Practical Day 26- Day 27

1. Configuring a Public IP Address

- Obtain a public IP from your ISP or use a cloud provider like AWS, DigitalOcean, or Linode.
- Assign the public IP to a server (Linux/Windows) either through your router's NAT configuration or directly on the machine.
- Edit network configuration files:
    - For Linux (e.g., Ubuntu):
        - bash
        - `sudo nano /etc/netplan/01-netcfg.yaml`
        - Set static IP, subnet mask, gateway, and DNS.
- Test external connectivity:
    - Use `ping`, `curl`, or visit `https://whatismyipaddress.com` to verify.
- Configure router/firewall to allow traffic on relevant ports (e.g., 80 for HTTP).

2. Installing Apache Server

1. Install Apache:
2. Start & enable the service:
3. Test the server:
    a. Open your browser and go to `http://your-public-ip` – you should see the Apache welcome page.
4. Configure the firewall
5. Basic web deployment:
    a. Drop your HTML files in `/var/www/html` or configure virtual hosts in `/etc/apache2/sites-available/`.

3. Installing and Configuring DBMS (MySQL, PostgreSQL)

- Download and install MySQL and PostgreSQL on Windows/Linux environments.
- Set up database instances using mysql_secure_installation and PostgreSQL's initdb.
- Configure network access in MySQL (my.cnf) and PostgreSQL (postgresql.conf + pg_hba.conf).
- Start and stop database services using systemctl or service managers.
- Connect to the DBMS using command-line tools (mysql, psql) and GUI tools (e.g. MySQL Workbench, pgAdmin).

4. Understanding Database Structures (Schemas, Tables, Queries)

- Create and visualize schemas using ERD diagrams or SQL tools.
- Define tables with data types, primary and foreign keys.
- Establish relationships (one-to-many, many-to-many) through foreign keys.
- Normalize tables to reduce redundancy and improve integrity.

- Interpret sample database structures and describe their logic.

5. Backing Up and Restoring Databases (mysqldump + Cron)
    - Back up a MySQL database using: mysqldump -u