full one-shot videos on :**JK Coding Pathshala YouTube channel**

JK Coding Pathshala

https://youtube.com/@jayeshkande9215?feature=shared

Unit VI	WEB APPLICATION DEPLOYMENT	(06 hrs)
Cloud: AWS Cloud, AWS Elastic Compute, AWS Elastic Load Balancer and its types, AWS VPC and Component of VPC, AWS storage, Deploy Website or Web Application on AWS, Launch an Application with AWS Elastic Beanstalk.		

Q7) a) What is EC2 service? Explain steps to deploy website on EC2. **[9]**

b) What is AWS cloud? What are the services provided by AWS? Explain any two in brief. **[8]**

OR

Q8) a) What is ELB? What are the ELB types? List advantages of ELB **[9]**

b) What is VPC? What are the components of VPC? **[8]**

Q7) a) What is VPC? Explain the components of VPC. **[6]**

b) What is cloud computing? What are the benefits of cloud computing? **[6]**

c) List and explain the steps to deploy the application on the elastic beanstalk. **[5]**

OR

Q8) a) What are the storage services provided by AWS? **[6]**

b) What is S3 bucket and how to create a bucket? **[6]**

c) What is PuTTY? How to connect the EC2 instance with PuTTY? **[5]**

Q7)	a)	Explain the features and benefits of AWS Cloud.	**[6]**
	b)	What is S3 bucket and how to create a bucket?	**[6]**
	c)	Explain the different components of VPC?	**[5]**

OR

Q8)	a)	What is PuTTY? How to connect EC2 instance with PuTTY?	**[6]**
	b)	Explain the process of deploying a website or web application on AWS.	**[6]**
	c)	What is Elastic Load Balancer and explain its working.	**[5]**

Q7) a) What is EC2 service? Explain Steps to deploy website on EC2. **[9]**

b) What is AWS cloud? What are the services provided by AWS? Explain any two in brief. **[8]**

OR

Q8) a) What is ELB? What are the ELB types? List advantages of ELB. **[9]**

b) What is VPC? What are the components of VPC? **[8]**

Q7)	a)	Explain in detail types of Elastic Load Balancer.	**[6]**
	b)	What are the different components of VPC?	**[5]**
	c)	What is elastic beanstalk and enlist the advantages of using it?	**[6]**

OR

Q8)	a)	Explain any three AWS Storage services.	**[6]**
	b)	What is Elastic Load Balancer and explain its working.	**[5]**
	c)	What is AWS cloud? List different services provided by it.	**[6]**

Cloud Kya Hai?
Cloud ka matlab yahan aasman wala cloud nahi hai 😄
Yahan **cloud** ka matlab hai **internet ke through data aur services ka access**.

Cloud Computing Kya Hai?

Definition:

Cloud Computing ek aisi technology hai jisme data, applications, aur services internet ke through access ki jaati hain — bina apne computer ya server ke.

Example:

Google Drive, Netflix, Gmail, YouTube = sab cloud ka use karte hain.

Types of Cloud Computing

1.☁️ **Public Cloud** – Shared by everyone (e.g., Google Cloud)

2. **Private Cloud** – Sirf ek organization ke liye

3.🌐 **Hybrid Cloud** – Public + Private ka mix

Cloud Computing Ke Fayde (Benefits)

1. **Cost Saving** – Server ya hardware ka kharcha nahi
2. **Scalability** – Use ke hisaab se increase ya decrease
3. **Accessibility** – Kahin se bhi access
4. **Security** – Data secure hota hai
5. **Backup & Recovery** – Data automatically safe
6. **Collaboration** – Team ke saath easy kaam

Challenges of Cloud Computing

1. Internet Dependence – Internet na ho to problem
2. Data Privacy – Kabhi-kabhi data leak ka risk
3. Ongoing Cost – Monthly ya yearly payment

Real Life Examples

- Google Drive – Files online store karna
- Gmail – Emails cloud par store
- Netflix – Movies cloud se stream hoti hain
- Instagram – Photos cloud me save hoti hain

AWS Cloud

☁️ What is AWS Cloud?

AWS (Amazon Web Services) ek cloud computing platform hai jo **Amazon** ne banaya hai.
Ye duniya ka **sabse bada aur trusted cloud provider** hai.
AWS allow karta hai:
•Servers, storage, databases, networking, AI, ML, security, aur bahut kuch **internet ke through access** karne ka.
•Pay-as-you-go model pe kaam karta hai – **jitna use karo, utna pay karo**

- **List of AWS Services:**

1. **EC2 (Elastic Compute Cloud)** – Virtual servers
2. **S3 (Simple Storage Service)** – File storage
3. **RDS (Relational Database Service)** – Databases
4. **Lambda** – Serverless compute
5. **VPC (Virtual Private Cloud)** – Networking
6. **CloudFront** – Content delivery network
7. **DynamoDB** – NoSQL database
8. **EBS (Elastic Block Store)** – Disk-like storage
9. **IAM (Identity and Access Management)** – Security
10. **Lightsail** – Easy cloud for beginners

🌟 Features & Benefits of AWS Cloud:

Feature	Benefit (Fayda)
🌐 Global Availability	Multiple data centers worldwide
📈 Scalability	Bada ya chhota karna as per need
🔒 Security	High-level security with encryption and IAM
💰 Pay-as-you-go	Sirf jitna use karo, utna hi paisa do
🧩 Integration	Easily integrate with tools like GitHub, Docker etc.
🔄 Auto Backup & Recovery	Data secure and recoverable
📊 Monitoring Tools	CloudWatch, CloudTrail se performance monitor karo

⚙️ AWS Elastic Compute Cloud (EC2)

📄 What is AWS EC2?

EC2 (Elastic Compute Cloud) ek AWS service hai jiske through aap **virtual servers** (computers) create aur use kar sakte ho internet ke zariye.

Yeh servers bilkul waise hi kaam karte hain jaise real computers, lekin ye AWS ke data center me hote hain — aap sirf internet se connect hote ho.

EC2 aapko yeh allow karta hai ki aap:

- Apna **website**, **app**, ya **software** kisi virtual server pe chala sako
- Us server ko kabhi bhi **start**, **stop**, ya **scale** kar sako
- Pay karo sirf utna jitna use karo

Feature	Description (Hinglish)
🖥️ Virtual Servers	Multiple sizes/types ke servers available hain (instances)
📈 Scalable	Load zyada ho to auto-scale kar sakte ho
🔒 Secure	IAM & Security Groups ke saath secure access
🕹️ On-Demand	Jab chaho tab server start karo, stop karo
💰 Flexible Pricing	On-demand, Spot, Reserved instances available

Common Use Cases:

- Websites host karna (e.g., WordPress site)
- 💼 Business applications chalana
- Software testing & development
- Machine Learning model train karna
- 🎮 Game servers host karna

✅ Benefits of EC2:

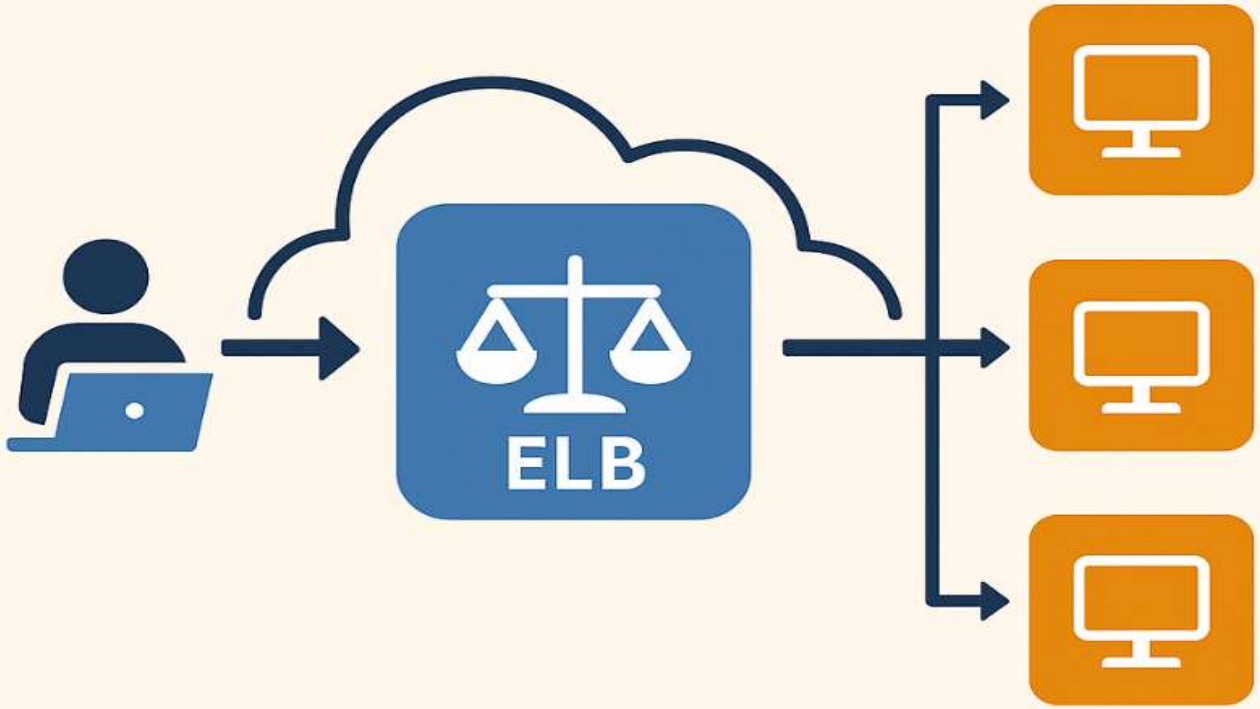
- No physical hardware needed
- Highly reliable and scalable
- Secure with firewall and encryption
- Available worldwide (multiple regions)
- Easy to start in just minutes

⚖️ AWS Elastic Load Balancer (ELB)

What is ELB?
Elastic Load Balancer (ELB) ek AWS service hai jo **incoming network traffic** ko **multiple servers (EC2 instances)** ke beech me distribute karta hai.

Simple shabdon me:
Agar aapki website ya app pe bahut zyada log ek saath aayen, to ELB traffic ko divide karke handle karta hai — taaki **server overload na ho** aur performance slow na ho.

Elastic Load Balancing



Step	Working Description (HiEnglish)
1	**User Request Aata Hai** — Jab user website ya app pe request bhejta hai (jaise HTTP/HTTPS request)
2	**ELB Request Receive Karta Hai** — Load balancer sabse pehle request accept karta hai
3	**Healthy EC2 Instance Ko Forward Karta Hai** — ELB check karta hai kaunsi EC2 instance healthy hai aur traffic wahaan forward karta hai
4	**Instance Response User Ko Jaata Hai** — EC2 instance process karta hai aur response user ko milta hai
5	**Auto Scaling Ke Saath Work Karta Hai** — Jab traffic zyada ho jaata hai, ELB naye instances ke saath bhi automatically kaam karta hai

Why use Load Balancer?

- Traffic control: Load divide hota hai
- High availability: Agar ek server down ho, dusra traffic le leta hai
- Auto scaling ke saath best kaam karta hai
- Secure communication (SSL support)

Real-Life Example:

Socho aapke paas ek shopping website hai. Sale ke time pe hazaaron log visit kar rahe hain.
Agar ek hi server pe sab aayenge to crash ho sakta hai.
ELB un sabhi users ko alag-alag servers pe bhejta hai — smooth experience sabke liye.

Types of Elastic Load Balancers in AWS

1️⃣ Application Load Balancer (ALB)

Layer: Works at **Layer 7** (Application layer - HTTP/HTTPS)

Use:

- Web apps, microservices, REST APIs
- URL-based routing (e.g., /login goes to one server, /products to another)
- Host-based routing supported

✅ *Best for: Complex routing, web-based apps*

2️⃣ **Network Load Balancer (NLB)**
Layer: Works at **Layer 4** (Transport layer - TCP/UDP)
Use:
•High performance, low latency traffic
•Handles **millions of requests per second**
•Supports static IPs and Elastic IPs
✅ *Best for: Gaming servers, video streaming, real-time apps*

3️⃣ Gateway Load Balancer (GWLB)
Use:
•Load balancing for **third-party virtual appliances** like firewalls, monitoring tools
•Integrates with security appliances
✅ *Best for: Network security and inspection use cases*

4️⃣ Classic Load Balancer (CLB) *(Legacy)*
Layer: Works at both **Layer 4 & Layer 7**
Use:
•Older generation of load balancer
•Still used in legacy systems
! *Not recommended for new apps — use ALB or NLB instead*

Feature	Benefit (Fayda)
🔄 Auto Load Distribution	Smooth traffic handling
🧭 Intelligent Routing	Smartly directs user to correct server
🔒 Secure SSL Support	Encrypted & secure connections
💻 Multi-Zone Availability	High availability & fault tolerance
💰 Pay-as-you-go	No upfront cost, usage-based billing

AWS VPC (Virtual Private Cloud)

🌐 What is AWS VPC?

VPC (Virtual Private Cloud) ek logically isolated network hai jo AWS ke cloud ke andar aapke liye banaya jata hai.

Simple shabdon mein:
AWS VPC aapko apna **private network** create karne deta hai jahan aap apne servers (EC2), databases, aur applications ko securely host kar sakte ho — jaise ki ek virtual data center.

🔐 VPC ka Use Kyu Hota Hai?

- Apna IP address range decide karne ke liye
- Subnets banane ke liye (public/private)
- Internet access control karne ke liye
- Security groups aur firewall rules set karne ke liye
- Private network + secure communication

Components of AWS VPC:

1 Subnets

- VPC ke andar chhote networks hote hain — inhe **subnets** kehte hain.
- Do tarah ke hote hain:
  - **Public Subnet** – Internet se connected
  - **Private Subnet** – Sirf internal access

2 Route Tables

- Batate hain ki network traffic kaise move karega
- Kis subnet se internet gateway ya NAT gateway jaye

3 Internet Gateway (IGW)

- VPC ko **internet se connect** karne ke liye hota hai
- Sirf **public subnets** ko internet access deta hai

4 NAT Gateway

- Private subnet ke resources ko **internet access** deta hai (e.g., updates install karna)
- But woh resources **internet se directly accessible nahi** hote

5 Security Groups

- Virtual firewall jaise kaam karta hai EC2 ke level pe
- Incoming aur outgoing traffic control karta hai

6 Network Access Control List (NACL)

- Subnet level pe firewall jaise kaam karta hai
- IP-based access rules set karta hai

7 DHCP Option Sets

- VPC ke andar IP address assign karne ke liye configurations deta hai

8 VPC Peering

- Do VPCs ke beech **secure connection** establish karta hai
- Useful jab alag region ya accounts me VPC ho

Benefit	Description
🔒 Secure	Apna private network banate ho AWS pe
⚙️ Customizable	Apne hisaab se IP, subnet, routing set kar sakte ho
📶 Scalable	Easily expand kar sakte ho jab traffic bade
🌐 Hybrid Ready	On-premise network se bhi connect kar sakte ho

What is AWS Storage?

AWS Storage ka matlab hai aapka data (files, folders, backups, videos, databases, etc.) ko **Amazon ke servers** par **securely store** karna.
Ye storage **cloud-based** hota hai — aap kahin se bhi access kar sakte ho, aur jitna chahiye utna use kar sakte ho.

🎁 **Types of Storage in AWS:**

1. **Object Storage** – Files aur media ke liye (e.g., S3)
2. **Block Storage** – Virtual hard drive jaise (e.g., EBS)
3. **File Storage** – Shared file system (e.g., EFS)
4. **Archive Storage** – Long-term backup ke liye (e.g., S3 Glacier)
5. **Hybrid Storage** – On-premises + Cloud (e.g., Storage Gateway)

🔰 Popular AWS Storage Services:

1️⃣ Amazon S3 (Simple Storage Service)
- **Object-based** storage for any type of data
- High durability (99.999999999%)
- Public ya private access
- Use: website images, videos, file backups, data lake

2️⃣ Amazon EBS (Elastic Block Store)
- **Block-level** storage for EC2 instances
- Jaise hard disk ya SSD
- Fast performance
- Use: running databases, apps on EC2

3 Amazon EFS (Elastic File System)

- **File storage** shared across multiple EC2 instances
- Automatically scalable
- Use: web servers, CMS, file sharing applications

4 Amazon S3 Glacier

- **Archive storage** for long-term backup
- Very low cost
- Retrieval time: minutes to hours
- Use: old documents, compliance records

5 AWS Storage Gateway

- Hybrid cloud storage service
- Connects on-premise environment to AWS Cloud
- Use: backup, disaster recovery, file sharing

□ Features of AWS Storage:

Feature	Description
□ Scalable	Petabytes tak data store kar sakte ho
🔒 Secure	Data encryption, IAM, access control
💰 Cost-effective	Pay only for what you use
🔄 Backup & Restore	Easy data recovery options
🌍 Global Access	Data accessible from anywhere
□ Lifecycle Rules	Auto move/archive data to cheaper storage

✅ **Benefits of AWS Storage:**

1. No physical hardware required
2. Fast, reliable, and secure
3. Integrates with other AWS services
4. Backup, disaster recovery, and archiving made easy
5. Globally available and scalable

🌐 Website/Web App Deployment on AWS (Step-by-Step)

Method 1: Static Website (HTML, CSS, JS) using Amazon S3

Best for simple websites with no backend (portfolio, blog, landing page)

📌 **Steps:**

1.Create S3 Bucket

- Go to AWS Console → S3 → Create a new bucket (bucket name = domain name)

2.Enable Static Website Hosting

- Bucket settings → Enable static website hosting
- Add index.html and error.html

3.Upload Website Files

- Upload all your HTML, CSS, JS files

4.Make Bucket Public

- Update bucket policy to allow public read access

5.(Optional) Connect Domain using Route 53

- Buy domain or use your domain
- Point domain to S3 website endpoint

Method 2: Dynamic Website (Backend + Frontend) using EC2

Best for full-stack apps (Node.js, Django, PHP, etc.)

📌 **Steps:**

1.Launch EC2 Instance

1. Go to AWS Console → EC2 → Launch instance
2. Select OS (e.g., Ubuntu, Amazon Linux)

2.Connect to Instance

•Use SSH to log in:

```
ssh -i "your-key.pem" ec2-user@<public-ip>
```

3.Install Web Server & App

•Example (for Node.js app):

```
sudo yum install nodejs npm
git clone your-repo-url
cd project-folder
npm install
node app.js
```

4.Allow Web Traffic

•In EC2 → Security Group → Allow port **80 (HTTP)** and **443 (HTTPS)**

- **Access Your App**
- Open your browser → Enter the **public IP** of EC2 instance
- **(Optional) Use Elastic IP**
- Allocate Elastic IP and attach it to your EC2 for fixed IP

✅ Method 3: Use AWS Elastic Beanstalk (for easy deployment)

Best for deploying web apps (Java, .NET, PHP, Python, Node.js)

📌 Steps:

1. Go to AWS Console → Elastic Beanstalk
2. Create a new Application
3. Upload your app code as ZIP
4. Select environment (e.g., Node.js, Python)
5. Beanstalk will auto-create EC2, load balancer, S3, etc.
6. App will be live on a URL like:
   http://your-app.us-east-1.elasticbeanstalk.com

Step No.	Step	Explanation
1	Go to AWS Console → Elastic Beanstalk	Sign in to the AWS Console, then search for and open **Elastic Beanstalk** service.
2	Create a New Application	Click on **"Create Application"**, enter the application name and optional details like description or tags.
3	Upload Your App Code as ZIP	Package your application code into a .zip file and upload it. This is the code that will be deployed.
4	Select Environment (e.g., Node.js, Python)	Choose the appropriate **Platform** for your app (Node.js, Python, etc.) and the **Environment type** (usually Web Server).
5	Beanstalk Auto-Creates Resources	Elastic Beanstalk will automatically create required AWS resources like **EC2**, **Load Balancer**, **S3 bucket**, **Auto Scaling**, and **Logs**.
6	App Will Be Live on a URL	After deployment, your app will be accessible at a URL like http://your-app.us-east-1.elasticbeanstalk.com.

What is an S3 Bucket?

Amazon S3 (Simple Storage Service) is an AWS service used to **store and retrieve files**, such as:

- Images
- Documents
- Backups
- Static website files
- Application code (e.g., for Elastic Beanstalk)

An **S3 Bucket** is like a **folder** in the cloud where your files are stored. Each bucket has a **unique name** and can hold unlimited files (called **objects**).

✓ Steps to Create an S3 Bucket:

Step No.	Action	Explanation
1	Go to AWS Console	Visit console.aws.amazon.com and sign in.
2	Open **S3**	In the search bar, type and select **S3** to open the S3 dashboard.
3	Click **Create bucket**	This button is usually at the top-right of the S3 dashboard.
4	Enter **Bucket Name**	Choose a unique name globally (e.g., my-app-bucket-2025).
5	Select **AWS Region**	Choose the region closest to your users or app (e.g., us-east-1).
6	Configure **Bucket Settings** (optional)	- You can enable versioning, encryption, logging, etc.
7	Set **Permissions**	Decide whether the bucket is private or public (default is **private**).
8	Click **Create bucket**	Your bucket will be created and ready to use.

✅ **PuTTY kya hai?**
PuTTY ek **free SSH client** hai jo aapke Windows system se kisi **remote server** (jaise AWS EC2 instance) ko **securely connect** karne ke liye use hota hai. AWS EC2 ka key pair .pem format mein hota hai, lekin PuTTY .ppk format chahiye hota hai — isliye hum **PuTTYgen** use karte hain conversion ke liye.

✅ PuTTY se EC2 instance connect karne ke steps:

Step No.	Step	Explanation (HiEnglish)
1	**PuTTY & PuTTYgen download karo**	Official website se download karo: https://www.putty.org
2	**.pem file ko .ppk mein convert karo**	- **PuTTYgen** open karo - **Load** button se .pem file select karo - **Save private key** dabao → .ppk ban jaayegi
3	**EC2 ka Public IP lo**	AWS Console → EC2 → Instances → apna instance select karo → **Public IPv4** address copy karo
4	**PuTTY open karo**	PuTTY application launch karo
5	**Host Name mein IP daalo**	Host Name mein likho: ec2-user@<your-public-ip> (jaise ec2-user@13.123.45.67)
6	**.ppk key add karo**	Left menu mein jao: **Connection → SSH → Auth** Yahan **Browse** karke .ppk file select karo
7	**Connect karo**	Ab **Open** dabao → SSH session start hoga
8	**Login ho jaayega**	Pehli baar prompt aayega, "Yes" dabao → EC2 instance se connect ho jaayega

Elastic Beanstalk kya hai?

AWS Elastic Beanstalk ek **Platform-as-a-Service (PaaS)** hai jo aapko web applications ko **aasani se deploy, manage, aur scale** karne deta hai bina infrastructure ki tension liye.

Aapko bas apna code (ZIP file) upload karna hota hai, aur Elastic Beanstalk automatically:

- Servers banata hai (EC2)
- Load balancer set karta hai
- Auto scaling karta hai
- Application ki health monitor karta hai

Fayda	Explanation (HiEnglish)
✅ **Easy Deployment**	Bas code upload karo, baaki sab kuch Elastic Beanstalk automatically manage karta hai
✅ **No Infrastructure Management**	Aapko servers, load balancer ya storage manage karne ki zarurat nahi hoti
✅ **Multiple Languages Supported**	Node.js, Python, Java, PHP, Ruby, Go, Docker jaise languages ko support karta hai
✅ **Auto Scaling & Load Balancing**	Traffic badhne par automatically new instances start karta hai, traffic kam hone par stop karta hai
✅ **Built-in Monitoring**	Logs aur health status easily dekh sakte ho AWS Console ya CloudWatch mein
✅ **Free to Start**	Elastic Beanstalk khud free hai, sirf jo AWS resources use hote hain unke liye pay karna hota hai
✅ **Easy Rollbacks**	Agar koi problem aaye to previous version pe easily wapas ja sakte ho
✅ **Custom Configuration**	Advanced settings ke liye .ebextensions file use karke apne environment ko customize kar sakte ho