

urls.py

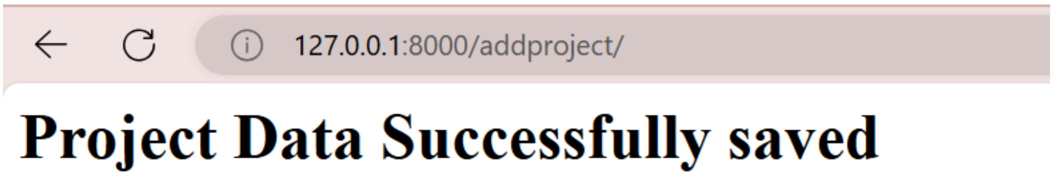
```
path('addproject/',add_project),
```



STUDENT COURSE REGISTRATION PORTAL

Student:	2BL21CS001 kumar1 6
Ptitle:	calendar
Planguage:	python
Pduration:	3
Add Project	

© Designed and Developeb by Dept. of CSE, BLDEACET.



Project Data Successfully saved

4a,b) For students enrolment developed in Module 2, create a generic class view which displays list of students and detailview that displays student details for any selected student in the list. Also display CSV and PDF generation for any models created in previous laboratory component.

Inside template ->create another folder->enrollapp->student_detail.html

```
<!-- students/student_detail.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>{{ student.name }} {{ student.usn }}</title>
</head>
<body>
    <h1>{{ student.name }} </h1>
    <p><strong>USN:</strong> {{ student.usn }}</p>
    <p><strong>SEM:</strong> {{ student.sem }}</p>
</body>
</html>
```

Inside template ->create another folder->enrollapp->student_list.html

```
<html>
{% block content%}

<h2>Students - {{ total_students}}</h2>
<ul>
    {% for student in student_list %}
    <li><a href="/students/{{ student.pk }}">  {{student.name }}</a></li>
    {% endfor %}
</ul>
{% endblock %}

</html>
```

Inside template ->studentlist.html

```
<html>
{% block content %}

<table  border="1px">
    <thead>
        <th>Name</th>
    </thead>
    <tr>
        <td>Student Count: {{total_students}}</td>
        </tr>
```

```
{% for student in student_list %}
    <tr>
        <td>
            <!-- <a href="{% url 'student_detail' student.pk %}">{{
student.name }} </a> -->
            <a href="/students/{{student.pk}}/">{{ student.name }} </a>

        </td>


    </tr>
{% endfor %}
</table>

<a href="/students/csv/">Download Student List</a>
{% endblock %}
</html>
```

Admin.py

```
from django.contrib import admin
from .models import Student,Course
# Register your models here.
class StudentAdmin(admin.ModelAdmin):
    list_display=['name','usn']
    list_filter=['sem']
    search_fields=['usn']
    ordering=['usn']

admin.site.register(Student,StudentAdmin)
admin.site.register(Course)
```

forms.py

```
from django import forms

from enrollapp.models import Course, Student

class ContactForm(forms.Form):
    name = forms.CharField(label='Your name', max_length=100)
    email = forms.EmailField(label='Your email')
    message = forms.CharField(widget=forms.Textarea, label='Your message')

class EnrollmentForm(forms.Form):
    student = forms.ModelChoiceField(queryset=Student.objects.all())
    courses = forms.ModelMultipleChoiceField(queryset=Course.objects.all(),
widget=forms.CheckboxSelectMultiple)
```

models.py

```
from django.db import models

# Create your models here.
class Student(models.Model):
    name=models.CharField(max_length=50)
    usn=models.CharField(max_length=10)
    sem=models.SmallIntegerField()



class Course(models.Model):
    name=models.CharField(max_length=100)
    code=models.CharField(max_length=15)
    credit=models.PositiveSmallIntegerField()
    students=models.ManyToManyField(Student,related_name='courses')
```

view.py

```
from typing import Any
from django.db.models.query import QuerySet
from django.http import HttpResponse
from django.shortcuts import redirect, render
from enrollapp.forms import EnrollmentForm
from .models import Student,Course

# Create your views here.

def slist(request):
    s=Student.objects.all()
    context={'students':s}
    return render(request,'studentlist.html',context)

def enlist(request,ccode):
    courseObj=Course.objects.get(code=ccode)
    sobject=courseObj.students.all()
    context={'course':courseObj,'students':sobject}
    return render(request,'enrolled.html',context)

def enroll_student(request):
    if request.method == 'POST':
        form = EnrollmentForm(request.POST)
        if form.is_valid():
```

```python
            student = form.cleaned_data['student']
            courses = form.cleaned_data['courses']
            student.courses.set(courses)
            return redirect('enroll_student')
    else:
        form = EnrollmentForm()
        return render(request, 'enroll_student.html', {'form': form})

from django.views.generic import ListView

class StudentList(ListView):
    model=Student
    #context_object_name="allstudents"
    template_name='studentlist.html'
    #queryset=Student.objects.filter(sem=4)

    def get_context_data(self, **kwargs: Any):
        data=super().get_context_data(**kwargs)
        data['total_students']=Student.objects.count()
        return data

from django.views.generic import DetailView
class StudentDetailView(DetailView):
    model = Student
    template_name = 'enrollapp/student_detail.html'
    context_object_name = 'student'



class StudentCourseList(ListView):
    model=Course



from django.http import HttpResponse

def example_xml_view(request):
    xml_content = """<?xml version="1.0" encoding="UTF-8"?>
    <response>
        <message>Hello, World!</message>
        <status>success</status>
    </response>"""
    return HttpResponse(xml_content, content_type="application/xml")

import csv
from django.http import HttpResponse
def student_details_csv2(request):
    # Create the HttpResponse object with the appropriate CSV header.
    response = HttpResponse(content_type='text/csv')
    response['Content-Disposition'] = 'attachment; filename="students.csv"'

    # Create the CSV writer using the HttpResponse as the "file".
    writer = csv.writer(response)
    writer.writerow(['Name', 'USN', 'Semester'])

    # Retrieve all student records from the database.
    students = Student.objects.all()

    # Write student data rows to the CSV file.
    for student in students:
        writer.writerow([student.name, student.usn, student.sem])

    return response


from reportlab.lib.pagesizes import letter
from reportlab.pdfgen import canvas
def hello_pdf(request):
    # Create a HttpResponse object with the appropriate PDF headers
    response = HttpResponse(content_type='application/pdf')
    response['Content-Disposition'] = 'attachment; filename="hello_db.pdf"'

    # Create the PDF object, using the response object as its "file"
    p = canvas.Canvas(response,pagesize=letter)
    width, height = letter
    # Draw things on the PDF. Here's where the PDF generation happens.
    students = Student.objects.all()
    y = height - 40
    for x in students:
         p.drawString(100, y, f'Name: {x.name}, USN: {x.usn},sem:{x.sem}')
         y -= 20

    # Close the PDF object cleanly, and we're done
    p.showPage()
    p.save()
    return response
```

urls.py

```
"""
URL configuration for studentproj project.

The `urlpatterns` list routes URLs to views. For more information please see:
    https://docs.djangoproject.com/en/4.2/topics/http/urls/
Examples:
```

```
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  path('', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  path('', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.urls import include, path
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import *
from enrollapp.views import *
from enrollapp import models

urlpatterns = [
    path('admin/', admin.site.urls),
    path('students/', slist,name='slist'),
    path('enroll/',enroll_student,name='enroll_student'),
    path('studentlist/',StudentList.as_view()),
    path('students/<int:pk>/', StudentDetailView.as_view(),
name='student_detail'),
    path('students/csv/',student_details_csv2,name='students_csv'),
    path('xmlv/',example_xml_view),
    path('hellopdf/',hello_pdf),



]
```