Day_3 Assignment

Overview

On Day 3, we were introduced to **Amazon Web Services (AWS)**, which is a cloud platform offering computing resources, storage, databases, and various other services over the internet. We learned about its importance, usage, and how to set it up.

What is AWS?
Amazon Web Services (AWS) is a secure cloud services platform that provides compute power, database storage, content delivery, and other functionality to help businesses scale and grow.

Why Use AWS?
- Scalability
- Cost-efficiency (pay-as-you-go model)
- High availability and reliability
- Wide range of services (like EC2, S3, RDS, etc.)

How to Create an AWS Account?
1. Visit https://aws.amazon.com
2. Click on **Create an AWS Account**
3. Fill in details including email, password, and account name
4. Enter billing information and verify identity
5. Choose a support plan and complete registration

Creating a New User on AWS
1. Go to **IAM (Identity and Access Management)** service
2. Click on **Users > Add user**
3. Provide username, select programmatic access and/or AWS Management Console access
4. Set permissions (e.g., AdministratorAccess or custom)
5. Review and create user to get **Access Key ID** and **Secret Access Key**

Launching an Ubuntu Machine (EC2 Instance)

1. Go to **EC2 Dashboard**
2. Click on **Launch Instance**
3. Select **Ubuntu AMI**
4. Choose an instance type (e.g., t2.micro - Free tier)
5. Configure and launch with a new or existing key pair

Hosting a Static Website on AWS Ubuntu EC2

Step 1: Connect to the Instance using SSH

Step 2: Host a Basic HTML Page

```bash
sudo apt update -y                    # Update package list
sudo apt install apache2 -y           # Install Apache web server
sudo systemctl start apache2          # Start Apache service
sudo systemctl status apache2         # Check Apache service status
sudo chmod -R 777 /var/www/html/      # Give write permission to HTML directory
cd /var/www/html/                     # Navigate to web root
sudo echo "Hey Ashish" > index.html   # Create a basic HTML file
```

Step 3: Host a Website from FreeCSS

```bash
wget [URL]                            # Download website template zip from free-css.com
unzip filename.zip                    # Unzip the downloaded file
mv /path/to/website/* /var/www/html/  # Move website content to Apache root directory
```

Linux Commands Used and Their Meanings

Command	Description
`ls -lh`	Lists files with human-readable sizes
`truncate -s 250M data.txt`	Creates a file of size 250MB named data.txt

`tar -czf regex.tar.gz data.txt`	Compresses data.txt into a tar.gz archive
`sudo apt update -y`	Updates package lists automatically with 'yes' to prompts
`sudo apt install apache2 -y`	Installs the Apache2 web server
`sudo systemctl start apache2`	Starts Apache web server service
`sudo systemctl status apache2`	Shows Apache server status
`sudo chmod -R 777 /var/www/html/`	Gives all users full access to the HTML directory
`cd /var/www/html/`	Navigates to Apache's web root directory
`sudo echo "Hey Ashish" > index.html`	Creates an index.html file with "Hey Ashish" text
`wget [URL]`	Downloads a file from the specified URL
`unzip filename.zip`	Extracts contents of a zip file
`mv /path/* /var/www/html/`	Moves all website files to the Apache web root

Conclusion

This session provided a hands-on introduction to AWS, user management, launching EC2 instances, and hosting static websites on Ubuntu using Apache.