TRAINING DAY8 REPORT

02 JULY 2025

What Are Login and Registration Forms?

LOGIN FORM:

A login form allows users to access a website or application by entering valid credentials (usually username/email and password).

REGISTRATION FORM:

A registration form allows new users to create an account by providing personal details such as name, email, password, etc.

Both are essential components of user authentication systems.

Purpose and Importance

Form	Purpose
Login	Authenticates users and grants access to protected resources.
Register	Collects user data and creates a new account in the database.

COMMON FIELDS

Login form fields:

1. Email or Username – used to identify the user.
2. Password – used for authentication.

Registration form fields:

1. Full Name
2. Username
3. Email
4. Password
5. Confirm Password
6. Optional: phone number, profile photo, address, etc.

Working Mechanism

Registration Process

- User fills in details and submits the form.
- Form validates required fields (e.g., unique email, matching passwords).
- If valid, data is saved in the user database table.
- User may be logged in automatically or redirected to the login page.

Login Process

- User submits email/username and password.
- Server checks if credentials match any existing user.
- If valid, a user session or token is created.
- User is redirected to the dashboard or home page.

Security Measures

Feature	Description
CSRF Protection	Prevents cross-site request forgery attacks.
Password Hashing	Passwords are encrypted before storage.
Input Validation	Prevents SQL injection or malicious inputs.
Login Throttling	Limits repeated login attempts to prevent brute force.

HTTP Method Used

- Both login and registration forms use the POST method.
- POST sends data securely in the request body, not in the URL.

Response Scenarios

Scenario	Action
Valid Login	Redirect to profile/dashboard
Invalid Login	Show error: "Invalid credentials"
Successful Register	Redirect to login page or auto-login
Form Errors	Display messages: "Email already exists", "Passwords do not match"

What is Authentication?

Authentication is the process of verifying the identity of a user. It ensures that only registered and verified users can access protected areas of a website or web application.

Example:

- A user enters their username and password to log in.
- If credentials are valid, the system authenticates the user and gives access.

Once authenticated, the user is logged in and associated with a session.

What is a Session?

A session is a way to store user-specific data across multiple requests in a web application.

Sessions start when a user logs in or visits the site.

Data stored in the session persists until:

- The session expires.
- The user logs out.
- The session is manually cleared.

How Sessions Work in Django

- A session ID is created and sent to the user's browser as a cookie.

- Django stores this ID and related data in the server (usually the database).
- On every request, the session ID is used to retrieve the stored data.

Enabling Sessions in Django

Sessions are enabled by default in Django. Make sure the following is present in your settings.py:

```
INSTALLED_APPS = [
    ...
    'django.contrib.sessions',
]

MIDDLEWARE = [
    ...
    'django.contrib.sessions.middleware.SessionMiddleware',
]
```

Using Sessions in Views

```
def set_session(request):
    request.session['username'] = 'saarvi'
    request.session['is_logged_in'] = True
    return HttpResponse("Session data set successfully!")
```

What it does:

- Stores the user's name and login state in session variables.
- These values are kept on the server side.