

SUBJECT: FULLSTACK DEVELOPMENT (21CS62)

LAB COMPONENT SOLUTIONS

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

course_search.html file in templates folder

```
<html>
  <body>
    <form method="POST" action="">
      Courses
      {% csrf_token %}
      <select name="cname">
        {%for course in courses %}
          <option value="{{course.id}}">{{course.course_name}}</option>
        {% endfor %}
      </select>
      <input type="submit" value="Search">
    </form>
  </body>
</html>
```

selected_students.html

```
<html>
  <body>
    <table border>
      <tr>
        <th>Student Name</th>
        <th>Student USN</th>
        <th>Sem</th>
      </tr>
      {% for student in student_list %}
      <tr>
        <td>{{student.student_name}}</td>
        <td>{{student.student_usn}}</td>
        <td>{{student.student_sem}}</td>
      </tr>
      {% endfor %}
    </table>
  </body>
```

```
</html>
```

views.py

```
from django.http import HttpResponseRedirect
from django.shortcuts import render

from ap3.models import Course, Meeting, ProjectReg, Student

def course_search(request):
    if request.method=="POST":
        cid=request.POST.get("cname")
        s=Student.objects.all()
        student_list=list()
        for student in s:
            if student.enrolment.filter(id=cid):
                student_list.append(student)
        if len(student_list)==0:
            return HttpResponseRedirect("<h1>No Students enrolled</h1>")
        return
    render(request,"selected_students.html",{"student_list":student_list})

    else:
        courses=Course.objects.all()
        return render(request,"course_search.html",{"courses":courses})
```

urls.py

```
from django.contrib import admin
from django.urls import path, re_path
from ap1.views import check_number, current_date_time
from ap1.views import four_hours_after, four_hours_before
from ap1.views import n_hours_after, display_string
from ap2.views import create_table_of_squares, vc, find_mode
from ap2.views import template_test, showlist, list_of_subjects
from ap2.views import aboutus, home, contactus, getpos, stable
from ap3.views import insert_demo, update_demo, delete_demo, retrieve_demo
from ap3.views import reg, course_search

urlpatterns = [
    path('secretadmin/', admin.site.urls),
    path('cdt/', current_date_time),
    path('fha/', four_hours_after),
    path('fhb/', four_hours_before),
    path('nha/<int:num>', n_hours_after),
    path('display_string/<slug:sentence>', display_string),
    re_path('check_number/(\d){1,2}/', check_number),
```

```

path('cts/<int:s>/<int:n>', create_table_of_squares),
path('vc/<str:sentence>', vc),
path('find_mode/<str:listofnum>', find_mode),
path('template_test/', template_test),
path('showlist/', showlist),
path('list_of_subjects/', list_of_subjects),
path('aboutus/', aboutus),
path('home/', home),
path('contactus/', contactus),
path('getpos/', getpos),
path('stable/', stable),
path('insert_demo/', insert_demo),
path('update_demo/', update_demo),
path('delete_demo/', delete_demo),
path('retreive_demo/', retreive_demo),
path('reg/', reg),
path('course_search/', course_search),

```

]

Output:

← → ↻ ⓘ 127.0.0.1:8000/course_search/

Courses

← → ↻ ⓘ 127.0.0.1:8000/course_search/

Student Name	Student USN	Sem
Sham	4JN22AI001	4
Manish	4JN22AI002	4

For student and course models created in Lab experiment for Module2, register admin interfaces, perform migrations and illustrate data entry through admin forms

python manage.py createsuperuser

make following changes to admin.py

```

from django.contrib import admin

from ap3.models import Course, Student

# Register your models here.
#admin.site.register(Student)
@admin.register(Student)
class StudentAdmin(admin.ModelAdmin):
    list_display = ('student_name', 'student_usn', 'student_sem')
    ordering = ('student_name',)
    search_fields = ('student_name',)
admin.site.register(Course)

```

urls.py

```

from django.contrib import admin
from django.urls import path, re_path
from ap1.views import check_number, current_date_time
from ap1.views import four_hours_after, four_hours_before
from ap1.views import n_hours_after, display_string
from ap2.views import create_table_of_squares, vc, find_mode
from ap2.views import template_test, showlist, list_of_subjects
from ap2.views import aboutus, home, contactus, getpos, stable
from ap3.views import insert_demo, update_demo, delete_demo, retrieve_demo
from ap3.views import reg, course_search

admin.site.site_header = "My Site Header"
admin.site.site_title = "My Site Title"
admin.site.index_title = "My Site Index"

urlpatterns = [
    path('secretadmin/', admin.site.urls),
    path('cdt/', current_date_time),
    path('fha/', four_hours_after),
    path('fhb/', four_hours_before),
    path('nha/<int:num>', n_hours_after),
    path('display_string/<slug:sentence>', display_string),
    re_path('check_number/(\d){1,2}/', check_number),
    path('cts/<int:s>/<int:n>', create_table_of_squares),
    path('vc/<str:sentence>', vc),
    path('find_mode/<str:listofnum>', find_mode),
    path('template_test/', template_test),
    path('showlist/', showlist),
    path('list_of_subjects/', list_of_subjects),
    path('aboutus/', aboutus),
    path('home/', home),
    path('contactus/', contactus),
    path('getpos/', getpos),

```

```

    path('stable/', stable),
    path('insert_demo/', insert_demo),
    path('update_demo/', update_demo),
    path('delete_demo/', delete_demo),
    path('retreive_demo/', retreive_demo),
    path('reg/', reg),
    path('course_search/', course_search),

]

```

Changes to models.py

```

from django.db import models
from django.forms import ModelForm

# Create your models here.
class Meeting(models.Model):
    meeting_code=models.CharField(max_length=100)
    meeting_dt=models.DateField(auto_now_add=True)
    meeting_subject=models.CharField(max_length=100)
    meeting_np=models.IntegerField()

class Course(models.Model):
    course_code=models.CharField(max_length=40)
    course_name=models.CharField(max_length=100)
    course_credits=models.IntegerField(blank=True, null=True)
    def __str__(self):
        return self.course_name

class Student(models.Model):
    student_usn=models.CharField(max_length=20)
    student_name=models.CharField(max_length=100)
    student_sem=models.IntegerField()
    enrolment=models.ManyToManyField(Course)
    def __str__(self):
        return self.student_name+"("+self.student_usn+")"

```

Perform remigrations before running:

```
python manage.py makemigrations ap3
```

```
python manage.py migrate
```

Output:

127.0.0.1:8000/secretadmin/login/?next=/secretadmin/

My Site Header

Username:

Password:

Log in

127.0.0.1:8000/secretadmin/

My Site Header

WELCOME, **AIMLJNN**. [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

My Site Index

AP3		
Courses	+ Add	✎ Change
Students	+ Add	✎ Change
AUTHENTICATION AND AUTHORIZATION		
Groups	+ Add	✎ Change
Users	+ Add	✎ Change

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

models.py

```

from django.db import models
from django.forms import.ModelForm

# Create your models here.
class Meeting(models.Model):
    meeting_code=models.CharField(max_length=100)
    meeting_dt=models.DateField(auto_now_add=True)
    meeting_subject=models.CharField(max_length=100)
    meeting_np=models.IntegerField()

class Course(models.Model):
    course_code=models.CharField(max_length=40)
    course_name=models.CharField(max_length=100)
    course_credits=models.IntegerField(blank=True, null=True)
    def __str__(self):
        return self.course_name

class Student(models.Model):
    student_usn=models.CharField(max_length=20)
    student_name=models.CharField(max_length=100)
    student_sem=models.IntegerField()
    enrolment=models.ManyToManyField(Course)
    def __str__(self):
        return self.student_name+"("+self.student_usn+")"

class Project(models.Model):
    student=models.ForeignKey(Student,on_delete=models.CASCADE)
    ptopic=models.CharField(max_length=200)
    plangauges=models.CharField(max_length=200)
    pduration=models.IntegerField()

class ProjectReg(ModelForm):
    required_css_class="required"
    class Meta:
        model=Project
        fields=['student','ptopic','plangauges','pduration']

```

views.py

```

from django.http import HttpResponse
from django.shortcuts import render

from ap3.models import Course, Meeting, ProjectReg, Student

def add_project(request):
    if request.method=="POST":
        form=ProjectReg(request.POST)
        if form.is_valid():

```

```

        form.save()
        return HttpResponseRedirect("<h1>Record inserted successfully</h1>")
    else:
        return HttpResponseRedirect("<h1>Record not inserted</h1>")
    else:
        form=ProjectReg()
        return render(request,"add_project.html",{ "form":form})

```

add_project.html inside templates folder

```

<html>
    <form method="post" action="">
        {% csrf_token %}
        <table>
            {{ form.as_table }}
            <tr>
                <td>
                    <input type="submit" value="Submit">
                </td>
            </tr>
        </table>
    </form>
</html>

```

urls.py

```

from django.contrib import admin
from django.urls import path, re_path
from ap1.views import check_number, current_date_time
from ap1.views import four_hours_after, four_hours_before
from ap1.views import n_hours_after, display_string
from ap2.views import create_table_of_squares, vc, find_mode
from ap2.views import template_test, showlist, list_of_subjects
from ap2.views import aboutus, home, contactus, getpos, stable
from ap3.views import insert_demo, update_demo, delete_demo, retrieve_demo
from ap3.views import reg, course_search, add_project

admin.site.site_header="My Site Header"
admin.site.site_title="My Site Title"
admin.site.index_title="My Site Index"
urlpatterns = [
    path('secretadmin/', admin.site.urls),
    path('cdt/', current_date_time),
    path('fha/', four_hours_after),
    path('fhb/', four_hours_before),
    path('nha/<int:num>', n_hours_after),
    path('display_string/<slug:sentence>', display_string),

```



```

re_path('check_number/(\d){1,2}/', check_number),
path('cts/<int:s>/<int:n>', create_table_of_squares),
path('vc/<str:sentence>', vc),
path('find_mode/<str:listofnum>', find_mode),
path('template_test/', template_test),
path('showlist/', showlist),
path('list_of_subjects/', list_of_subjects),
path('aboutus/', aboutus),
path('home/', home),
path('contactus/', contactus),
path('getpos/', getpos),
path('stable/', stable),
path('insert_demo/', insert_demo),
path('update_demo/', update_demo),
path('delete_demo/', delete_demo),
path('retreive_demo/', retreive_demo),
path('reg/', reg),
path('course_search/', course_search),
path('add_project/', add_project),

```

]

Perform remigrations before running:

`python manage.py makemigrations ap3`

`python manage.py migrate`

Output:

← → ↻ 127.0.0.1:8000/add_project/

Student: Sham(4JN22AI001) ▾

Ptopic: AI

Plangauges: Python,HTML

Pduration: 22

Submit

← → ↻ 127.0.0.1:8000/add_project/

Record inserted successfully

BackEnd

Browse
 Structure
 SQL
 Search
 Insert
 Export
 Refresh

✓ Showing rows 0 - 1 (2 total, Query took 0.0010 seconds.)

SELECT * FROM `ap3_project`

☐ Profiling [Edit inline

☐ Show all | Number of rows: 25 | Filter rows: Search this table

Extra options

				id	ptopic	plangauges	pduration	student_id
<input type="checkbox"/>	Edit	Copy	Delete	1	IoT	Python,Arduino	20	1
<input type="checkbox"/>	Edit	Copy	Delete	2	AI	Python,HTML	22	1

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

views.py

```
from django.views import generic
class StudentListView(generic.ListView):
    model=Student
    template_name="student_list.html"

class StudentDetailView(generic.DetailView):
    model=Student
    template_name="student_detail.html"
```

student_list.html inside templates folder

```
<html>
  <body>
    {% if student_list %}
      <table border>
        <tr>
          <th>USN</th>
          <th>Courses Enrolled</th>
        </tr>
        {% for student in student_list %}
          <tr>
            <td><a href="/student_detail/{{student.pk}}">{{
student.student_usn }}</a></td>
            <td>{% for course in student.enrolment.all %}
              <span>{{ course.course_name }}</span>
              {% endfor %}
            </td>
```

```

        </tr>
        {% endfor %}
    </table>
    {% else %}
        <h1>No Students Enrolled</h1>
    {% endif %}

</body>
</html>

```

student_detail.html inside templates folder

```

<h1>Student Name: {{ student.student_name }}</h1>
<h1>Student USN: {{ student.student_usn }}</h1>
<h1>Student Sem: {{ student.student_sem }}</h1>

```

urls.py

```

from django.contrib import admin
from django.urls import path, re_path
from ap1.views import check_number, current_date_time
from ap1.views import four_hours_after, four_hours_before
from ap1.views import n_hours_after, display_string
from ap2.views import create_table_of_squares, vc, find_mode
from ap2.views import template_test, showlist, list_of_subjects
from ap2.views import aboutus, home, contactus, getpos, stable
from ap3.views import insert_demo, update_demo, delete_demo, retrieve_demo
from ap3.views import reg, course_search, add_project
from ap3.views import StudentListView, StudentDetailView

admin.site.site_header="My Site Header"
admin.site.site_title="My Site Title"
admin.site.index_title="My Site Index"
urlpatterns = [
    path('secretadmin/', admin.site.urls),
    path('cdt/', current_date_time),
    path('fha/', four_hours_after),
    path('fhb/', four_hours_before),
    path('nha/<int:num>', n_hours_after),
    path('display_string/<slug:sentence>', display_string),
    re_path('check_number/(\d){1,2}/', check_number),
    path('cts/<int:s>/<int:n>', create_table_of_squares),
    path('vc/<str:sentence>', vc),
    path('find_mode/<str:listofnum>', find_mode),
    path('template_test/', template_test),
    path('showlist/', showlist),
    path('list_of_subjects/', list_of_subjects),
    path('aboutus/', aboutus),

```

```

path('home/', home),
path('contactus/', contactus),
path('getpos/', getpos),
path('stable/', stable),
path('insert_demo/', insert_demo),
path('update_demo/', update_demo),
path('delete_demo/', delete_demo),
path('retreive_demo/', retreive_demo),
path('reg/', reg),
path('course_search/', course_search),
path('add_project/', add_project),
path('student_list/', StudentListView.as_view()),
path('student_detail/<int:pk>/', StudentDetailView.as_view()),

```

]

Output:

← → ↻ ⓘ 127.0.0.1:8000/student_list/

USN	Courses Enrolled
4JN22AI001	DAA MPC UHV
4JN22AI002	DAA MPC
4JN22AI003	

← → ↻ ⓘ 127.0.0.1:8000/student_detail/1/

Student Name: Sham

Student USN: 4JN22AI001

Student Sem: 4

Program to illustrate returning a constructed csv file to browser

views.py

```

from django.http import HttpResponse
from django.shortcuts import render

from ap3.models import Course, Meeting, ProjectReg, Student

```

```

import csv
def construct_csv(request):
    districts=["Shimoga","Bhadravathi","Bangalore","Dharwad","Raichur"]
    temperatures=[38,36,34,35,40]
    response=HttpResponse(content_type="text/csv")
    response['Content-Disposition'] = 'attachment;
filename="district_temperature.csv"'
    writer=csv.writer(response)
    writer.writerow(["Districts","Temperatures"])
    for d,t in zip(districts,temperatures):
        writer.writerow([d,t])
    return response

```

urls.py

```

from django.contrib import admin
from django.urls import path, re_path
from ap1.views import check_number, current_date_time
from ap1.views import four_hours_after, four_hours_before
from ap1.views import n_hours_after,display_string
from ap2.views import create_table_of_squares,vc,find_mode
from ap2.views import template_test,showlist,list_of_subjects
from ap2.views import aboutus,home,contactus,getpos,stable
from ap3.views import insert_demo,update_demo,delete_demo,retreive_demo
from ap3.views import reg,course_search,add_project
from ap3.views import StudentListView,StudentDetailView,construct_csv

```

```

admin.site.site_header="My Site Header"
admin.site.site_title="My Site Title"
admin.site.index_title="My Site Index"
urlpatterns = [
    path('secretadmin/', admin.site.urls),
    path('cdt/', current_date_time),
    path('fha/', four_hours_after),
    path('fhb/', four_hours_before),
    path('nha/<int:num>', n_hours_after),
    path('display_string/<slug:sentence>', display_string),
    re_path('check_number/(\d){1,2}/',check_number),
    path('cts/<int:s>/<int:n>', create_table_of_squares),
    path('vc/<str:sentence>', vc),
    path('find_mode/<str:listofnum>', find_mode),
    path('template_test/', template_test),
    path('showlist/', showlist),
    path('list_of_subjects/', list_of_subjects),
    path('aboutus/', aboutus),
    path('home/', home),
    path('contactus/', contactus),
    path('getpos/', getpos),

```

```

path('stable/', stable),
path('insert_demo/', insert_demo),
path('update_demo/', update_demo),
path('delete_demo/', delete_demo),
path('retreive_demo/', retreive_demo),
path('reg/', reg),
path('course_search/', course_search),
path('add_project/', add_project),
path('student_list/', StudentListView.as_view()),
path('student_detail/<int:pk>/', StudentDetailView.as_view()),
path('construct_csv/', construct_csv),

```

```
]
```

Output:

CSV file is generated and downloaded

Develop example Django app that performs CSV generation for any models created in previous laboratory component

views.py

```

from django.http import HttpResponseRedirect
from django.shortcuts import render

from ap3.models import Course, Meeting, ProjectReg, Student

import csv
def construct_csv_from_model(request):
    courses=Course.objects.all()
    response=HttpResponse(content_type="text/csv")
    response['Content-Disposition'] = 'attachment;
filename="courses_data.csv"'
    writer=csv.writer(response)
    writer.writerow(["Course Name","Course Code","Credits"])
    for course in courses:
        writer.writerow([course.course_name,course.course_code,course.course_c
redits])
    return response

```

urls.py

```

from django.contrib import admin
from django.urls import path, re_path
from ap1.views import check_number, current_date_time
from ap1.views import four_hours_after, four_hours_before
from ap1.views import n_hours_after,display_string
from ap2.views import create_table_of_squares,vc,find_mode

```

```

from ap2.views import template_test,showlist,list_of_subjects
from ap2.views import aboutus,home,contactus,getpos,stable
from ap3.views import insert_demo,update_demo,delete_demo,retreive_demo
from ap3.views import reg,course_search,add_project
from ap3.views import StudentListView,StudentDetailView,construct_csv
from ap3.views import construct_csv_from_model

admin.site.site_header="My Site Header"
admin.site.site_title="My Site Title"
admin.site.index_title="My Site Index"
urlpatterns = [
    path('secretadmin/', admin.site.urls),
    path('cdt/', current_date_time),
    path('fha/', four_hours_after),
    path('fhb/', four_hours_before),
    path('nha/<int:num>', n_hours_after),
    path('display_string/<slug:sentence>', display_string),
    re_path('check_number/(\d){1,2}/', check_number),
    path('cts/<int:s>/<int:n>', create_table_of_squares),
    path('vc/<str:sentence>', vc),
    path('find_mode/<str:listofnum>', find_mode),
    path('template_test/', template_test),
    path('showlist/', showlist),
    path('list_of_subjects/', list_of_subjects),
    path('aboutus/', aboutus),
    path('home/', home),
    path('contactus/', contactus),
    path('getpos/', getpos),
    path('stable/', stable),
    path('insert_demo/', insert_demo),
    path('update_demo/', update_demo),
    path('delete_demo/', delete_demo),
    path('retreive_demo/', retreive_demo),
    path('reg/', reg),
    path('course_search/', course_search),
    path('add_project/', add_project),
    path('student_list/', StudentListView.as_view()),
    path('student_detail/<int:pk>', StudentDetailView.as_view()),
    path('construct_csv/', construct_csv),
    path('construct_csv_from_model/', construct_csv_from_model),
]

```

Output:

CSV file is generated and downloaded

Develop example Django app that performs PDF generation for any models created in previous laboratory component

views.py

```
from django.http import HttpResponseRedirect
from django.shortcuts import render

from ap3.models import Course, Meeting, ProjectReg, Student
from reportlab.pdfgen import canvas

def construct_pdf_from_model(request):
    courses=Course.objects.all()
    response=HttpResponse(content_type="application/pdf")
    response['Content-Disposition'] = 'attachment;
filename="courses_data.pdf"'
    c=canvas.Canvas(response)

    c.drawString(70,720,"Course Name")
    c.drawString(170,720,"Course Code")
    c.drawString(270,720,"Credits")
    y=660
    for course in courses:
        c.drawString(70,y,course.course_name)
        c.drawString(170,y,course.course_code)
        c.drawString(270,y,str(course.course_credits))
        y=y-60
    c.showPage()
    c.save()
    return response
```

urls.py

```
from django.contrib import admin
from django.urls import path, re_path
from ap1.views import check_number, current_date_time
from ap1.views import four_hours_after, four_hours_before
from ap1.views import n_hours_after, display_string
from ap2.views import create_table_of_squares, vc, find_mode
from ap2.views import template_test, showlist, list_of_subjects
from ap2.views import aboutus, home, contactus, getpos, stable
from ap3.views import insert_demo, update_demo, delete_demo, retrieve_demo
from ap3.views import reg, course_search, add_project
from ap3.views import StudentListView, StudentDetailView, construct_csv
from ap3.views import construct_csv_from_model
from ap3.views import construct_pdf_from_model
```



```

admin.site.site_header="My Site Header"
admin.site.site_title="My Site Title"
admin.site.index_title="My Site Index"
urlpatterns = [
    path('secretadmin/', admin.site.urls),
    path('cdt/', current_date_time),
    path('fha/', four_hours_after),
    path('fhb/', four_hours_before),
    path('nha/<int:num>', n_hours_after),
    path('display_string/<slug:sentence>', display_string),
    re_path('check_number/(\d){1,2}/', check_number),
    path('cts/<int:s>/<int:n>', create_table_of_squares),
    path('vc/<str:sentence>', vc),
    path('find_mode/<str:listofnum>', find_mode),
    path('template_test/', template_test),
    path('showlist/', showlist),
    path('list_of_subjects/', list_of_subjects),
    path('aboutus/', aboutus),
    path('home/', home),
    path('contactus/', contactus),
    path('getpos/', getpos),
    path('stable/', stable),
    path('insert_demo/', insert_demo),
    path('update_demo/', update_demo),
    path('delete_demo/', delete_demo),
    path('retreive_demo/', retreive_demo),
    path('reg/', reg),
    path('course_search/', course_search),
    path('add_project/', add_project),
    path('student_list/', StudentListView.as_view()),
    path('student_detail/<int:pk>', StudentDetailView.as_view()),
    path('construct_csv/', construct_csv),
    path('construct_csv_from_model/', construct_csv_from_model),
    path('construct_pdf_from_model/', construct_pdf_from_model),
]

```

Output:

PDF file is generated and downloaded