

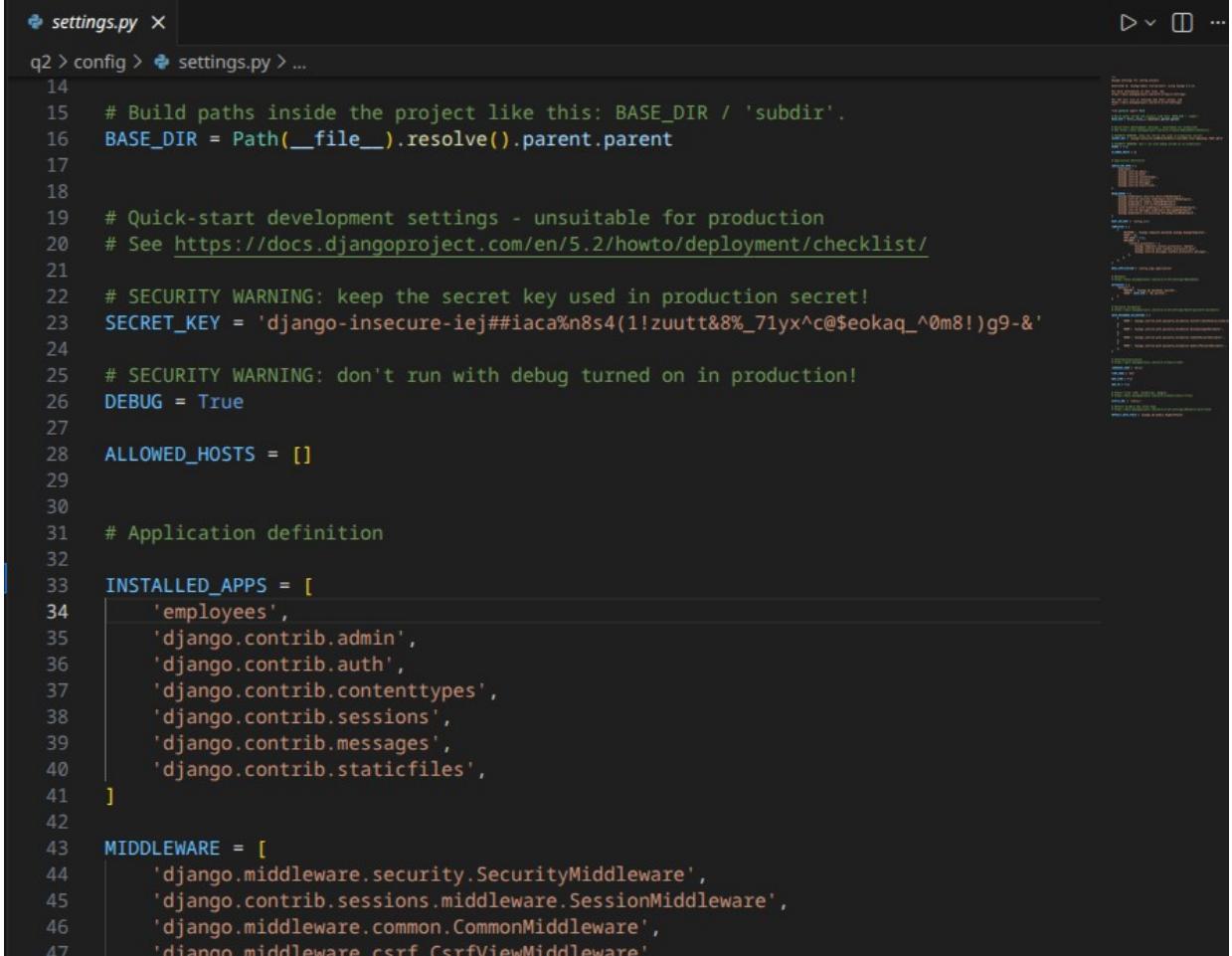
# WPL LAB 5

name: Abhinav Singh Bhagtana

roll no: A-029

reg no: 230905225

Q2.Create a web form with employee ids in dropdown list. Use a textbox for accepting date of joining. Add a button named “Am I Eligible for Promotion”. On clicking the button, if he has more than 5 years of experience, then display “YES” else “NO” in the label control.



The screenshot shows a code editor window with the file 'settings.py' open. The code is a configuration file for a Django application. It defines paths, security settings, application definitions, and middleware. The 'INSTALLED\_APPS' section includes 'employees', which is likely the name of the application being developed. The 'MIDDLEWARE' section lists several middleware classes. The code is color-coded for syntax highlighting, and line numbers are visible on the left.

```
❶ settings.py < ...>
❷ config > settings.py > ...
❸ 14
❹ 15 # Build paths inside the project like this: BASE_DIR / 'subdir'.
❺ 16 BASE_DIR = Path(__file__).resolve().parent.parent
❻ 17
❼ 18
➋ 19 # Quick-start development settings - unsuitable for production
⌃ 20 # See https://docs.djangoproject.com/en/5.2/howto/deployment/checklist/
⌁ 21
⌂ 22 # SECURITY WARNING: keep the secret key used in production secret!
⌃ 23 SECRET_KEY = 'django-insecure-iej##iaca%n8s4(1!zuutt&8%_71yx^c@$eokaq_~0m8!)g9-&'
⌁ 24
⌂ 25 # SECURITY WARNING: don't run with debug turned on in production!
⌃ 26 DEBUG = True
⌁ 27
⌂ 28 ALLOWED_HOSTS = []
⌁ 29
⌂ 30
⌃ 31 # Application definition
⌁ 32
⌂ 33 INSTALLED_APPS = [
⌃ 34     'employees',
⌁ 35     'django.contrib.admin',
⌂ 36     'django.contrib.auth',
⌁ 37     'django.contrib.contenttypes',
⌂ 38     'django.contrib.sessions',
⌁ 39     'django.contrib.messages',
⌂ 40     'django.contrib.staticfiles',
⌁ 41 ]
⌁ 42
⌂ 43 MIDDLEWARE = [
⌃ 44     'django.middleware.security.SecurityMiddleware',
⌁ 45     'django.contrib.sessions.middleware.SessionMiddleware',
⌂ 46     'django.middleware.common.CommonMiddleware',
⌁ 47     'django.middleware.csrf.CsrfViewMiddleware'
```

Urls.py

```
urls.py  X
q2 > config > urls.py > ...
1 """
2     URL configuration for config project.
3
4     The `urlpatterns` list routes URLs to views. For more information please see:
5         https://docs.djangoproject.com/en/5.2/topics/http/urls/
6     Examples:
7         Function based views
8             1. Add an import: from my_app import views
9                 2. Add a URL to urlpatterns: path('', views.home, name='home')
10    Class-based views
11        1. Add an import: from other_app.views import Home
12        2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')
13    Including another URLconf
14        1. Import the include() function: from django.urls import include, path
15        2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
16 """
17 from django.contrib import admin
18 from django.urls import path, include
19
20 urlpatterns = [
21     path('admin/', admin.site.urls),
22     path('employees/', include('employees.urls'))
23 ]
24
```

forms.py

The screenshot shows the Visual Studio Code interface with two tabs open: 'forms.py' and 'check.html'. The left sidebar displays the project structure under 'WEEKS': q2, demo, q1 (selected), q2, .venv, config, employees, \_\_pycache\_\_, migrations, templates, \_\_init\_\_.py, admin.py, apps.py, forms.py (selected), models.py, tests.py, urls.py, views.py, .gitignore, db.sqlite3, and manage.py. The 'forms.py' tab contains the following code:

```
forms.py  X
q2 > employees > forms.py > PromotionForm
1 from django import forms
2
3 class PromotionForm(forms.Form):
4     # Example Employee IDs for the dropdown
5     EMPLOYEE_CHOICES = [
6         ('E101', 'E101'),
7         ('E102', 'E102'),
8         ('E103', 'E103'),
9         ('E104', 'E104'),
10    ]
11
12     employee_id = forms.ChoiceField(choices=EMPLOYEE_CHOICES, label="Select Employee")
13     date_of_joining = forms.DateField(
14         label="Date of Joining",
15         widget=forms.DateInput(attrs={'type': 'date'})
16     )
```

check.html

The screenshot shows a code editor interface with a dark theme. On the left is the file explorer, displaying a project structure for 'WEEKS' containing files like `check.html`, `check.html` (under templates), `employees`, `migrations`, and `views.py`. The main editor area shows the content of `check.html`:

```
<!DOCTYPE html>
<html>
<head>
    <title>Promotion Check</title>
</head>
<body>
    <h2>Employee Promotion Eligibility</h2>

    <form method="post">
        {% csrf_token %}
        {{ form.as_p }}
        <button type="submit">Am I Eligible for Promotion</button>
    </form>

    <hr>

    {% if result %}
        <div style="font-size: 20px;">
            Eligible:
            {% if result == 'YES' %}
                <b style="color: green;">{{ result }}</b>
            {% else %}
                <b style="color: red;">{{ result }}</b>
            {% endif %}

        </div>
    {% endif %}
</body>
</html>
```

Below the editor is a terminal window showing the command `python manage.py runserver` being run in a virtual environment. The terminal output includes a warning about using a development server:

```
WARNING: This is a development server. Do not use it in a production setting. Use a production WSGI or ASGI server instead.
For more information on production servers see: https://docs.djangoproject.com/en/5.2/howto/deployment/
Not Found: /
```

The terminal also lists several log entries from February 4, 2026, at 10:37:21, 10:37:30, 10:41:03, 10:41:12, and 10:41:14, detailing HTTP requests and responses.

website:

### Employee Promotion Eligibility

Select Employee ID:

Date of Joining:

Am I Eligible for Promotion

Eligible: YES

**Q1.** Create a web form which allows the teacher to enter the student details like name, date of birth, address, contact number, email id and marks of English, Physics and Chemistry. After filling all the information and on clicking submit button, details should be added to a textarea displayed on the same page. Display the total percentage of marks obtained in a label.

File Edit Selection View ... ← → Q week5

EXPLORER

WEAKS

- > demo
- ✓ q1
  - > .venv
  - ✓ config
    - > \_\_pycache\_\_
    - ⚙️ \_\_init\_\_.py
    - ⚙️ asgi.py
  - ⚙️ settings.py
  - ⚙️ urls.py
  - ⚙️ wsgi.py
- > students
- > users
- ❖ .gitignore
- ≡ db.sqlite3
- ⚙️ manage.py
- ≡ requirements.txt
- > q2

OUTLINE

TIMELINE

settings.py

q1 > config > settings.py > AUTH\_USER\_MODEL

```
19 # Quick-start development settings - unsuitable for production
20 # See https://docs.djangoproject.com/en/5.2/howto/deployment/checklist/
21
22 # SECURITY WARNING: keep the secret key used in production secret!
23 SECRET_KEY = 'django-insecure-u*9bpknuai-p(xuw6-q)#ew5ti@inqg_rmlriee@2rdiduw1'
24
25 # SECURITY WARNING: don't run with debug turned on in production!
26 DEBUG = True
27
28 ALLOWED_HOSTS = []
29
30 AUTH_USER_MODEL = 'users.CustomUser'
31
32 # Application definition
33
34 INSTALLED_APPS = [
35     'users',
36     'students',
37     'django.contrib.admin',
38     'django.contrib.auth',
39     'django.contrib.contenttypes',
40     'django.contrib.sessions',
41     'django.contrib.messages',
42     'django.contrib.staticfiles',
43 ]
44
45 MIDDLEWARE = [
46     'django.middleware.security.SecurityMiddleware',
47     'django.contrib.sessions.middleware.SessionMiddleware',
48     'django.middleware.common.CommonMiddleware',
49     'django.middleware.csrf.CsrfViewMiddleware',
50     'django.contrib.auth.middleware.AuthenticationMiddleware',
51     'django.contrib.messages.middleware.MessageMiddleware',
52     'django.middleware.clickjacking.XFrameOptionsMiddleware'
53 ]
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

(.venv) VI\_WPL\_A1@cs1-11:~/lab/college/Web Programming Lab/week5/q1\$ python manage.py runserver

Django version 5.2.11, using settings 'config.settings'

Starting development server at http://127.0.0.1:8000/

Quit the server with CONTROL-C.

**WARNING:** This is a development server. Do not use it in a production setting. Use a production WSGI or ASGI server instead.

For more information on production servers see: <https://docs.djangoproject.com/en/5.2/howto/deployment/>

Not Found: /

[04/Feb/2026 10:37:21] "GET / HTTP/1.1" 404 2309

[04/Feb/2026 10:37:30] "GET /students/ HTTP/1.1" 200 1846

Ln 30, Col 1 Spaces: 4 UTF-8 LF () Python 3.11.2 (.venv)

## forms.py:

The screenshot shows a code editor interface with a dark theme. On the left is the Explorer sidebar, which lists project files and folders. In the center is the code editor window, and on the right is a terminal window.

**Explorer:**

- WEEKS
- demo
- q1
- .env
- config
- \_pycache\_
- asgi.py
- settings.py
- urls.py
- wsgi.py
- students
- \_pycache\_
- migrations
- templates
- \_init\_.py
- admin.py
- apps.py
- forms.py
- models.py
- tests.py
- urls.py
- views.py
- users
- .gitignore
- db.sqlite3
- manage.py
- requirements.txt
- q2

**Code Editor (forms.py):**

```
q1 > students > forms.py > StudentForm
1  from django import forms
2
3  class StudentForm(forms.Form):
4      name = forms.CharField(label="Student Name", max_length=100)
5      dob = forms.DateField(label="Date of Birth", widget=forms.DateInput(attrs={'type': 'date'}))
6      address = forms.CharField(label="Address", widget=forms.Textarea(attrs={'rows': 3}))
7      contact = forms.CharField(label="Contact Number", max_length=10)
8      email = forms.EmailField(label="Email ID")
9
10     #marks
11     english = forms.IntegerField(label="English Marks", min_value=0, max_value=100)
12     physics = forms.IntegerField(label="Physics marks", min_value=0, max_value=100)
13     chemistry = forms.IntegerField(label="Chemistry Marks", min_value=0, max_value=100)
```

**Terminal Output:**

```
(.venv) VI_WPL_A1@cs1-11:~/Lab/college/Web Programming Lab/week5/q1$ python manage.py runserver
February 04, 2026 - 10:37:18
Django version 5.2.11, using settings 'config.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.

WARNING: This is a development server. Do not use it in a production setting. Use a production WSGI or ASGI server instead.
For more information on production servers see: https://docs.djangoproject.com/en/5.2/howto/deployment/
Not Found: /
[04/Feb/2026 10:37:21] "GET / HTTP/1.1" 404 2309
[04/Feb/2026 10:37:30] "GET /students/ HTTP/1.1" 200 1846
```

Bottom status bar: Ln 13, Col 86 Spaces: 4 UTF-8 LF {} Python 3.11.2 (.venv)

urls.py

The screenshot shows a code editor interface with a dark theme. On the left is the Explorer sidebar displaying the project structure:

- WEEKS
- q1
- .venv
- config
- \_pycache\_
- \_init\_.py
- asgi.py
- settings.py
- urls.py
- wsgi.py
- students
- \_pycache\_
- migrations
- templates
- \_init\_.py
- admin.py
- apps.py
- forms.py
- models.py
- tests.py
- urls.py
- views.py
- users
- .gitignore
- db.sqlite3
- manage.py
- requirements.txt
- q2

The file `urls.py` is open in the main editor area, showing the following code:

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

urlpatterns = [
    path('', student_form_view, name='student_form')
]
```

Below the editor is a terminal window showing the output of running the development server:

```
python manage.py runserver
February 04, 2026 - 10:37:18
Django version 5.2.11, using settings 'config.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.

WARNING: This is a development server. Do not use it in a production setting. Use a production WSGI or ASGI server instead.
For more information on production servers see: https://docs.djangoproject.com/en/5.2/howto/deployment/
Not Found: /
[04/Feb/2026 10:37:21] "GET / HTTP/1.1" 404 2309
[04/Feb/2026 10:37:30] "GET /students/ HTTP/1.1" 200 1846
```

The status bar at the bottom indicates the current file is `views.py`.

views.py

The screenshot shows a Django project structure in the Explorer sidebar:

- demo
- .venv
- q1
  - .venv
  - config
    - \_\_pycache\_\_
    - \_\_init\_\_.py
    - asgi.py
    - settings.py
    - urls.py
    - wsgi.py
  - students
    - \_\_pycache\_\_
    - migrations
    - templates
    - \_\_init\_\_.py
    - admin.py
    - apps.py
    - forms.py
    - models.py
    - tests.py
    - urls.py
- views.py
- users
- .gitignore
- db.sqlite3
- manage.py
- requirements.txt
- q2

The code editor displays `views.py` with the following content:

```
1 from django.shortcuts import render
2 from .forms import StudentForm
3
4 def student_form_view(request):
5     display_data = ""
6     percentage = 0
7     form = StudentForm()
8
9     if request.method == "POST":
10         form = StudentForm(request.POST)
11         if form.is_valid():
12             data = form.cleaned_data
13             #calculate percentage
14             total = data['english'] + data['physics'] + data['chemistry']
15             percentage = round((total/300) * 100, 2)
16
17             #format data for the TextArea
18             display_data = (
19                 f"Name: {data['name']}\n"
20                 f"DOB: {data['dob']}\n"
21                 f"Address: {data['address']}\n"
22                 f"Contact: {data['contact']}\n"
23                 f"Email: {data['email']}\n"
24                 f"Marks: Eng: {data['english']}, Phys: {data['physics']}, Chem: {data['chemistry']}\n"
25             )
26
27     return render(request, 'students/form.html', {
28         'form':form,
29         'display_data':display_data,
30         'percentage':percentage
31     })
```

The terminal shows the command `python manage.py runserver` being run:

```
(.venv) VI_WPL_A1@escl-11:~/lab/college/Web Programming Lab/week5/q1$ python manage.py runserver
February 04, 2026 - 10:37:18
Django version 5.2.11, using settings 'config.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```

A warning message is displayed at the bottom:

**WARNING: This is a development server. Do not use it in a production setting. Use a production WSGI or ASGI server instead.**  
For more information on production servers see: <https://docs.djangoproject.com/en/5.2/howto/deployment/>

Log entries from the terminal:

```
[04/February 2026 10:37:21] "GET / HTTP/1.1" 404 2309
[04/February 2026 10:37:30] "GET /students/ HTTP/1.1" 200 1846
```

The screenshot shows a VS Code interface with the following components:

- File Explorer (Left):** Shows a tree view of the project structure. The current file, `form.html`, is selected.
- Code Editor (Top Right):** Displays the content of `form.html`. The code includes an H2 header, a form for submitting student details, and a processed details section with a text area and a label for total percentage.
- Terminal (Bottom):** Shows the command `python manage.py runserver` being run in a virtual environment. The output indicates the server is running at `http://127.0.0.1:8000/`.
- Status Bar (Bottom):** Shows the line number (Ln 19), column number (Col 7), and file encoding (UTF-8).

website

# Teacher's Student Entry Portal

Student Name: abhinav

Date of Birth: 01 / 01 / 2000

Address:

MIT Manipal

Contact Number: 123456789

Email ID: abhinav@gmail.com

English Marks: 50

Physics marks: 50

Chemistry Marks: 51

## Processed Details:

Name abhinav  
DOB: 2000-01-01  
Address: MIT Manipal  
Contact: 123456789  
Email: abhinav@gmail.com  
Marks: Eng: 50, Phys: 50, Chem: 51

**Total Percentage: 50.33%**