

CSV FILE

Comma Separated Value

1. PROJECT: CREATE A WEB APPLICATION WITH CSV FILE

1. Create Project and Application

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.16299.1087]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\systech>d:

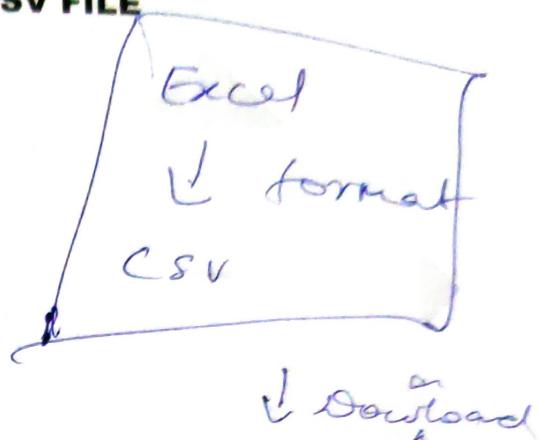
D:\>cd Django

D:\Django>django-admin startproject csvproject1

D:\Django>cd csvproject1

D:\Django\csvproject1>python manage.py startapp csvapp

D:\Django\csvproject1>
```



2. Add application name in settings file.

A screenshot of a Sublime Text editor window. The title bar says 'D:\Django\csvproject1>settings.py - Sublime Text (UNREGISTERED)'. The code in the editor is:

```
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     'django.contrib.admin',
35     'django.contrib.auth',
36     'django.contrib.contenttypes',
37     'django.contrib.sessions',
38     'django.contrib.messages',
39     'django.contrib.staticfiles',
40     'csvapp'
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',
48     'django.contrib.auth.middleware.AuthenticationMiddleware',
49     'django.contrib.messages.middleware.MessageMiddleware',
50     'django.middleware.clickjacking.XFrameOptionsMiddleware',
51 ]
52
53 ROOT_URLCONF = 'csvproject1.urls'
```

3. Create Views file

```
views.py
from django.shortcuts import render
import csv
from django.http import HttpResponse
NAME=['abdul','bala','dinesh','arjun']
AGE=[25,24,26,25,28]
SUB=['tamil','english','maths','science','computer']

def store(request):
    response=HttpResponse(content_type='text/csv')
    response['Content-Disposition']='attachment;filename=file.csv'
    writer=csv.writer(response)
    writer.writerow(['name','age','subject'])
    for (name,age,subject) in zip(NAME,AGE,SUB):
        writer.writerow([name,age,subject])
    return response
```

wordpad / Excel

4. Create url file.

```
urls.py
from django.contrib import admin
from django.urls import path
from csvapp import views

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

name	Age	Sub
Aibani	24	tamil

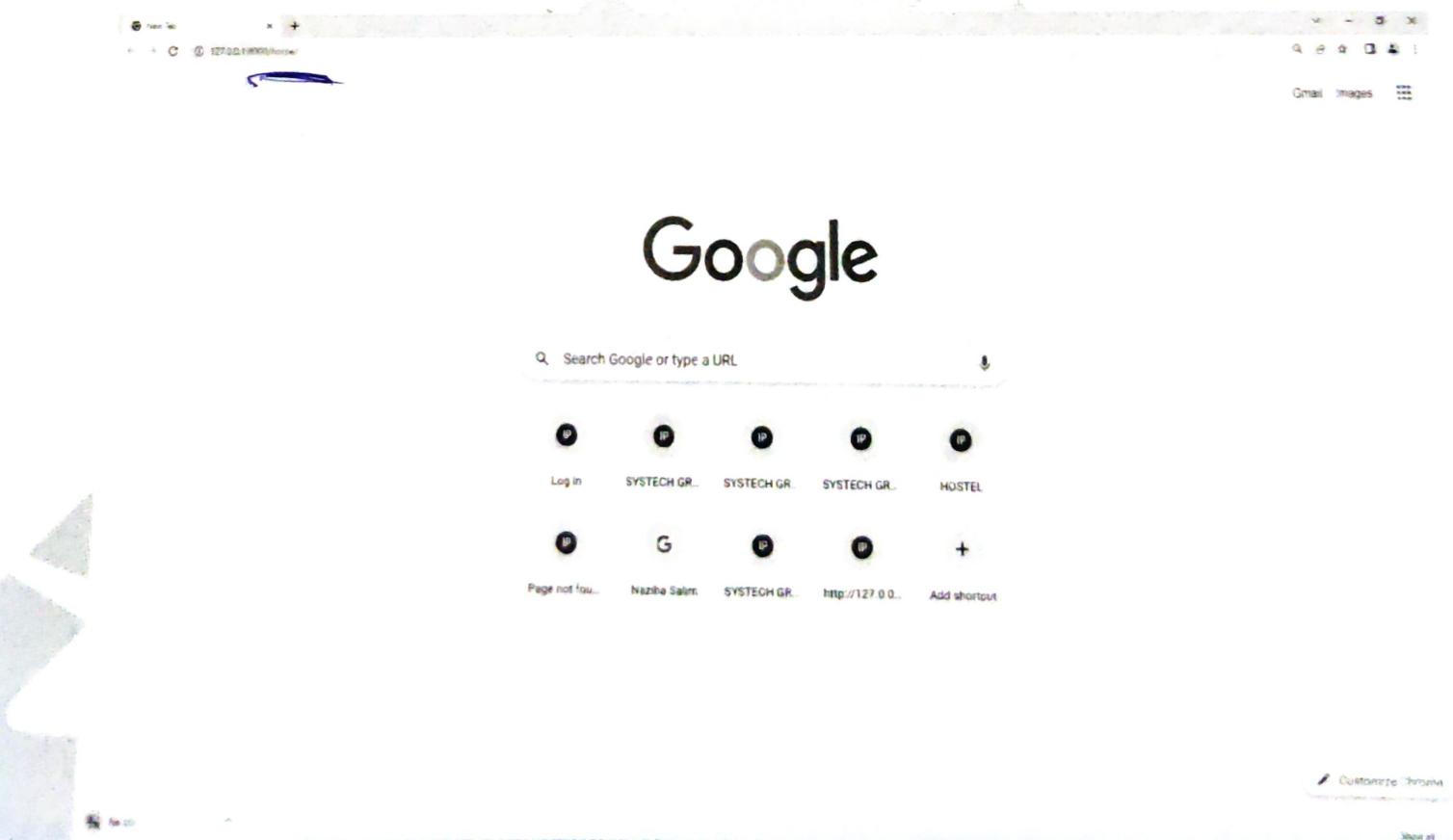
----- WEB DEVELOPMENT -----

5. Type command: **python manage.py runserver**.

```
C:\Windows\system32\cmd.exe - python manage.py runserver
D:\Django\csvproject1>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...
System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work
ons.
Run 'python manage.py migrate' to apply them.
April 23, 2022 - 18:54:42
Django version 4.0.3, using settings 'csvproject1.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

OUTPUT:



WEB DEVELOPMENT

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	name	age	subject													
2	abdul		25	tamil												
3	bala		24	english												
4	dinesh		26	maths												
5	arjun		25	science												
6																
7																
8																
9																
10																
11																
12																
13																
14																
15																
16																
17																
18																
19																
20																
21																
22																
23																

2. PROJECT: CREATE A WEB APPLICATION WITH CSV FILE & DATABASE

1. Create Project and Application

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.16299.1087]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\systech>D:
D:\>cd Django
D:\Django>django-admin startproject csvproject2
D:\Django>cd csvproject2
D:\Django\csvproject2>python manage.py startapp csvapp
D:\Django\csvproject2>
```

2. Add application name in settings file.

The screenshot shows a code editor with the file path 'D:\Django\project2\csvproject2\settings.py'. The code is a Python dictionary defining various settings for a Django application. The 'INSTALLED_APPS' key lists several Django contrib modules and a custom app named 'csvapp'. The 'MIDDLEWARE' key lists several middleware classes. The 'ROOT_URLCONF' key points to 'csvproject2.urls'. The 'TEMPLATES' key specifies a template engine with a 'BACKEND' of 'django.template.backends.django.DjangoTemplates'.

```
28 ALLOWED_HOSTS = []
29
30
31 # Application definition
32
33 INSTALLED_APPS = [
34     'django.contrib.admin',
35     'django.contrib.auth',
36     'django.contrib.contenttypes',
37     'django.contrib.sessions',
38     'django.contrib.messages',
39     'django.contrib.staticfiles',
40     'csvapp'
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',
48     'django.contrib.auth.middleware.AuthenticationMiddleware',
49     'django.contrib.messages.middleware.MessageMiddleware',
50     'django.middleware.clickjacking.XFrameOptionsMiddleware',
51 ]
52
53 ROOT_URLCONF = 'csvproject2.urls'
54
55 TEMPLATES = [
56     {
57         'BACKEND': 'django.template.backends.django.DjangoTemplates',
58     }
59 ]
```

3. Configure Views.py file

The screenshot shows a code editor with the file path 'views.py'. The code defines a function-based view named 'store' that generates a CSV response. It uses the 'render' shortcut from Django's shortcuts module. The function starts by creating a list of product data with variables 'NO', 'NAME', 'PRICE', 'WARRANTY', and 'STOCK'. It then creates an HttpResponse object with 'text/csv' content type and sets the 'Content-Disposition' header to 'attachment;filename=product.csv'. A CSV writer is created for this response, and it writes the header row and then iterates through the product data to write each row. Finally, it returns the response.

```
from django.shortcuts import render
import csv
from django.http import HttpResponseRedirect
NO=[101,102,103,104,105]
NAME=['LAPTOP','MOBILE','MOUSE','PRINTER','SCANNER']
PRICE=[50000,25000,500,35000,15000]
WARRANTY=['3 YEAR','1 YEAR','3 MONTHS','1 YEAR','2 YEAR']
STOCK=[15,20,50,13,25]
def store(request):
    response=HttpResponse(content_type='text/csv')
    response['Content-Disposition']='attachment;filename=product.csv'
    writer=csv.writer(response)
    writer.writerow(['NO','PRODUCTNAME','PRICE','WARRANTY','STOCK'])
    for (a,b,c,d,e) in zip (NO,NAME,PRICE,WARRANTY,STOCK):
        writer.writerow([a,b,c,d,e])
    return response
```

4. Configure url file.

urls.py

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

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

5. Run the server.

```
D:\Django\csvproject2>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
April 23, 2022 - 19:28:58
Django version 4.0.3, using settings 'csvproject2.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
Not Found: /
[23/Apr/2022 19:29:04] "GET / HTTP/1.1" 404 2169
[23/Apr/2022 19:29:11] "GET /home/ HTTP/1.1" 200 177
[23/Apr/2022 19:29:42] "GET /home HTTP/1.1" 301 0
[23/Apr/2022 19:29:42] "GET /home/ HTTP/1.1" 200 177
[23/Apr/2022 19:30:17] "GET /home/ HTTP/1.1" 200 177
[23/Apr/2022 19:30:47] "GET /home/ HTTP/1.1" 200 177
```

6. Verify the result in browser.

The image shows a dual-screen setup. The top screen displays a web browser window with a video player interface. The video player has a play button in the center, surrounded by icons for volume, brightness, and other controls. Below the video player is a search bar with placeholder text "Search Google or type a URL". Underneath the search bar are several small circular icons, likely for social media sharing or quick links. The bottom screen shows a Microsoft Excel spreadsheet titled "product (1) - Microsoft Excel". The spreadsheet contains a table with columns labeled "NO", "PRODUCTNAME", "PRICE", "WARENTY", and "STOCK". The data rows are as follows:

NO	PRODUCTNAME	PRICE	WARENTY	STOCK
1	101 LAPTOP	50000	3 YEAR	15
2	102 MOBILE	25000	1 YEAR	20
3	103 MOUSE	500	3 MONTHS	50
4	104 PRINTER	35000	1 YEAR	13
5	105 SCANNER	15000	2 YEAR	25

WEB DEVELOPMENT

7. Configure models.py file

```
models.py
from django.db import models
class Product(models.Model):
    no=models.IntegerField()
    name=models.CharField(max_length=64)
    price=models.IntegerField()
    warenty=models.CharField(max_length=100)
    stock=models.IntegerField()
```

8. Configure admin.py file

```
admin.py
from django.contrib import admin
from csvapp.models import Product
class ProductAdmin(admin.ModelAdmin):
    list_display=['no','name','price','warenty','stock']
admin.site.register(Product,ProductAdmin)
```

9. Type command: **python manage.py makemigrations**
Python manage.py migrate

```
C:\Windows\system32\cmd.exe

D:\Django\csvproject2>python manage.py makemigrations
Migrations for 'csvapp':
  csvapp\migrations\0001_initial.py
    - Create model Product

D:\Django\csvproject2>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, csvapp, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying csvapp.0001_initial... OK
  Applying sessions.0001_initial... OK
```

10. Type command: **python manage.py createsuperuser**

```
C:\Windows\system32\cmd.exe

D:\Django\csvproject2>python manage.py createsuperuser
Username (leave blank to use 'systech'): admin
Email address: admin@gmail.com
Password:
Password (again):
This password is too short. It must contain at least 8 characters.
This password is too common.
This password is entirely numeric.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.
```

D:\Django\csvproject2>

11. Type command: **python manage.py runserver**.

```
C:\Windows\system32\cmd.exe - python manage.py runserver
D:\Django\csvproject2>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
April 23, 2022 - 19:42:15
Django version 4.0.3, using settings 'csvproject2.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

12. Open admin panel and create some data in database.

The screenshot shows the Django Admin interface for a 'product' model. The top navigation bar includes links for 'Home', 'Logout', and 'View site'. The main title is 'Django administration'. Below the title, there are three tabs: 'Groups', 'Users', and 'Products'. The 'Products' tab is selected, showing a list of three products: 'mouse' (ID 103), 'mobile' (ID 102), and 'laptop' (ID 101). Each product row includes columns for 'NAME', 'PRICE', 'WARRANTY', and 'STOCK'. A total of 3 products are listed. On the right side of the page, there is a 'ADD PRODUCT' button.

NAME	PRICE	WARRANTY	STOCK
mouse	600	6 months	30
mobile	25000	3 years	20
laptop	55000	2 years	20

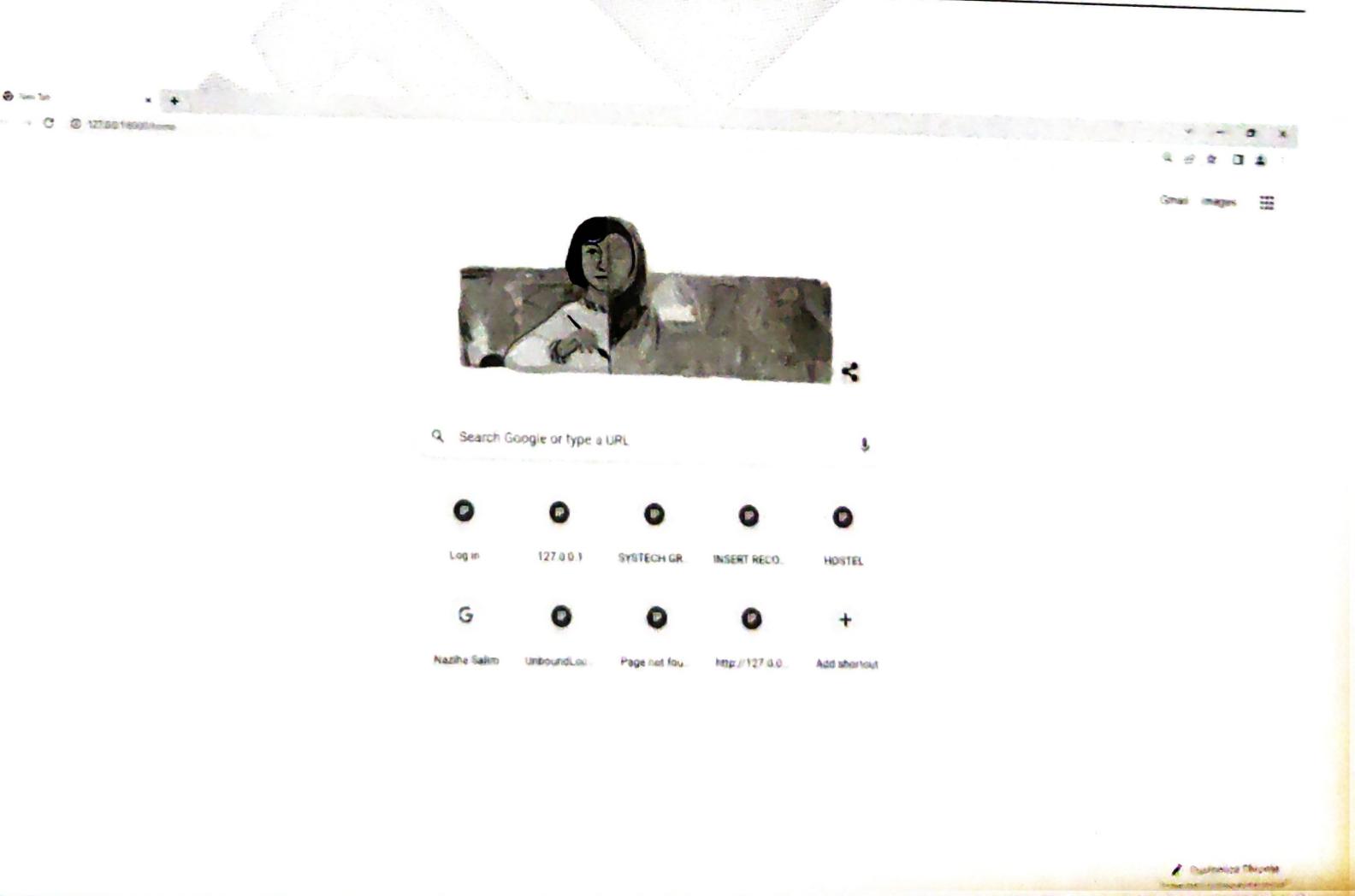
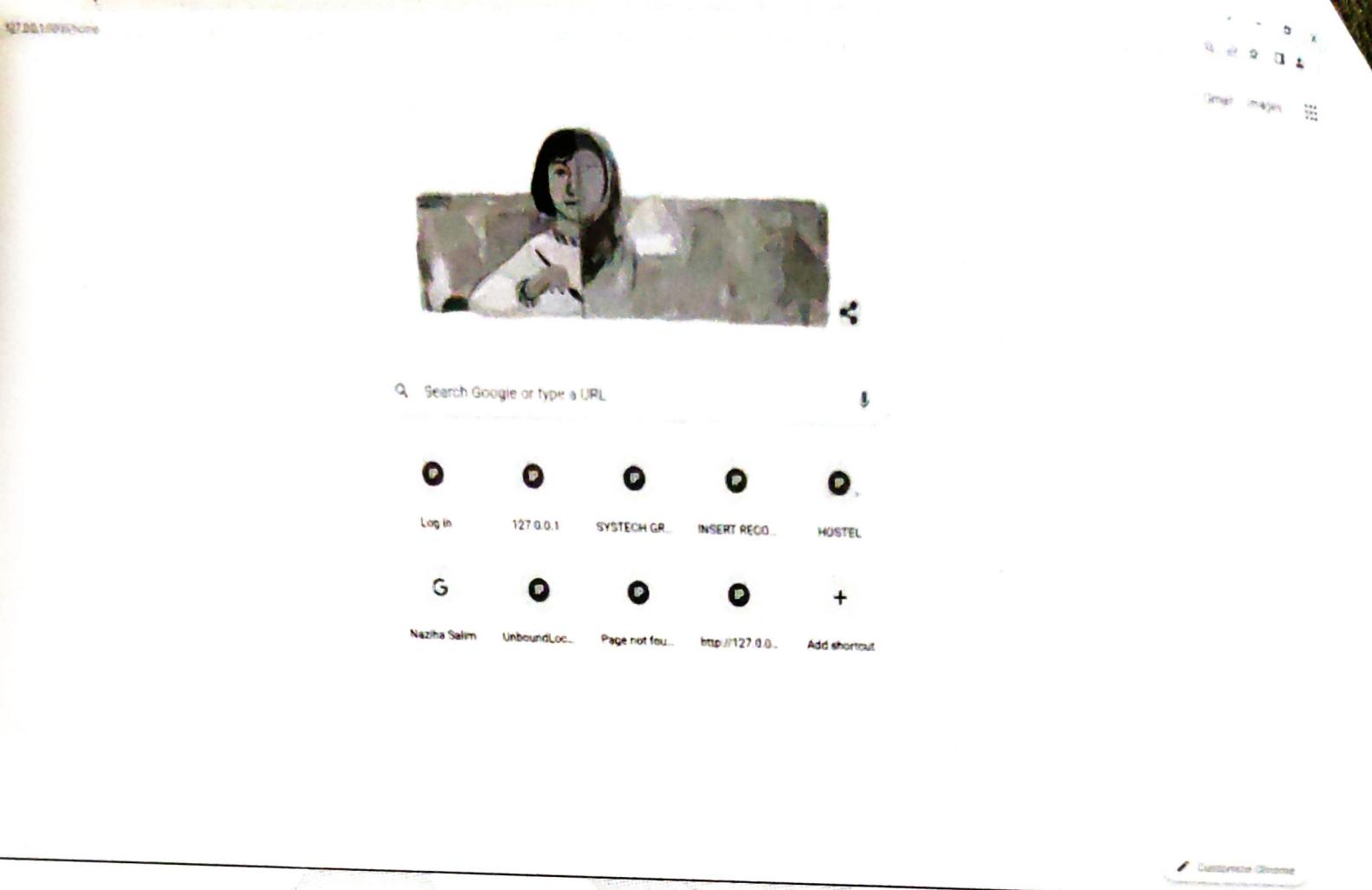
13. Configure views.py file

```
views.py
from django.shortcuts import render
from csvapp.models import Product
import csv
from django.http import HttpResponse
def store(request):
    response=HttpResponse(content_type='text/csv')
    response['Content-Disposition']='attachment;filename=product.csv'
    product=Product.objects.all()
    writer=csv.writer(response)
    writer.writerow(['NO','PRODUCTNAME','PRICE','WARENTY','STOCK'])
    for i in product:
        writer.writerow([i.no,i.name,i.price,i.warranty,i.stock])
    return response
```

14. Type command: python manage.py runserver

```
D:\Django\csvproject2>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
April 23, 2022 - 19:55:19
Django version 4.0.3, using settings 'csvproject2.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```



3. PROJECT: CREATE A WEB APPLICATION WITH CSV FILE & DATABASE

1. Create Project and application.

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.18363.1556]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Reception1>d:

D:\>cd Django

D:\Django>django-admin startproject csvproject3

D:\Django>cd csvproject3

D:\Django\csvproject3>python manage.py startapp csvapp

D:\Django\csvproject3>
```

2. Add Application name in settings file

```
settings.py

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'django-insecure-h%mi)c4x#%e*8*a_dbexmn5w4g8mga5^zcg=@***+'

# 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',
    'csvapp'
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.middleware.locale.LocaleMiddleware',
    'django.middleware.authentication.AuthenticationMiddleware'
]
```

WEB DEVELOPMENT

3. Create database in models file

```
models.py
from django.db import models
class Student(models.Model):
    no=models.IntegerField()
    name=models.CharField(max_length=64)
    age=models.IntegerField()
    city=models.CharField(max_length=100)
```

4. Create admin file

```
admin.py
from django.contrib import admin
from csvapp.models import Student
class StudentAdmin(admin.ModelAdmin):
    list_display=['no','name','age','city']
admin.site.register(Student,StudentAdmin)
```

5. Type command: `python manage.py makemigrations`

```
C:\Windows\system32\cmd.exe
D:\Django\csvproject3>python manage.py makemigrations
Migrations for 'csvapp':
  csvapp\migrations\0001_initial.py
    - Create model Student

D:\Django\csvproject3>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, csvapp, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying csvapp.0001_initial... OK
  Applying sessions.0001_initial... OK
```

6. Type command: `python manage.py createsuperuser.`

```
C:\Windows\system32\cmd.exe
D:\Django\csvproject3>python manage.py createsuperuser
Username (leave blank to use 'reception1'): admin
Email address: admin@gmail.com
Password:
Password (again):
This password is too short. It must contain at least 8 characters.
This password is too common.
This password is entirely numeric.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.

D:\Django\csvproject3>
```

----- WEB DEVELOPMENT -----

7. Type command: `python manage.py runserver`

```
C:\Windows\system32\cmd.exe - python manage.py runserver
D:\Django\csvproject3>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
April 24, 2022 - 17:51:41
Django version 4.0.4, using settings 'csvproject3.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

8. Open admin panel and store the data.

The screenshot shows the Django Admin interface for the 'Students' model. The top navigation bar includes links for 'Home', 'Csvapp', and 'Students'. The title 'Django administration' is displayed. On the left, there's a sidebar with 'AUTHENTICATION AND AUTHORIZATION' sections for 'Groups' and 'Users', and a 'CSVAPP' section with a 'Students' link. The main content area is titled 'Select student to change' and shows a table with three student entries:

Action	NAME	AGE	CITY
<input type="checkbox"/> NC	arjun	24	trichy
<input type="checkbox"/> 103	abdul	23	chennai
<input type="checkbox"/> 101	raj	24	trichy

At the bottom of the table, it says '3 students'. There are also 'ADD STUDENT' and 'Go' buttons at the top right of the table area.

9. Configure the Views file

```
views.py
from django.shortcuts import render
from csvapp.models import Student
import csv
from django.http import HttpResponse
def file(request):
    response=HttpResponse(content_type='text/csv')
    response['Content-Disposition']='attachment;filename=student.csv'
    student=Student.objects.all()
    writer=csv.writer(response)
    writer.writerow(['NO','STUDENT NAME','AGE','CITY'])
    for i in student:
        writer.writerow([i.no,i.name,i.age,i.city])
    return response
```

10. Configure url.py file

```
urls.py
from django.contrib import admin
from django.urls import path
from csvapp import views
urlpatterns = [
    path('admin/', admin.site.urls),
    path('home/',views.file)
]
```

----- WEB DEVELOPMENT -----

11. Type command: `python manage.py runserver`

```
C:\Windows\system32\cmd.exe - python manage.py runserver
```

```
D:\Django\csvproject3>python manage.py runserver
```

```
Watching for file changes with StatReloader
```

```
Performing system checks...
```

```
System check identified no issues (0 silenced).
```

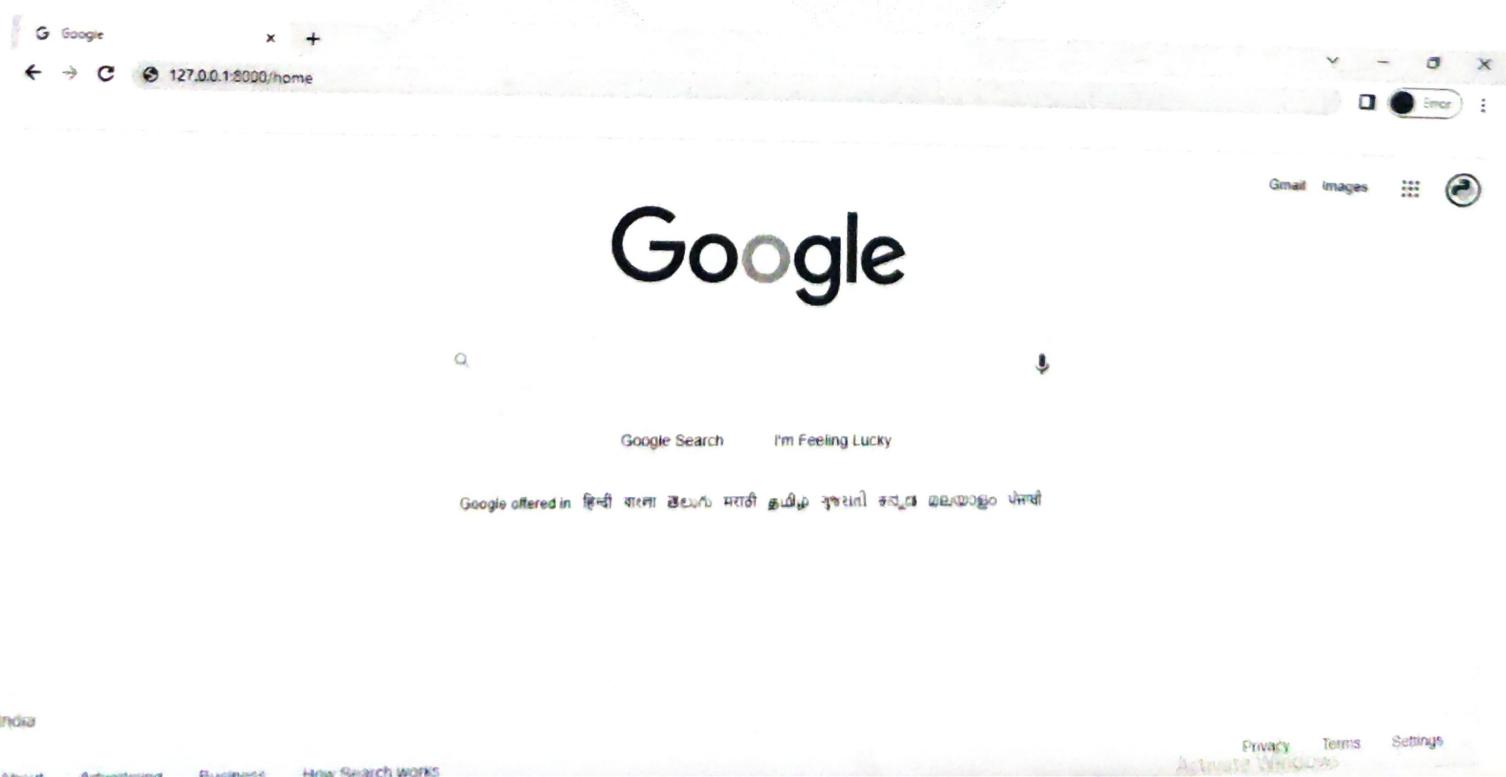
```
April 24, 2022 - 17:55:30
```

```
Django version 4.0.4, using settings 'csvproject3.settings'
```

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

```
Quit the server with CTRL-BREAK.
```

OUTPUT:



WEB DEVELOPMENT

A screenshot of Microsoft Excel showing a table with student data. The table has columns for NO, STUDENT, AGE, and CITY. The data is as follows:

NO	STUDENT	AGE	CITY
1	101 raj	24	trichy
2	102 abdul	23	chennai
3	103 arjun	24	trichy
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			

4. PROJECT: CREATE A WEB APPLICATION WITH CSV FILE & DATABASE

1. Create Project and Application

C:\Windows\system32\cmd.exe

```
D:\Django>django-admin startproject csvproject4
```

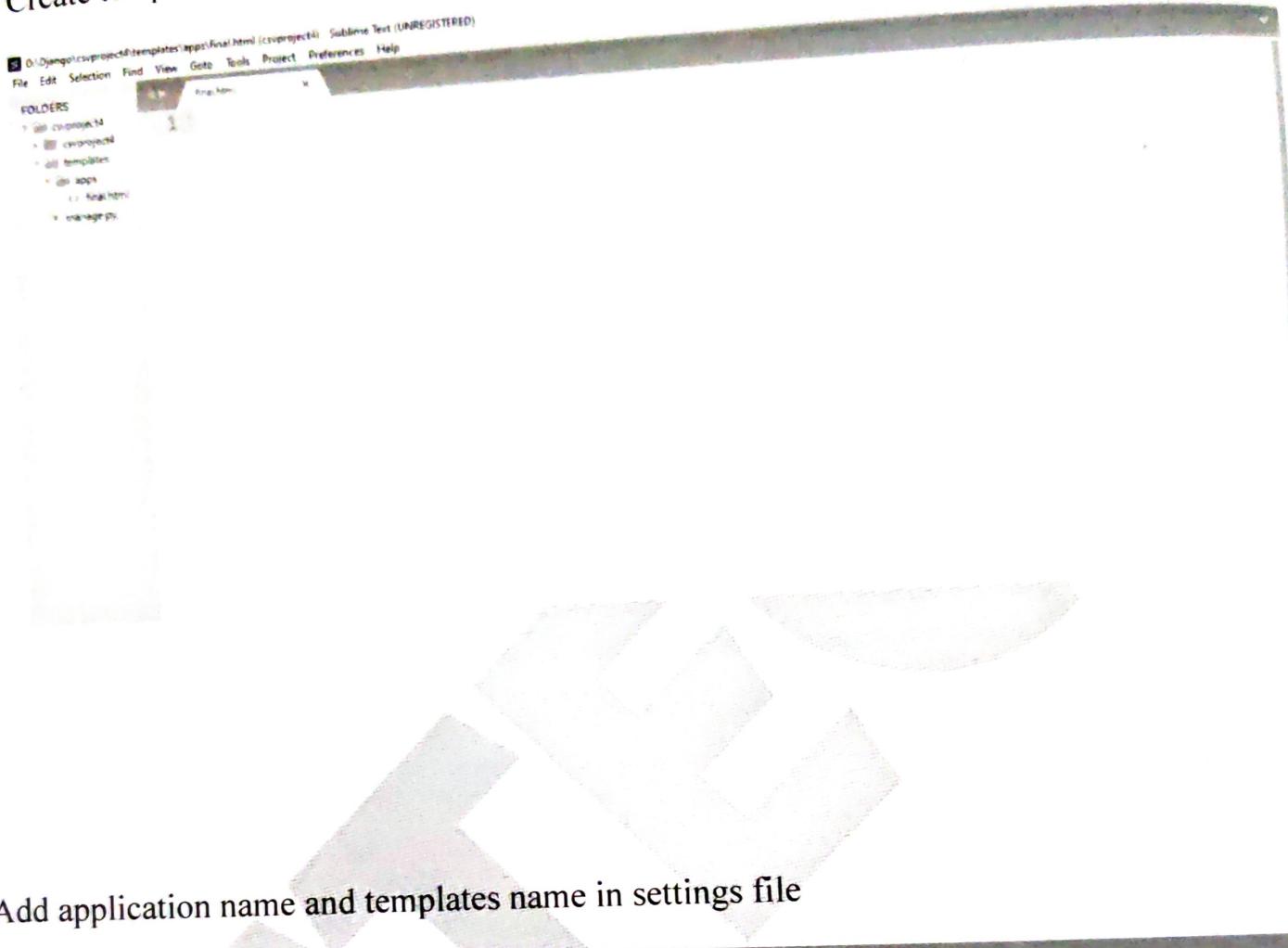
D:\Django>cd csvproject4

```
D:\Django\csvproject4>python manage.py startapp csvapp
```

D:\Django\csvproject4>

WEB DEVELOPMENT

2. Create templates and apps folder and create html files.



3. Add application name and templates name in settings file

```
settings.py
```

```
from pathlib import Path
import os
# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent
T_dir=os.path.join(BASE_DIR,'templates')

# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/4.0/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'django-insecure-rn#r9n%y!cx0$999&t1+e5u60unsgj=oex5lh+-uk-837yd)*'

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

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

```
TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [T_dir],
        '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',
            ],
        },
    },
]
```

4. Create database in models file

```
models.py
from django.db import models
class Bank(models.Model):
    no=models.IntegerField()
    name=models.CharField(max_length=64)
    amount=models.IntegerField()
    city=models.CharField(max_length=100)
```

5. Configure admin.py file

```
admin.py
from django.contrib import admin
from csvapp.models import Bank
class BankAdmin(admin.ModelAdmin):
    list_display=['no','name','amount','city']
admin.site.register(Bank,BankAdmin)
```

6. Type command: python manage.py makemigrations

```
C:\Windows\system32\cmd.exe
D:\Django\csvproject4>python manage.py makemigrations
Migrations for 'csvapp':
  csvapp\migrations\0001_initial.py
    - Create model Bank

D:\Django\csvproject4>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, csvapp, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying csvapp.0001_initial... OK
  Applying sessions.0001_initial... OK
```

----- WEB DEVELOPMENT -----

7. Type command: python manage.py createsuperuser

```
C:\Windows\system32\cmd.exe

D:\Django\csvproject4>python manage.py createsuperuser
Username (leave blank to use 'reception1'): admin
Email address: admin@gmail.com
Password:
Password (again):
This password is too short. It must contain at least 8 characters.
This password is too common.
This password is entirely numeric.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.

D:\Django\csvproject4>
```

8. Type command: python manage.py runserver.

```
C:\Windows\system32\cmd.exe - python manage.py runserver

D:\Django\csvproject4>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
April 24, 2022 - 18:09:31
Django version 4.0.4, using settings 'csvproject4.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

9. Create forms.py

```
forms.py x
from django import forms
from csvapp.models import Bank
class BankForm(forms.ModelForm):
    class Meta:
        model=Bank
        fields='__all__'
```

10. Configure Views.py file

```
views.py x
from django.shortcuts import render
from csvapp.models import Bank
from csvapp.forms import BankForm
import csv
from django.http import HttpResponseRedirect
def final(request):
    bank=Bank.objects.all()
    return render(request,'apps/final.html',{'s':bank})
def forms(request):
    form=BankForm()
    if request.method=="POST":
        form=BankForm(request.POST)
        if form.is_valid():
            form.save()
        return final(request)
    return render(request,'apps/form.html',{'form':form})
def file(request):
    response=HttpResponse(content_type='text/csv')
    response['Content-Disposition']='attachment;filename=account.csv'
    bank=Bank.objects.all()
    writer=csv.writer(response)
    writer.writerow(['NO','NAME','AMOUNT','CITY'])
    for i in bank:
        writer.writerow([i.no,i.name,i.amount,i.city])
    return response
```

11. Configure url.py file

```
urls.py x

from django.contrib import admin
from django.urls import path
from csvapp import views
urlpatterns = [
    path('admin/', admin.site.urls),
    path('final/',views.final),
    path('insert/',views.forms),
    path('home/',views.file)
]
```

12. Create final.html file

```
final.html

<html>
<body>
<center>
<h1>BANK INFORMATION</h1>
{% if s %}
<table border=5px;><thead>
<th>BANK NO</th>
<th>NAME</th>
<th>AMOUNT</th>
<th>CITY</th></thead>
{% for x in s %}
<tr>
<td>{{x.no}}</td>
<td>{{x.name | title}}</td>
<td>{{x.amount}}</td>
<td>{{x.city}}</td></tr>
{% endfor %}
{% else %}
<p>NO RECORDS FOUND</p>
{% endif %}
</center></body>
</html>
```

----- WEB DEVELOPMENT -----

13. Create form.html file

```
form.html
<!DOCTYPE html>
<html>
<head>
    <title>INSERT RECORD</title>

</head>
<center>
<body>
<h2>INSERT BANK RECORD</h2>
<form method="POST">
    {{form.as_p}}
    {%csrf_token%}
    <input type="submit" value="SUBMIT">
</form>
</body>
</center>
</html>
```

14. Type command: python manage.py runserver.

```
C:\Windows\system32\cmd.exe - python manage.py runserver
```

```
D:\Django\csvproject4>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
April 24, 2022 - 18:44:44
Django version 4.0.4, using settings 'csvproject4.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

OUTPUT:

The screenshot shows a web browser window with the URL `127.0.0.1:8000/insert/`. The page title is "INSERT BANK RECORD". It contains four input fields: "No." with value "2022104", "Name" with value "leo", "Amount" with value "40000", and "City" with value "trichy". A "SUBMIT" button is at the bottom.

The screenshot shows a web browser window with the URL `127.0.0.1:8000/final/`. The page title is "BANK INFORMATION". It displays a table with the following data:

BANK NO	NAME	AMOUNT	CITY
2022101	Raj	20000	trichy
2022102	Arjun	30000	chennai
2022104	Leo	40000	trichy

BANK INFORMATION

BANK NO	NAME	AMOUNT	CITY
2022101	Raj	20000	trichy
2022102	Arjun	30000	chennai
2022104	Leo	40000	trichy

BANK INFORMATION

BANK NO	NAME	AMOUNT	CITY
2022101	Raj	20000	trichy
2022102	Arjun	30000	chennai
2022104	Leo	40000	trichy

A screenshot of Microsoft Excel showing a table of bank information. The table has four columns: NO, NAME, AMOUNT, and CITY. The data is as follows:

NO	NAME	AMOUNT	CITY
2022101	raj	20000	trichy
2022102	arjun	30000	chennai
2022104	leo	40000	trichy

5. PROJECT: CREATE A WEB APPLICATION WITH CSV FILE & DATABASE

1. Create Project and Application

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.18363.1556]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Reception1>d:
D:\>cd Django
D:\Django>django-admin startproject csvprojects
D:\Django>cd csvprojects
D:\Django\csvproject5>python manage.py startapp csvapp
D:\Django\csvproject5>
```

2. Add Application and templates name in settings file

```
settings.py

from pathlib import Path
import os
# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent
T_dir=os.path.join(BASE_DIR,'templates')

# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/4.0/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'django-insecure-2f$^wt39h+bt#qxp8r^7x-738*0^f6npn-gvo-a6%9q_h5e^3z'

# 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',
    'csvapp'
]
```

----- WEB DEVELOPMENT -----

```
TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [T_dir],
        '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',
            ],
        },
    },
]
```

3. Create database in models file

models.py

x

```
from django.db import models
class Office(models.Model):
    no=models.IntegerField()
    name=models.CharField(max_length=64)
    mobile=models.IntegerField()
    city=models.CharField(max_length=100)
    course=models.CharField(max_length=100)
```

4. Configure admin.py file

```
admin.py
from django.contrib import admin
from csvapp.models import Office
class OfficeAdmin(admin.ModelAdmin):
    list_display=['no','name','mobile','city','course']
admin.site.register(Office,OfficeAdmin)
```

5. Type command: python manage.py makemigrations

```
C:\Windows\system32\cmd.exe
D:\Django\csvproject5>python manage.py makemigrations
Migrations for 'csvapp':
  csvapp\migrations\0001_initial.py
    - Create model Office

D:\Django\csvproject5>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, csvapp, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying csvapp.0001_initial... OK
  Applying sessions.0001_initial... OK
```

WEB DEVELOPMENT

6. Type command: `python manage.py createsuperuser`

C:\Windows\system32\cmd.exe

```
D:\Django\csvproject5>python manage.py createsuperuser
Username (leave blank to use 'reception1'): admin
Email address: admin@gmail.com
Password:
Password (again):
This password is too short. It must contain at least 8 characters.
This password is too common.
This password is entirely numeric.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.
```

D:\Django\csvproject5>

7. Create `forms.py` file

forms.py

```
from django import forms
from csvapp.models import Office
class OfficeForm(forms.ModelForm):
    class Meta:
        model=Office
        fields='__all__'
```

WEB DEVELOPMENT

8. Configure Views file

```
views.py
from django.shortcuts import render
from csvapp.models import Office
from csvapp.forms import OfficeForm
import csv
from django.http import HttpResponseRedirect
def final(request):
    office=Office.objects.all()
    return render(request,'apps/final.html',{'s':office})
def forms(request):
    form=OfficeForm()
    if request.method=="POST":
        form=OfficeForm(request.POST)
        if form.is_valid():
            form.save()
    return final(request)
    return render(request,'apps/form.html',{'form':form})
def file(request):
    response=HttpResponse(content_type='text/csv')
    response['Content-Disposition']='attachment;filename=enquiry.csv'
    office=Office.objects.all()
    writer=csv.writer(response)
    writer.writerow(['NO','NAME','MOBILE','CITY','COURSE'])
    for i in office:
        writer.writerow([i.no,i.name,i.mobile,i.city,i.course])
    return response
def page(request):
    return render(request,'apps/page.html')
```

9. Configure url file

```
urls.py
from django.contrib import admin
from django.urls import path
from csvapp import views
urlpatterns = [
    path('admin/', admin.site.urls),
    path('final/',views.final),
    path('form/',views.forms),
    path('home/',views.file),
    path('index/',views.page)
]
```

10. Create index.html file

```
index.html x  
-----  
|<!DOCTYPE html>  
<html>  
<head>  
  <title>SYSTECH GROUP</title>  
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhFlvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBqy12QvZ6jIW3" crossorigin="anonymous">  
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-ka7Sk0Gln4gmtz2MlQnkt1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p" crossorigin="anonymous"></script>  
<nav class="navbar navbar-dark bg-dark">  
  <div class="container-fluid">  
    <a class="navbar-brand" href="/index">SYSTECH GROUP</a>  
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">  
      <span class="navbar-toggler-icon"></span>  
    </button>  
    <div class="collapse navbar-collapse" id="navbarNav">  
      <ul class="navbar-nav">  
        <li class="nav-item">  
          <a class="nav-link" href="/final">REPORT</a>  
        </li>  
        <li class="nav-item">  
          <a class="nav-link" href="/form">UPDATE</a>  
        </li>  
      </ul>  
    </div>  
  </div>  
</nav>
```

Activate Windows
Go to Settings to activate Wi-Fi

```
index.html
  x
</li>
<li class="nav-item">
  <a class="nav-link" href="/home">DOWNLOAD</a>
</li>
</ul>
</div>
</div>
</head>
<body style="background-color: #a6ffff">WELCOME TO SYSTECH GROUP </h1>
  <h1 style="text-align: center; size: 20; color: red">WELCOME TO SYSTECH GROUP </h1>
</body>
<footer style="position: fixed; padding: 10px 10px 0px 10px; bottom: 0; width: 100%; height: 40px; background: #36353b">
  <a href="http://www.systechgroup.in" style="text-align: right; font-size: large; color: white">www.systechgroup.in</a>
</footer>
{% block body_block %}
{% endblock %}
</html>
```

11. Create page.html file

```
page.html
  •
<!DOCTYPE html>
{% extends 'apps/index.html' %}
{% block body_block %}
<html>

<body style="background-color: red">
<main>
  <h1 style="color: red; font-size: xx-large; font-family: arial">Hi Everyone </h1>
  <p style="color: blue; font-size: x-large; text-align: justify;">Systech is a leading
  Training Development and Placement company in Trichy. We are known for our practical
  approach towards training that enables students to gain real-time exposure to competitive
  technologies. Training is offered by employees from MNCs to give real corporate exposure to
  the students.</p>
  <article>
    <h2 style="color: green; font-size: xx-large; font-family: arial">DJANGO WEB DEVELOPMENT
    COURSE</h2>
    <p style="color: blue; font-size: x-large; text-align: justify;">Django is a Web
    Development course provided by systech Trichy. It is the best framework for developing a
    Web application framework in Python programming language. Django follows Model View
    Template (MVT) Architecture which is based on the Model View Controller(MVC)
    Architecture. The demand for Django remains is high at all times because of its fast
    development feature. It can be used to build any type of website and web application.</p>
  </article>
</main>
</body>
{% endblock %}
</html>
```

Activate Windows
Go to Settings to activate

12. Create final.html file

```
final.html
{x% extends 'apps/index.html'%}
{block body_block%}
<html>
<body>
<center>
<h1>ENQUIRY INFORMATION</h1>
{if s%}
<table border=5px;><thead>
<th>NO</th>
<th>NAME</th>
<th>MOBILE</th>
<th>CITY</th>
<th>COURSE</th></thead>
{for x in s%}
<tr>
<td>{{x.no}}</td>
<td>{{x.name | title}}</td>
<td>{{x.mobile}}</td>
<td>{{x.city}}</td>
<td>{{x.course}}</td></tr>
{endfor%}</table>
{else%}
<p>NO RECORDS FOUND</p>
{endif%}
</center></body>
{endblock%}
</html>
```

13. Create form.html file

```
form.html
{x% extends 'apps/index.html'%}
{block body_block%}
<!DOCTYPE html>
<html>
<head>
    <title>INSERT RECORD</title>
</head>
<center>
<body>
<h2>INSERT ENQUIRY RECORD</h2>
<form method="POST">
    {{form.as_p}}
    {%csrf_token%}
    <input type="submit" value="SUBMIT">
</form>
</body>
</center>
{endblock%}
</html>
```

14. Type command: `python manage.py runserver`

```
C:\Windows\system32\cmd.exe - python manage.py runserver
```

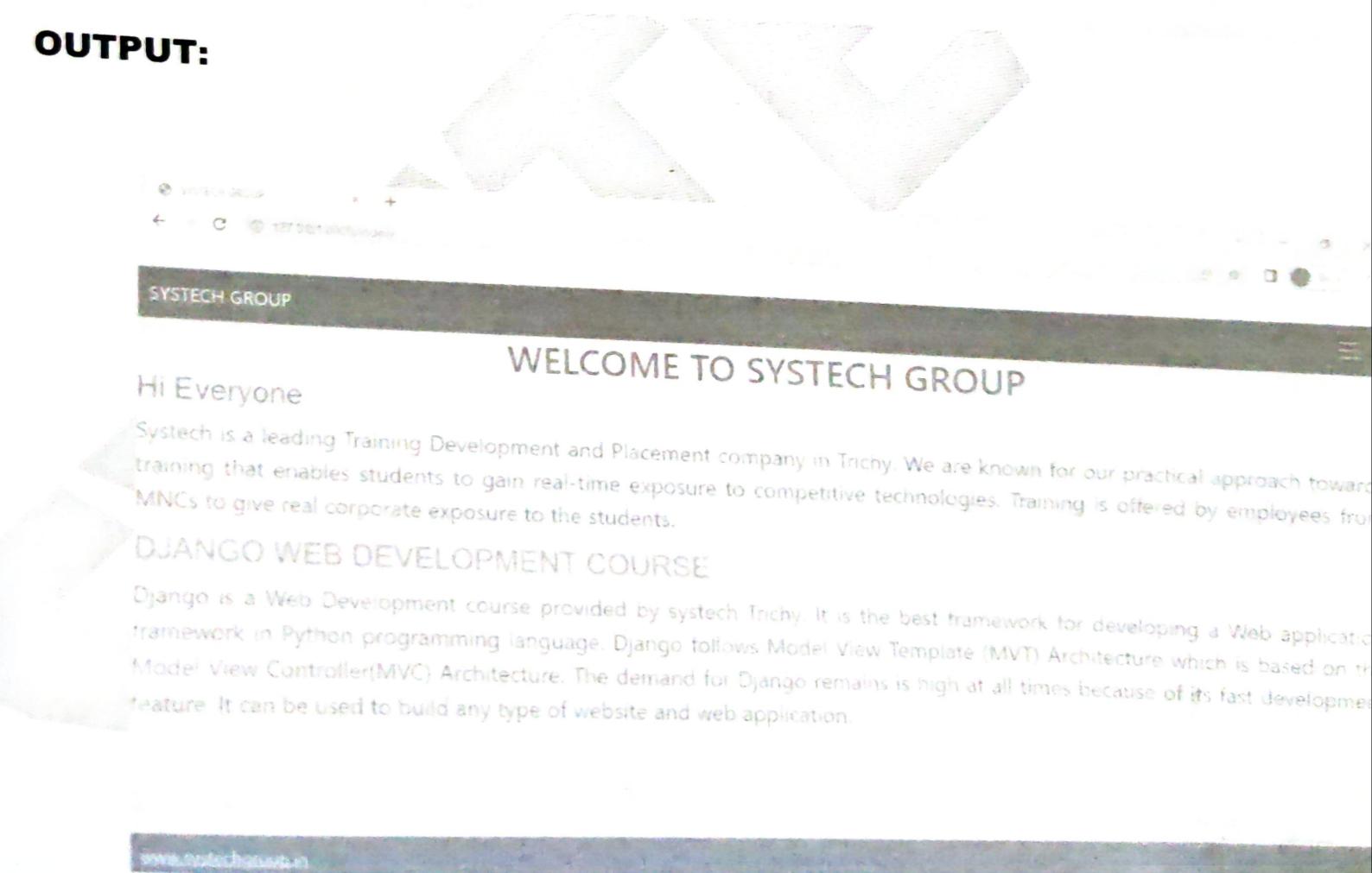
```
D:\Django\csvproject5>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...
```

```
System check identified no issues (0 silenced).
```

```
April 24, 2022 - 19:56:07
```

```
Django version 4.0.4, using settings 'csvproject5.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

OUTPUT:



www.sysTechgroup.in

SYSTECH GROUP

WELCOME TO SYSTECH GROUP ENQUIRY INFORMATION

NO	NAME	MOBILE	CITY	COURSE
1	John	9899999999	Mumbai	Java
2	Ashish	9899999999	Mumbai	Java

www.sysTechgroup.in

SYSTECH GROUP

WELCOME TO SYSTECH GROUP INSERT ENQUIRY RECORD

Name:	<input type="text"/>
Mobile:	<input type="text"/>
City:	<input type="text"/>
Course:	<input type="text"/>
GMAIL:	

WEB DEVELOPMENT

File Edit View Insert Page Layout Formulas Data Review View

WELCOME TO SYSTECH GROUP
INSERT ENQUIRY RECORD

No. _____
Name _____
Mobile _____
City _____
Course _____

SUBMIT

www.systechgroup.in

File Home Insert Page Layout Formulas Data Review View

Cut Copy Paste Format Painter

Font Alignment Number Styles Cells Editing

A1	NO	NAME	MOBILE	CITY	COURSE
1	1 raj		9849494939	trichy	python
2	102 arjun		9820399409	chennai	django
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					