AWS CLOUD PRACTITIONER CERTIFICATION

MODULE 1 - CLOUD CONCEPTS OVERVIEW

- Introduction to Cloud Computing
- Advantages of Cloud computing
- Introduction to Amazon Web Services (AWS)
- AWS Cloud Adoption Framework (AWS CAF)

* Introduction to Cloud Computing

- On-demand delivery of compute power, DB, storage, apps, & other IT resources over the Internet with pay-as-you-go pricing.
- It enables us to stop considering infrastructure as hardware but use it as a software (IaaS)
- 3 main cloud service models - IaaS (Infrastructure as a service), PaaS (Platform as a service), SaaS (Software as a service).

IaaS → PaaS → SaaS

More control over IT services ← → Less control over IT service.

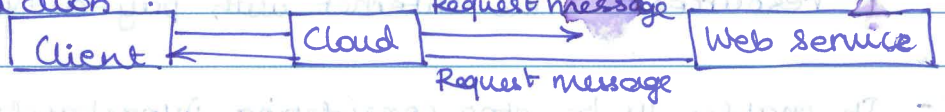
- 3 main cloud computing deployment models - cloud, hybrid, & on-premises (private cloud)

* Advantages of Cloud Computing

- Trade capital asset/expense (property, equipment, etc) for variable expense (resources on cloud, use how much it's needed)
- Reduced maintenance as resources that are needed are used.
- Benefit from massive economies of scale - Lower variable costs, because AWS is aggregate usage from all customers.
- Stop guessing capacity - How many users, is it overestimated or

underestimated. Cloud allows scaling on demand.
- Increased speed & agility. It takes minutes to access resources & don't have to wait for resources to arrive (in hardware)
- Stop spending money on running & maintaining data centres
- Go global in minutes. This ensures lower latency & better experience.

* Introduction to Amazon Web Services (AWS)
- Web services are pieces of software that is available over the internet & uses a format (XML / JSON) for API interaction.



- AWS is secure cloud platform that provides on-demand access to compute, storage, network, DB, etc.
- It is flexible, pay only for resources used, & works together like building blocks.

- 3 ways to interact with AWS tools.- AWS Management Console (GUI), AWS Command Line Interface (CLI), AWS Software Development Kits (SDK).

* AWS Cloud Adoption Framework (AWS CAF)
- CAF provides guidance & best practices to help organizations build cloud computing across organization & IT lifecycle to accelerate successful cloud adoption.

- Consists of 6 perspectives, each having a set of capabilities
- Business capabilities - Business, People, Governance,
- Technical capabilities - Platform, security, Operations
- Every department of organization can implement cloud by CAF

MODULE 2 - CLOUD ECONOMICS & BILLING

- Fundamentals of pricing
- Total Cost of Ownership
- AWS Organizations
- AWS Billing & Cost Management
- Technical support

* Fundamentals of pricing

- 3 fundamental drivers of cost with AWS - compute, storage, & data transfer.
- You save as usage increases. Tiered pricing for AWS S3, EBS, EFS ensures you use more but pay less per GB.
- Since 2006, AWS has lowered pricing 75 times. Future higher performing resources replace current resources for no extra charge
- AWS offers custom pricing, available for high volume projects with unique requirements.
- Services for no charge include - Amazon VPC, Beanstalk, Autoscaling, AWS Cloud Formation, IAM, But charges can be associated with resources provisioned along with these.

* Total Cost of Ownership

- On-premises differs from AWS based on deployment.

Traditional Infrastructure		AWS Cloud
• Equipment		• Pay-as-you-go
• Resources & Administration	≠	• Improve time to market & agility
• Contracts		• Scaling up & down
• Cost		• Self service infrastructure

- On-premises is very costly due to capital expenses, scaling cost, etc.

- Total Cost of Ownership (TCO) is financial estimate to help identify direct & indirect costs of a system.
- TCO is used to compare costs of running an entire infra environment or specific workload on-premises or AWS.
- Also used to budget & build the business case for moving to cloud.

- TCO consideration include server costs, storage costs, network costs, IT Labour costs.

- AWS Pricing Calculator is used to:
  - Estimate monthly cost
  - Identify opportunities to reduce monthly cost.
  - Model your solutions before building them
  - Explore price points & calculation behind your estimate
  - Find available instance types & contract terms.
  - Name your estimate & create & name groups of services.

- Estimate is broken into first 12 months, total upfront, & total monthly.

- Additional benefits include hard & soft benefits.
- Hard benefits include reduced spending on compute, storage, network, reduced capital expenses, reduced operational expenses, backup, disaster recovery, reduced operations personnel.
- Soft benefits include reusing cloud services, increased dev productivity, improved customer satisfaction, global tech, agile processes.

* AWS Organizations
- Used for consolidated billings of multiple accounts.
- We use IAM to allow/deny access to AWS services for users, groups, roles.
- In organizations, we use service control policies to allow or deny access to AWS services for individuals or groups in an organizational unit.

- We can access AWS organizations using AWS Management Console, AWS CLI, SDK, HTTPS Query application programming interface

* AWS Billing & Cost Management
- Used to pay AWS bills, monitor costs, & select bill as daily or based on monthly usage.
- AWS Cost & Usage Report Tool enables optimization by understanding cost & usage data.
- Dashboard is used to view status of month-to-date expenses, spend summary, forecast of how you're likely to spend this month.

- We have access to tools like AWS Budget, Cost & Usage Report, Cost Explorer.
- We use cost explorer to understand & manage AWS costs & usage over time.
- AWS Budgets are used to send alerts when we go over the budget threshold. Tracked quarterly, monthly, annually. Alerts can be sent using AWS Simple Notification System (SNS)
- Reports give comprehensive information about AWS costs & usage.

* Technical Support

- Provides unique combination of tools & expertise. Support is for experimenting AWS, production use of AWS, business support for AWS.

- AWS Trusted Advisor is an automated service that acts like customized cloud expert. There are Technical Account Managers (TAMs), & AWS Support Concierge.

- AWS Support offers 4 plans -
  - Basic Support - Resource Centre Access, Service Health Dashboard, product FAQs, discussion forums, support for health check.
  - Developer Support - Early development on AWS.
  - Business Support - Customers that run prod. workloads
  - Enterprise Support - Customers that run business & mission critical workloads.

MODULE 3 - AWS GLOBAL INFRASTRUCTURE OVERVIEW

- AWS Global Infrastructure
- AWS services & Service categories

* AWS Global Infrastructure

- Designed ~~to~~ & built to deliver a flexible, reliable, scalable, & secure cloud computing with high-quality global network performance

- AWS Region is a geographical location with one or more availability Zones.
- Availability Zones consists of one or more data centers.
- Data replication in data centre is isolated. So, it is controlled by users.
- Communication for the same uses AWS backbone network infra.

- When selecting a region, consider these factors-
  - Consider Data governance & legal requirements.
  - Choose data centres / regions closer to us. This decreases latency.
  - Not all AWS services are available in all regions.
  - Costs as it varies by region.

- There are 69 availability zones worldwide (AWS Region has multiple availability zones).
- These zones consists of discrete data centers, designed for fault isolation.
- These centers are connected with other availability zones by using high-speed private networking.
- While we choose our zone, AWS recommends replicating data & resources across zones for resiliency.

- AWS data centers are designed for security. They have redundant power, networking, & connectivity. Data centers have 50000 to 80000 physical servers.

- AWS CloudFront is a content delivery network used to distribute content to end-users in order to reduce latency.

- AWS Route 53 is a Domain Name System service where request going to either of these services will be routed to nearest Edge location to lower latency.

- AWS Points of Presence provides global network of 187 locations, consisting of 176 edge locations, 11 regional edge caches.

- A global content delivery network delivers content to end users with reduced latency (works with CloudFront)

- Features of AWS infrastructure:
  - Elasticity & scalability – Dynamic adaption & accommodates growth.
  - Fault-tolerance – Continues operating in presence of failure & built-in redundancy of components.
  - High availability – High level of operational performance, minimized downtime without human intervention.

* AWS services & Service Categories
- Broken down into 3 elements:
  - Regions
  - Availability zones
  - Points of Presence (Edge locations)

- Broad set of cloud-based services, totaling upto 23 different service/product categories.

- AWS Storage Services is represented by AWS S3 (Simple storage service) which is object storage service that offers scalability data availability, security, & performance.
- AWS Elastic Block Store (EBS) is high performance block storage designed for use with AWS EC2 for throughput & transaction-intensive workloads.
- AWS Elastic File system (EFS) that provides a scalable, fully managed elastic network file system (NFS) for use with Cloud services & on-premise resources.
- AWS S3 Glacier (Simple storage service Glacier), a secure, durable, & low cost S3 Cloud storage for data archiving & long-term backup.

- AWS Compute services include AWS Elastic Compute Cloud (EC2) provides resizable compute capacity as VM in cloud.
- AWS EC2 Auto scaling enables automatic adding/removing EC2 instances according to certain condition you define
- AWS Elastic Container Service (ECS), a highly scalable, high performance container orchestration that supports Docker container
- AWS EC2 container registry (ECR) which is fully managed Docker container registry that makes it easy for devs to store, manage, & deploy Docker container images.
- AWS Elastics Beanstalk, used to deploy & scale web apps & services on servers like Apache & Microsoft Internet Info Servic
- AWS Lambda, allows us to run code without provisioning o managing servers.
- AWS Elastic Kubernetes services (EKS), allows to deploy, manage, & scale containerized application that use Kubernetes on AWS.
- AWS Fargate, a compute engine for ECS that allows us to run containers without having to manage servers or clusters.

- AWS Database category includes AWS Relational Database service (RDS) to easily setup, operate, & scalable relational database in the cloud. It provides resizable capacity while automating time-consuming admin tasks.
- AWS Aurora, a MySQL and PostgreSQL compatible relational DB, 5 times faster than MySQL & 3 times faster than PostgreSQL databases.
- AWS Redshift enables running queries against petabyte of data, stored locally in Amazon. Delivers fast performance at any scale.
- AWS DynamoDB, fully managed key-value & document NoSQL DB that delivers single digit millisecond performance at any

scale. with built-in security, backup, & restore

- AWS Networking & content delivery service Category includes AWS Virtual Private Cloud (VPC), enabling us to provision logically isolated section of the cloud to launch AWS resources in a virtual network
- AWS Load Balancing, automatically distributes incoming app traffic across targets such as EC2, containers, IP addresses, and lambda functions.
- AWS CloudFront is a fast network CDN, securely delivers data, videos, & apps. & API globally with low latency & high transfer speeds.
- AWS Transit Gateway service enables to connect VPC and on-premise networks to a single centrally managed gateway.
- AWS Route 53, scalable cloud domain name system, designed for a reliable way to route end users to Internet apps. It translates URL to IP addresses
- AWS Direct Connect provides/establish a dedicated private network connection from data center/office to AWS, reducing cost & increase bandwidth.
- AWS Virtual Private Network (VPN), provides a secure private tunnel for your network / device to AWS global network.

- AWS security, identity, & compliance services include AWS Identity & Access Management (IAM), used to manage access to AWS services & resources securely.
- AWS Organizations allows us to restrict what services & actions are allowed in your accounts.
- AWS Cognito, adds user auth. & access control to web & mobile apps.
- AWS Artifact service provides on-demand access to Amazon

Web security and compliance reports, & online agreements.

- AWS Key Management service (KMS), enable creating & managing encryption keys. We can control the use of encryption across a wide range of AWS in apps.
- AWS shield, a managed distributed denial of service protection service that safeguards apps on AWS.

- AWS cost management service category include AWS cost & usage reports, containing most comprehensive set of AWS cost & usage data, incl. metadata about AWS, pricing & reservation.
- AWS Budgets, allows setting custom budgets that alerts us when AWS costs or usage exceeds or forecasted to exceed budgeted amount.
- AWS Cost explorer enables to visualize, understand, & manage AWS cost & usage overtime.

- AWS Management & government services include AWS Mgmt. Console, a web-based user interface for accessing AWS account.
- AWS Config allows tracking resource inventory & changes
- AWS CloudWatch allows monitoring resources & apps.
- AWS Auto Scaling provides scaling multiple resources to meet demand.
- AWS Command Line Interface (CLI), a unified tool to manage AWS.
- AWS Trusted Advisor, helps optimize performance & security using AWS best practices.
- AWS Well-Architected Tool helps reviewing & improving workloads.
- AWS CloudTrail tracks user activity & API across accounts.

MODULE 4 - AWS CLOUD SECURITY

- AWS Shared Responsibility Model
- AWS IAM
- Securing a new AWS Account
- Securing Accounts
- Securing Data
- Working to ensure Compliance

* AWS Shared Responsibility Model

- AWS secures the cloud - physical facilities & systems.
- Shared responsibility indicates which part of security is handled by AWS & which by customers.
- Customers are responsible for data in cloud.
- AWS Operates, manages, & controls software virtualization layer & hardware & global infrastructure components.
- Customers ensure that network is configured for security.

- AWS Responsibility - Security Features / Responsibilities
  - Physical data centres.
  - AWS Regions, Availability Zones, & Egde Locations.
  - Network Infrastructure like intrusion detection
  - Virtualization infrastructure

- Customer Responsibility
  - What gets deployed when using AWS Services.
  - Secure apps for configuring security groups & network settings.
  - Managing & securing AWS data.
  - Encrypt data & network when needed on the cloud.

- Service characteristics & security responsibility depends on the types of service.
- IaaS are services we maintain control over configuring networks & storage settings. We configure access controls & managing more aspects of security.
- PaaS are services managed by AWS like OS, DB patching, firewall configuration & disaster recovery.
- SaaS are services where software is centrally hosted. They are accessed via web browser, app or API. Customers don't manage the infrastructure that supports the service.

* **AWS Identity & Access Management (IAM)**
- Used to manage access to AWS resources.
- Resource is an entity we work with (EC2, S3, etc.)
- Users & type of access can be controlled by IAM. It is a free & global service.
- It allows us to control access to all services using policies & assign to users or groups like system admins, DB admins, storage & security admins.
- It handles authentication & verification for access of user, role or specific resource.
- Who can access, how they can access & which resources can be accessed are all IAM features.

- **IAM user** is a person / app that can authenticate AWS accounts
- **IAM group** is a collection of IAM users that are granted identical authorization.
- **IAM policy** is document that defines which resources can be accessed & level of access to each resource.
- **IAM Role** is used to grant a set of permissions for AWS requests.

- Authentication access can be given to the user. 2 different access can be assigned to users - programmatic access & AWS Management console access.
- Programmatic access needs the user to authenticate using an access key ID & a secret access key.
- Mgmt console access needs the user to authenticate using 12-digit account ID or alias, IAM user name, IAM password. If needed, MFA prompts as well.

- MFA adds extra security. It is a unique auth code.

- Authorization are what actions are permitted. By default, IAM users don't have access to anything. Permission / policy must be granted to allow them to access.
- The policy of least previlige is followed for new user.

- IAM policy is a document that defines permissions.
- 2 types of policies - Identity based & resource based
- Identity based policy attaches a policy to any IAM user / group / role.
- Single Policy → Multiple entities or Multiple Policies → Single entity is allowed.
- Resource based policy is attached to resources (S3, for e.g.)
- This type of policy specifies who has access to resource & what they can do. Only few AWS resources have this policy.

- IAM permissions specify the permissions that are denied / granted to users, roles, groups, & some resources.

- IAM group is a collection of IAM users. It is used to grant same permissions to multiple users.
- User → Multiple groups
- No default group & groups can't be nested.

- IAM role is IAM with specific permissions. It attaches policies to it & different from an IAM user. It can be for a person, group, or resource.
- Roles provide temporary access

* Securing a new AWS account
- AWS account, when created initially begins with single identity that has access to all services (root account).
- Do not use root access, unless needed. Use IAM instead to assign permissions, give access to users, etc.
- Enable password & MFA for users.
- Creating a group & attaching a policy is good.

- AWS cloudtrail tracks user activity on your account. It tracks all API interactions.

- Enable billing reports about use of AWS resources & cost.

* Securing accounts
- Using AWS organizations enables consolidation of multiple AWS accounts for managing centrally.
- It groups AWS accounts into Org. units (OUs) & attach different policies to each OU.
- Integration & support for IAM
- User control policies allow establishing control over AWS services & API actions that the account can access

- Service control Policies (SCP) offer centralized control over accounts, ensuring that accounts comply with access control guidelines.
- SCP are similar to IAM permission policies. However, SCP never grants permissions. It specifies maximum permission for an organization.

- AWS Key Management Service (KMS) enables to create & manage encryption keys.
- It enables the controlled use of encryption across AWS services & in your apps.
- Integrates AWS CloudTrail to log all key usage.
- Uses hardware security modules validated by Federal Information Processing Standards (FIPS) 140-2 to protect keys.

- AWS Cognito adds user sign-up, sign-in & access control to web/app. It scales to millions of users & supports sign-in with FB, MS, Google, etc. via Security Assertion Markup Language (SAML).

- AWS shield is a managed distributed DOS protection service.
- It safeguards apps on AWS, provides always-on detection & automatic inline mitigations
- AWS shield standard is free but Advanced is optional paid service.
- Typically used to minimize app downtime & latency.

* Securing Data

- Data encryption is important. AWS kms manages encrypti keys.
- AWS supports data at rest (on devices/tapes) using AES25
- Data can be encrypted in any service by kms incl. S3, EBS, EFS, RDS, etc.

- Data in transit (data moving across a network). This is encrypted using SSL, now TSL (Transport Layer Security)
- AWS Certificate Manager provides a way to manage, deploy & renew TSL/SSL certificates.
- HTTPS creates secure tunnel & uses TSL/SSL for bidirectional exchange of data.

- Newly created S3 buckets & objects are private & protected by default.
- S3 bucket can be protected by S3 Block Public Access, IAM policies, Bucket policies, ACL (Access Control Lists), AWS Trusted Advisor.

* Working to ensure Compliance

- AWS compliance programs - AWS engages with certifying bodies & independent auditors to provide customers with detailed info about policies, processes & controls est. & operated by AWS.
- Certifications & Attestations (ISO cert., for ex.)
- Laws, Regulations, & Privacy (EU General Data Protection Regulation, HIPAA)
- Alignments & Frameworks (EU-US Privacy Shield certified)

- AWS Config allows us to assess, audit, & evaluate configurations of AWS resources.
- It automatically evaluates recorded configuration vs desired configuration.
- Simplify compliance auditing & security analysis.

- AWS Artifact is a resource for compliance related info. It provides access to security & compliance reports, select online agreements.
- It provides documents related to AWS.

MODULE 5 - NETWORKING & CONTENT DELIVERY

- Networking Basics
- Amazon VPC
- VPC Networking
- VPC Security
- Route 53
- CloudFront

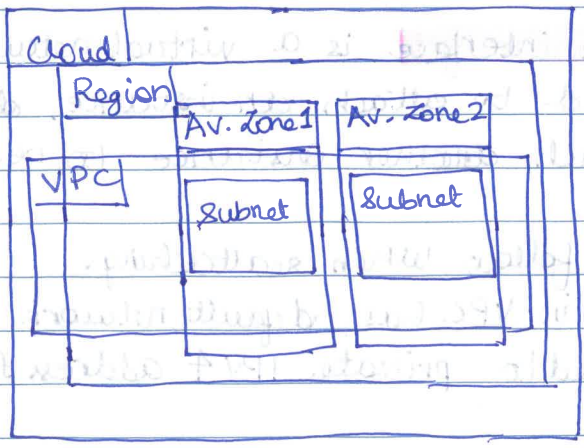
Networking Basics

- Computers are connected to each other using networks.
- A network can be logically partitioned into subnets.
- With a unique IP address, a machine identifies the network. (Ex: 192.0.0.1) is an IP address which is 32-bit. This 32 bit address is called IPv4. IPv6 has 128 bits.
- A common method to describe networks & groups of IP addresses is Classless Inter Domain Routing (CIDR)
- CIDR uses IP address followed by '/' and a number after that. This number tells how many bits of routing prefix must be steady & allocated to network identifier.

- Open Systems Interconnection (OSI) model is a conceptual model that is used to explain data that travels over a network.
- Consists of 7 layers & common protocols used to send data at each layer.

* Amazon VPC
- It enables us to provision a logically isolated section of AWS cloud where AWS instances / resources can be launched in a virtual network.
- Gives control of our virtual networking resources, incl. IP add. range, create subnets, & configure route tables & network gateways.
- It enables customization of network configuration for VPC & enables multiple layers of security.
- VMs can be launched in VPC.

- VPC & subnets are important in cloud.
- VPC logically is isolated from other VPCs, it is dedicated to our AWS account, belongs to single AWS region & can span multiple Availability zones.
- Subnets are range of IP addresses that divide VPC. They belong to single availability zone & are public / private.



- IP addressing enables resources in our VPC to communicate with each other & over internet as well.
- IPv4 CIDR block is assigned when a VPC is created.
- IP address range cannot be changed after creating VPC.
- Largest IPv4 CIDR block size is /16 & smallest is /28.
- CIDR blocks of subnets cannot overlap.

- When subnets are created, it needs its own CIDR block.
- For each CIDR block, AWS reserves 5 IP addresses & are not available for us to use. It reserves for -
  - Network address
  - VPC local routing & internal communication
  - DNS resolution
  - Future use
  - Network broadcast address

- Public IP addresses are 2 types - public IPv4 & elastic IP.
- Public IPv4 are manually assigned through elastic IP address and automatically through auto-assign public IP address settings at subnet level.
- Elastic IP addresses are associated with Amazon account & can be reallocated & remapped anytime.

- Elastic network interface is a virtual network interface that can be used to attach an instance, detach from instance & attach another instance to redirect network traffic.
- The attributes follow when reattaching.
- Each instance in VPC has default network interface that is assigned to private IPV4 address from range of VPC.

- Route tables contains set of rules called route that directs network traffic to & from your subnet.
- Each route specifies a destination & target. Every route table contains a local route for internal communication within VPC.
- Each subnet in VPC must be associated with a route table & only one route table is associated with a subnet but multiple subnets can be associated with same route table.

VPC Networking

- Internet Gateway - highly scalable, redundant VPC allowing communication between instances in VPC & public internet.
- 2 purposes - Provide target in VPC route tables for internet traffic & perform network address translation for instances that were assigned public IPv4 addresses.
- Making public subnet, add gateway & route entry to route table associated with subnet.
- NAT (Network address translation) gateway enables instances in private subnet to connect to internet / AWS services.
- To create NAT gateway, specify public subnet in which NAT gateway should live. Specify elastic IP address to associate with NAT gateway & update route table.

- VPC sharing enables multiple AWS accounts to create their app resources.

- VPC peering enables us to privately route traffic b/n 2 VPCs.
- This can be done by creating rules in route table to allow communication with each other.

- There are restrictions that IP addresses range cannot overlap, transitive peering is not supported, & can have one peering resource b/n 2 VPCs.

- VPC that are ~~if~~ launched cannot communicate with your own remote network.
- To enable access to our remote network from VPC, attach a virtual private gateway to VPC, create a custom route table, updating security group rules, create site-to-site VPN connection, & configure routing to pass through traffic to connection.

- AWS Direct connect enables us to establish a dedicated private connection b/n network & one of direct connect locations.
- It uses open standard 802.1q virtual LAN.
- VPC Endpoint is a virtual device that enables us to privately connect VPC to AWS services.
- VPC Gateway endpoint allows us to specify as a target for a a route in route table for traffic destined to S3, DynamoDB, etc.
- AWS privatelink simplifies security of data shared with cloud-based apps.

- AWS Transit Gateway is network transit hub that is used to interconnect virtual private clouds & can also connect on-premises network.
- We can attach VPC, VPN connections, AWS Direct connect to Transit Gateway.

* VPC Security

- Security groups (SG) acts as a virtual firewall that controls inbound & outbound traffic. to & from our instance.
- We can assign each instance in VPC to different security groups.
- They act as firewalls for EC2 instances. They have rules & by default are sealed shut to inbound traffic.
- They are stateful & allow outbound rules.

- Network Access Control List (network ACL) work at subnet level & controls traffic in & out of subnet.
- We specify ports, rules & protocols to control traffic.
- Each subnet in VPC must be associated with network ACL. If not specified, default is used.
- Network ACL can be associated to multiple subnets.
- It is stateless, has separate inbound & outbound rules that require configuration to allow/deny traffic.
- Default network ACL allow all inbound & outbound IPv4 traffic.

- The difference b/n security groups & network ACLs are
  - Scope - SG are instance level, NACL are subnet level.
  - Supported rules - Allow rules only (SG), NACL allow & deny
  - State - SG are stateful, NACL are stateless
  - Order of rules - All rules are evaluated in SG, NACL rule are evaluated in number order.

* AWS Route 53

- Gives ability to register a domain name.
- It is available & scalable DNS web service compliant with IPv4 & IPv6 addresses.

- It supports simple routing - user in single environment, Weighted routing - assign weights to resource record sets to specify frequency.
Latency routing - Improve global apps.
Geolocation routing - Route traffic based on location of users
Geoproximity routing - Route traffic based on location of resources
Failover routing - Fail over to a backup site if primary site becomes unreachable
Multivalue answer routing - Respond to DNS Queries with upto 8 healthy records selected at random.

- Route 53 DNS failover enables improving availability of apps that run on AWS by configuration of backup & failover scenarios for apps, enabling highly available multi-region architectures on AWS. & Creating health checks.

* AWS CloudFront
- Fast Content delivery service that securely delivers data to customers at high transfer speeds.
- It relies on Rote 53's geolocation routing.

MODULE 6 - COMPUTE
- Compute Services Overview
- AWS EC2 Part 1
- AWS EC2 Part 2
- AWS EC2 Part 3
- AWS EC2 Cost Optimization
- Container Services
- Introduction to AWS Lambda
- Introduction to Elastic Beanstalk

* Compute Services Overview

- EC2 provides resizable VMs.
- EC2 auto scaling supports app availability by defining condition that will automatically launch/terminate EC2 instances.
- ECR is used to store & retrieve Docker images.
- ECS is container Orchestration service that supports Docker.
- VMWare Cloud enables a hybrid cloud with a custom hardware
- Elastic Beanstalk provides a way to run & manage web apps.
- AWS Lambda is serverless compute soln.
- Eks enables to run managed kubernetes on AWS.
- Lightsail provides simple-to-use service for building an app or a website.
- Batch provides a tool for running batch jobs at any scale.
- Fargate provides way to run containers that reduce the need for managing clusters/servers.
- Outposts helps to run AWS services in on-premises data center.
- Serverless App Repository provides a way to discover, deploy & publish serverless app.

- Compute services are categorized -
- • IaaS provides VM, serverless computing, container-based computing & PaaS for web apps.

* EC2 Part 1

- EC2 provides VMs where apps are hosted.
- Ex of EC2 instances - App & web servers, Mail servers, game, etc
- It gives full access/control over guest OS
- We can launch instances of any size into Availability zone
- Launching of instances is from Amazon Machine Images (AMI) & we control traffic to & from instances.

- Nine decisions when you create an EC2 instance -

° **AMI to choose from?** AMI is a template to create EC2.
- It contains Windows/ Linux OS, additional software pre-installed.
- 4 categories of AMI - Quick Start (Linux & Windows), My AMI (AMI I created), AWS Marketplace (pre-configured templates from third parties), Community AMI (shared by others, use at own risk)
- Creating an AMI ex.

° **Choosing the instance type** depends on use case. It depends on memory (RAM), CPU, storage, Network performance.
- They are categorized as General purpose, compute optimized, memory optimized, storage optimized & accelerated computing.
- They offer family, generation, & size.
- Instance type sizes are named like 't3.nano'. Here, 't' is family name, the number is the generation number & the last is the size.
- Every instance has a network performance level.

* **EC2 Part 2**

- ° **Specify network settings.** Specify location, in the region to specify, to place it in the ~~instance~~ subnet, into any VPC.

° **Specify IAM role** if the instance interacts with other AWS services.
- Never store AWS credentials on EC2 instances.

° **Passing user data** is optional choice to customize runtime environment of instance, reduce no. of custom AMI that you build.

- This user data script will execute only once / first time the instance starts.

° Specifying storage & configuring is needed.
- configuring root volume, attach additional storage volume, where we specify size of volume & type (SSD or HDD).
- Elastic Block store (EBS) is durable, block-level storage volume. EBS offers various volume size.
- EFS, S3 are other storage options.

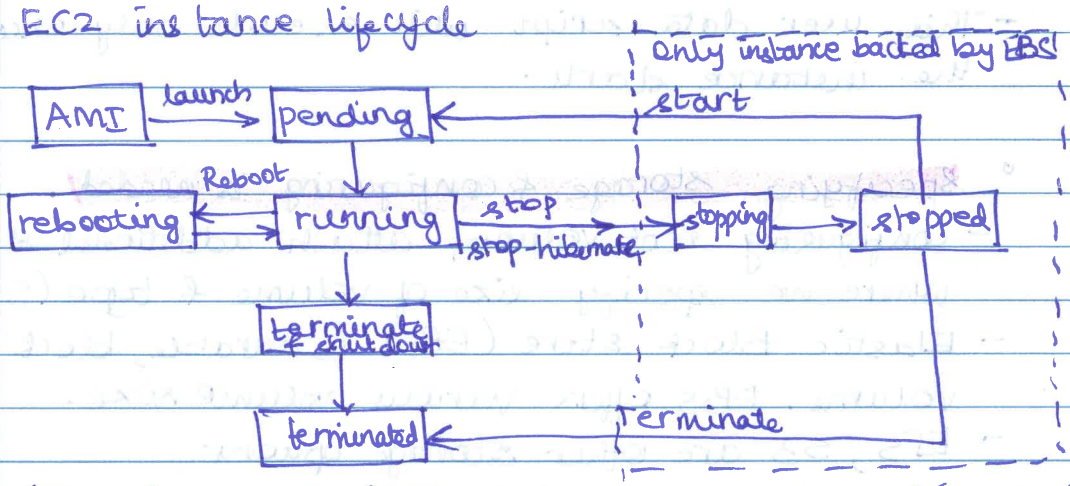
* EC2 Part 3

° Tags can be added. It is a label that can be assigned to AWS resource.
- It is how we can attach metadata to an EC2 instance.
- Benefits are filtering, automation, access control, etc.

° Security groups can be attached. Adding rules to specify source / ports that network communications can use.

° Key pair is the last choice to be made. It is a collection of public key & private key for enabling secure connection.
- For Windows AMI, use private key to obtain admin password that you need to login
- For Linux AMI, use private key to use SSH to securely connect to your instance.

- EC2 instance lifecycle



- After launch of AMI, instances are in pending state before running state.
- A running instance can also be rebooted & then it's ready
- Running instances can be terminated. They will enter a temporary shutting down state before getting terminated
- Instances backed by EBS can be stopped.
  If started, it will go back to pending state

- Consider using elastic IP address.
- Rebooting will not change DNS / IP address. but stopping & restarting it will change the DNS & IP.
- The private IPv4 will not change, however.
- If there is a need for persistent IP, associate elastic IP address with the instance.
- Elastic IP address can be associated with instances in the region. It remains allocated to your account until I choose to release it.

- EC2 instance metadata is data about your instance.
- You can view it as a website or in terminal.
- You can retrieve public IP, private IP, public hostname, instance ID, security groups, region, availability zone.

- Instances can be monitored using Cloud Watch.
- AWS Cloudwatch provides real-time metrics, charts in EC2 console, monitoring tab that can be viewed, etc.
- It can perform basic monitoring at not additional cost & data is sent to Cloudwatch every 5 mins.
- Detailed monitoring is chargeable for 7 pre selected metrics. It delivers data every minute.

* AWS EC2 Cost Optimization
- On-demand instances are eligible for free tier. Lowest upfront cost & most flexibility.
- Dedicated hosts are physical servers where instance capacity is dedicated for your use.
- Dedicated instances that run in a VPC on hardware that is dedicated to a customer.
- Reserved instances enable you to reserve computing capacity for a year / 3-yr terms
- Scheduled reserved instances enable purchase capacity reservation that recur daily, weekly, monthly basis with specified duration for a year.
- Spot instances enable you to bid unused EC2 instance capacity which can lower costs.
- Per second billing is for on-demand instance, reserved instances & spot instances.

- Benefits of pricing models.
- On demand - Low cost & flexibility
- Spot instances - Large scale, dynamic workload
- Reserved instances - Predictibility ensures compute capacity is available when needed.
- Dedicated Hosts - Save money on licensing costs. Help meet compliance & regulatory requirements.

- Use cases of various pricing models.
- On-demand instance pricing works well for spiky workloads or app dev / testing.
- Spot instances - app with flexible start/end times
- Reserved instances - steady state or predictable usage workloads
- Dedicated Hosts - Bring your own license.

- 4 pillars of cost optimization - right size, increase elasticity, optimal pricing models, optimizing storage choices.
- For right size, provision instances to match the need. CloudWatch metrics to review CPU, RAM, storage & network. Select the right size, then reserve.
- For increasing elasticity, stop / hibernate EBS backed instances that are not in use. Use auto-scaling to match needs based on usage.
- For optimal pricing models, optimize & combine purchase types. Consider AWS Lambda for serverless solutions
- For optimizing storage choices, resize EBS volumes, change EBS volume types, delete EBS snapshots when not in use, identify most appropriate destination for specific types of data.

- Measure, monitor, & improve as cost optimization is ongoing process.
- Define & enforce cost allocation tagging
- Define metrics, set targets, & review regularly.
- Encourage teams to architect for cost.

* Container Services

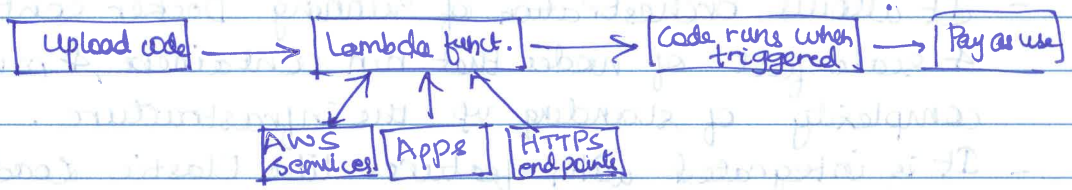
- Containers are method of OS virtualization, often smaller than VM
- They are repeatable, deliver environmental consistency due to self-contained environments.
- Help ensure apps deploy quickly. They are smaller in terms of image.
- Docker is an example.
- Docker is software that enables build, test, & deploy apps quickly.
- Containers are run on Docker & containers are created from image.
- They have libraries, system tools, code, & compiler, etc.
- Container Vs VM
- VM run on hypervisor but containers run on OS.
- Elastic Container Service (ECS) is highly scalable, fast, container mgmt. service.
- It allows orchestration of running Docker containers, maintain & scale fleet of nodes that run containers, & removes the complexity of standing up the infrastructure.
- It is integrated with features like Elastic Load Balancing, Amazon EC2 security groups, EBS volumes, IAM roles
- If we want to manage the EC2 clusters that manage the containers, we create the ECS cluster backed by EC2. If not, create AWS ECS cluster backed by AWS Fargate.
- Kubernetes is an open-source software for container orchestration. It is used to deploy & manage containerized apps at scale.

- This complements Docker. Docker enables running multiple containers on a single OS host & Kubernetes orchestrates multiple hosts.
- It automates container provisioning, networking, load distribution, & scaling.

- AWS Elastic Kubernetes service enables running Kubernetes on AWS. It is compatible with Kubernetes community tools & supports Kubernetes add-ons.

- AWS Elastic Container Registry is fully managed Docker container Registry that makes it easy for developers to store, manage, & deploy Docker container images.

* Introduction to AWS Lambda
- It is a serverless compute service.
- It lets us to run code without provisioning or managing servers.



- It supports multiple languages.
- Completely automated administration
- Built-in fault tolerance
- Orchestration of multiple functions
- Pay-per-use pricing

- Lambda functions are triggered by event sources.
- It can be an AWS service or app that trigger AWS Lambda to run.

- Configuring Lambda functions can be done by function name, runtime environment, execution role.
- Later add a trigger

- Ex = Create a thumbnail image using lambda function.
  - User creates function that invokes objects through S3.
  - Lambda reads the image & creates a thumbnail in the target bucket using policies.

- AWS Lambda limits include soft & Hard limits
- It limits the amount of Compute & storage resources.
- It limits 1000 concurrent executions in a region
- It can run upto 15 minutes at a time.
- Deployment package size is 250 MB
- Container image code package size is 10 GB.

* Introduction to Elastic Beanstalk
- A way to get web apps up & running
- Managed service automatically handles infrastructure config & provisioning, deployment, load balancing, auto-scaling, logging, etc.
- No charge for Elastic Beanstalk but pay only for resources that are used.

- Deployments support web apps written for platforms like Java, .NET, PHP, Go, Ruby, etc.
- It deploys on servers such as Apache, NGINX, Puma, etc.

- It is fast & simple, increases developer productivity, difficult to outgrow, & have complete resource control.

MODULE 7 - STORAGE

- AWS EBS
- AWS S3
- AWS EFS
- AWS S3 Glacier

AWS Elastic Block Storage

- Provides persistent block storage volumes for use with EC2 instances. It is also called non-volatile storage.
- It is automatically replicated within its availability zone to protect you from component failure.
- You can scale your usage up/down within minutes while paying a low price.

- There are block level & object level storage
- Block storage allows changing a character.
- Object storage will need to update the entire file.

- EBS enables individual storage volumes & attach to EC2.
- It offers block storage & volumes are replicated within its availability zone.
- It can be backed up automatically to S3 through snapshots.
- Uses of EBS include boot volumes & storage for EC2,
- Data storage with file system, DB hosts, enterprise appl
- Reduce costs by selecting type of storage accordingly

- Snapshots can be created & recreated at any time.
- EBS volumes can be encrypted at no additional cost.
- It offers elastic solutions that help increase capacity, change drives & move to a higher storage capacity.

- Pricing depends on various things.
  - Volumes - Persist independently from instance. All volume types are charged by amount provisioned per month.
  - IOPS - General purpose SSD are charged by the amount you provision in GB per month until storage is released.
  - Magnetic IOPS are charged by the number of requests to the volume
    Provisioned IOPS are charged by amount that is provisioned in IOPS.

  - Snapshots have added costs of EBS to S3 is per GB month of data.
  - Data Transfer - Inbound is free, outbound has charges.

* AWS S3

- Simple storage Service (S3) is object level storage - if something has to be changed, you change & reupload the entire file.
- Objects are stored in buckets. It has virtually unlimited data but a single object is limited to 5 TB.
- It is designed for 11.9's of durability.
- Bucket names must be unique across the world.

- S3 has a range of object-level storage classes -
  - S3 standard
  - S3 intelligent tiering
  - S3 Standard-Infrequent access
  - S3 One zone-Infrequent access
  - S3 Glacier
  - S3 Glacier Deep Archive

- S3 standard is designed for high availability, durability, & performance for frequently accessed data, used for content distribution & big data analytics.

- S3 standard-Infrequent access is used for accessing data that is less frequent but requires rapid access. High durability, high throughput, & low latency.

- S3 One Zone-Infrequent Access is for data accessed less frequently but requires rapid access when needed. Good for backups or easy-to-recreate data.

- S3 Intelligent Tiering is designed to optimize cost by moving data to most-effective access tier.

- S3 Glacier is a secure, durable, & low cost storage for data archiving.

- S3 Glacier deep Archive is lowest cost storage class. It supports long term retention & digital preservation for once/twice a year.

- S3 Bucket URLs are created by uploading data. You create a bucket in AWS Region & upload any number of objects to the bucket.
- It consists of region code & bucket name in the URL.
- Bucket names must be unique, contain a combination of only letters, numbers, & dashes.

- S3 pricing is pay for what you use. It includes GBs per month, transfer OUT to other regions, PUT, COPY, POST, LIST, & GET requests.

- No paying for Transfers IN S3, Transfer to AWS Cloud Front or EC2 in same region.

- To estimate pricing, consider type of storage class, amount of storage, requests made, data transfer

* AWS EFS
- storage for EC2 instances that multiple VM can access at the same time.

- Fully managed service that offers file storage in cloud.
- Works for big data analytics, media processing workflow, etc.
- It has shared storage, elastic capacity & supports Network File System
- It is compatible with all Linux based AMI for EC2.

- EFS architecture includes mounting EFS on EC2 instance & then read & write data to & from your file system.
- It can be mounted on VPC & EFS can be accessed by instances in different availability zone.

- To implement EFS, create EC2 resources & launch EC2 instance
- Create EFS
- Create mount targets in opt subnets & connect EC2 to them.
- Verify & protect AWS account

- File system has mount target, consisting of subnet ID, security groups, one / more per file system, etc.
- It has tags (key-value pairs)

* AWS S3 Glacier

- It is a secure, durable, & low cost storage service.
- It is for data archiving & long-term backup.
- Data stored in glacier can take several hours to retrieve.
- 3 key terms - ° archive (any media stored in glacier). It has own ID, & desc.
  ° Vault is container to store archives. Vault name & Region. Policies can be created for security.
- 3 options for retrieving data - Expedited data can take 1-5 minutes, bulk data takes 5-12 hours, & standard data takes 3-5 hours.
- It is used for media asset archiving, digital preservation, magnetic tape replacement, etc.
- To communicate apart from what is available in mgmt console, you use S3 Glacier REST APIs, CLI, or .NET SDKs.
- Lifecycle Policy enables deleting / moving objects based on age.

MODULE 8 - DATABASES

- Amazon RDS
- Amazon DynamoDB
- Amazon Redshift
- Amazon Aurora

* Amazon RDS

- RDS can help you manage app optimization.
- It has a database instance. It is an isolated DB environment that can contain multiple user created DB.
- DB instances & storage differ in performance & price.
- We specify DB instance to run. It (RDS) has 6 engines - MySQL, Aurora, Microsoft SQL server, PostgreSQL, MariaDB, Oracle.
- We can run DB instance in a VPC. RDS is in the private subnet & is made accessible to apps you choose.
- It can be used for high availability with Multi-AZ deployment. It is used for backup. It is kept in different availability zone & in case main instance fails, the multi AZ instance is used.
- RDS supports read replicas for MySQL, MariaDB, PostgreSQL & Aurora. It is asynchronous & can be promoted to master.
- RDS is used for web & mobile apps, ecommerce apps, games,
- It has clock-hour billing. They incur charges when running. The cost depends on DB characteristics (engine, size, memory)
- DB purchase type also impacts cost. On-demand instances or reserved instances. It also depends on no. of DB instances.
- Provisioned storage - No charge for active DB but charged for backup storage for terminated instances.
- No. of requests made. Inbound is free, outbound are charged.
- Data transfer is charged for outbound transfer.

* Amazon DynamoDB
- DynamoDB is non-relational DB.
- Relational DB works with structured data that is organized by tables, records & columns. It uses SQL.
- Non relational DB works with unstructured data that contains document, graph, key-value pair. It focuses on collection of documents. JSON is usually used.

- DynamoDB is NoSQL DB, fast & flexible for all apps
- It has virtually unlimited storage & low latency queries.

- Tables, items & attributes are core DynamoDB components
- It supports 2 keys - Partition key & partition & sort key
- To find an item in DynamoDB apart from PK, you use scan op
- Simple key is partition key. Composite key is partition & sort key.

* Amazon Redshift
- Fast, fully managed data warehouse that is used to analyse data by using SQL & existing business intelligence tools.
- It consists of cluster of leader and compute nodes.
- Leader node communicates & performs steps needed to obtain results from complex queries.
- Compute nodes run the compiled code & send intermediate results back to leader node.
- Redshift can scale up or down based on needs.
- It also provides Java Database Connectivity (JDBC)
- Redshift can be used for enterprise data warehouse, big data and SaaS.

* Amazon Aurora
- A MySQL & PostgreSQL RDS built on cloud.
- It automates time consuming tasks like patching, backup recovery, failure detection, etc.
- It is highly availability & resilient design. It stores multiple copies of your data across different Availability zones.
- This is continuously backed to S3.
- It is used for recovery. After DB crash, it does not need to replay redo log from last DB checkpoint but performs this on every read operation.

MODULE 9 - CLOUD ARCHITECTURE
- AWS Well Architected Framework Design Principles
- Operational Excellence
- Security
- Reliability
- Performance Efficiency
- Cost Optimization
- Reliability & High Availability
- AWS Trusted Advisor

* AWS Well Architected Framework Design Principles.
- Art of managing large architecture to manage size & complexity
- It is a guide for designing infrastructures that are secure, high performing, resilient, & efficient. It is a way to provide best practices that were developed through lessons learned by reviewing architectures.

- Pillars of framework - operational excellence, security, reliability, performance efficiency, & cost optimization

* **Operational Excellence**

- Focuses on Ability to run and monitor systems to deliver business value and to improve operations.
- It focuses on managing & automating changes, respond to events & define standards to successfully manage daily operations.

- 6 design principles -
  - Perform operation as code - Define workload
  - Annotate documentation - Automate creation of docs
  - Make frequent, small, reversible changes
  - Refine operations procedure frequently
  - Anticipate failure
  - Learn from operational events & failures

- You have to prepare, operate, & evolve in operational excellence.

* **Security**

- Focuses on ability to protect information, systems, & assets while delivering business value through risk assessment & mitigation strategies.
- It focuses on maintaining confidentiality, identifying & managing who can do what, protecting systems etc.

- 7 design principles -
  - Implement strong identity foundation
  - Enable traceability
  - Apply security at all layers

- Automate security best practice.
- Protect data in transit & at rest
- Keep people away from data
- Prepare for security events

- Foundational questions for security fall under IAM, detective controls, infrastructure protection, Data protecti & incident response.

* **Reliability**
- Focuses on ability to prevent & quickly recover from failures to meet business & customer demand.
- It focuses on setup, cross-project requirements, recovery planning, & handling change.

- 5 design principles -
- Test recovery procedures
- Automatically recover from failure
- Scale horizontally to increase aggregate system availabil
- Stop guessing capacity
- Manage change in automation

- Foundational questions for reliability fall under foundatio change management, & failure management.

* **Performance Efficiency**
- Focuses on ability to use IT & computing resources efficiently to meet system requirements & maintain efficiency as demand changes & tech evolves.

- It focuses on selecting right resource types & sizes based on requirement, monitor performance, & make informed decisi

to maintain efficiency as business needs evolve.

- 5 design principles-
  - Democratize advanced tech
  - Go global in minutes
  - Use serverless architecture
  - Experiment more often
  - Have mechanical sympathy

- Foundational questions fall under selection, review, monitoring, & tradeoffs.

* Cost Optimization
- Focus on ability to run systems to deliver business value at the lowest price point.
- It focuses on understanding & controlling when money is being spent, selecting most apt & right no. of resource type, analysing spending over time, & scaling to meet business needs without overspending.

-
  - 5 design principles
    - Adopt a consumption model
    - Measure overall efficiency
    - stop spending money on data center operations
    - Analyse & attribute expenditure
    - Use managed & app level services to reduce cost of ownership.

- Foundational questions fall under expenditure awareness, cost-effective resources, matching supply & demand, & optimizing over time.

* Reliability & High Availability
- Reliability is the measure of system's ability to provide functionality when desired by the user.
- System = all hardware, firmware, & software.
- Mean time b/n failures = total time in service / no. of failures

- Availability is % of time a system is operating normally or correctly performing operations of it over total time.
- A % of uptime (ex: 99.9%) over time (ex: 1 year)
- Number of 9s — Five 9s means 99.999% availability.

- High availability is a system that can withstand some measure of degradation while remaining available.
- Downtime is minimized & minimal human intervention is needed.

- Factors that influence availability are - fault tolerance, scalability, & recoverability.

* AWS Trusted Advisor
- Online Tool that provides real time guidance to help provision your resources following AWS best practices.
- Looks at entire AWS environment & gives you real time recommendations in 5 categories - cost optimization, performance, security, fault tolerance, & service limits.

MODULE 10 - AUTO SCALING & MONITORING
- Elastic Load Balancing
- Amazon Cloud Watch
- Amazon EC2 Autoscaling

* Elastic Load Balancing

- It distributes incoming app or network traffic across multiple targets in a single or multiple Av. zones.
- Scales load balancer as traffic to apps increase over time

- 3 types of load balancers
  - Application load balancer - Balances HTTP/HTTPS traffic, routes traffic based on content of request & operates at app layer of OSI.
  - Network load balancer - Balances TCP, UDP, TLS traffic, routes traffic based on IP protocol data & operates at transport layer.
  - Classic load balancer balances HTTP, HTTPS, TLS, TCP & operates both at app & transport layer. It balances across EC2.

- It accepts incoming traffic from clients and routes requests to its registered targets in one or more Av. zones.
- It has a set of listeners (process that checks for conn. requests). They have port no. from load balancer to target.

- Used for containerized apps, invoke lambda, highly available & fault tolerant apps, elasticity & scalability, etc.

- Load Balancers can be monitored using Amazon CloudWatch, access logs, AWS CloudTrail logs.

* Amazon CloudWatch

- It monitors AWS resources & ensures apps that run on AWS.
- Collect & track standard & custom metrics
- Alarms - send notifications to Amazon SNS topic, perform

EC2 auto scaling or EC2 action

- Events that define rules to match changes in AWS environment & route these to one or more target functions for processing.

* Amazon EC2 Auto Scaling

- Helps maintain app availability. It automatically adds or remove EC2 instances according to conditions defined
- Detects impaired EC2 instances & unhealthy apps & replace it without our intervention.
- It provides manual, scheduled, dynamic or ondemand & predictive way to scale.

- Auto scaling group is collection of EC2 instances that are treated as logical grouping for the purposes of automatic scaling & management.

- Launching instances is called scaling out & terminating is called scaling in.

- To launch EC2 instances, an auto scaling group use a launch configuration (AMI ID, Instance type, IAM role, EBS volume & security group) to keep a track of type of instan
- Then, we specify where to scale (min & max no. of groups) Then, we launch in subnet & VPC or load balancer.
- We lastly specify when to scale the event. We might have manual scaling, scheduled, dynamic, predictive scaling or maintain the current number.

- Dynamic scaling can be done by creating Cloudwatch alarm
- When performance limit is breached, alarm is triggered by Cloudwatch & an auto scaling event scales out or in based

on the alarm triggered.

- AWS Auto Scaling monitors apps & adjusts capacity to maintain steady, predictable performance at lowest possible cost.
- Provides simple, powerful user interface that enables building scaling plans for resources like EC2, DynamoDB tables & indexes, Amazon Aurora Replicas, etc.