MODULE 1 · STUDY NOTES

Cloud Concepts Overview

Cloud computing defined, service and deployment models, the six advantages, the AWS service catalog, and the Cloud Adoption Framework

Everything else in this course builds on the vocabulary this module establishes: what cloud computing actually is (a one-sentence definition worth memorizing), how the three service models and three deployment models divide control between you and the provider, why the cloud wins economically — six specific advantages the exam quotes by name — what AWS itself offers and how its catalog organizes into categories of building-block services, and, because adoption is an organizational problem as much as a technical one, how the AWS Cloud Adoption Framework structures the journey into six perspectives spanning people, process, and technology.

LEARNING OBJECTIVES

- Define different types of cloud computing models
- Describe six advantages of cloud computing
- Recognize the main AWS service categories and core services
- Review the AWS Cloud Adoption Framework (AWS CAF)

1.1 Introduction to cloud computing

Cloud computing is the on-demand delivery of compute power, database, storage, applications, and other IT resources via the internet with pay-as-you-go pricing. Those resources run on server computers in large data centers in different locations around the world — when you use a provider like AWS, the provider owns the computers you are using. The resources work together like building blocks that you assemble into solutions to meet business goals and technology requirements.

The deeper idea is a mental shift: **stop thinking of infrastructure as hardware, and think of — and use — it as software.** In the traditional model, infrastructure is physical: it needs space, staff, physical security, planning, and capital expenditure, and it comes with a long procurement cycle of acquiring, provisioning, and maintaining. Worst of all, you must provision capacity by **guessing your theoretical maximum peak**. Guess high and you pay for expensive resources that sit idle; guess low and you can't meet demand; and when needs change, a new solution costs more time, effort, and money. Provisioning even a simple website the traditional way means buying hardware, racking and stacking it, placing it in a data center, and managing it — expensive and slow.

TRADITIONAL – INFRASTRUCTURE AS	CLOUD – INFRASTRUCTURE AS SOFTWARE
physical, slow, and guessed	*flexible, elastic, disposable*
▪ Space, staff, physical security, planning, capital expenditure ▪ Long hardware procurement cycle: acquire, provision, maintain ▪ Capacity = a guess at the theoretical maximum peak ▪ Overestimate → idle resources · underestimate → insufficient capacity	▪ Select services that match your needs; provision and terminate on demand ▪ Pay for what you use; scale elastically, automatically ▪ Treat resources as temporary and disposable ▪ New solutions launch quickly with low upfront cost

Software-defined infrastructure also removes the **undifferentiated heavy lifting** — procurement, maintenance, and capacity planning — so developers and IT departments can focus on what matters. As cloud computing grew, distinct **service models** and **deployment strategies** emerged, each offering a different level of control, flexibility, and management. Understanding the differences tells you which set of services is right for your needs.

TABLE 1.1 The three cloud service models

MODEL	WHAT THE PROVIDER GIVES YOU	WHAT YOU FOCUS ON	CONTROL
IaaS	Basic building blocks for cloud IT: networking features, computers (virtual or on dedicated hardware), data storage space	Everything above the infrastructure	Highest flexibility and management control — most similar to existing IT
PaaS	The underlying infrastructure managed for you — usually hardware and operating systems	Deployment and management of your applications	Less control, fewer chores
SaaS	A completed product that the provider runs and manages — in most cases, end-user applications	Only how you plan to use the software (e.g., web-based email)	Least control — none of the plumbing

TABLE 1.2 The three cloud deployment models

MODEL	WHAT IT MEANS	WORTH NOTING
Cloud	The application is fully deployed in the cloud; all parts run there	Either built in the cloud or migrated to it; can use low-level infrastructure pieces or higher-level services that abstract management, architecting, and scaling
Hybrid	Connects infrastructure and applications between cloud-based resources and existing resources not in the cloud	Most commonly cloud + existing on-premises infrastructure — extend into the cloud while staying connected to internal systems
On-premises	Virtualization and resource-management tools in your own data center — sometimes called *private cloud*	Provides dedicated resources but few cloud benefits; in most cases the same as legacy IT infrastructure

AWS is not an alien landscape — nearly every traditional data-center component has a direct AWS analog, and with AWS services you can do almost everything you would do with a traditional data center.

TABLE 1.3 AWS analogs for traditional IT

IN A TRADITIONAL DATA CENTER	ON AWS
Firewalls, access control lists, administrators	Security groups, network ACLs, IAM
Routers, network pipelines, switches	Elastic Load Balancing, Amazon VPC
On-premises servers	AMIs (Amazon Machine Images) + Amazon EC2 instances
DAS, SAN, NAS, RDBMS	Amazon EBS, Amazon EFS, Amazon S3, Amazon RDS

SECTION TAKEAWAYS

- Cloud computing = on-demand delivery of IT resources via the internet with pay-as-you-go pricing.
- Think of — and use — your infrastructure as software, not hardware.
- Three service models: IaaS, PaaS, SaaS. Three deployment models: cloud, hybrid, on-premises (private cloud).
- Almost anything you can implement with traditional IT has an AWS cloud analog.

COMMON PITFALL

Two favorite distortions: (1) "private cloud" sounds cloudy but is **on-premises legacy IT** with virtualization — it delivers dedicated resources, not the cloud's benefits; (2) the service models order by **control**: IaaS gives the most, SaaS the least. If a question mentions managing the OS, you're in IaaS territory; if it mentions only using an application, that's SaaS.

1.2 Advantages of cloud computing

Why are so many companies moving to the cloud? The course's answer is six named advantages — and the names matter, because exam questions quote them. The first is economic vocabulary worth pinning down: **capital expenses (capex)** are funds a company uses to acquire, upgrade, and maintain physical assets such as property, buildings, or equipment — in a data center you pay for everything whether you use it or not. A **variable expense**, by contrast, is one that the person who bears the cost can easily alter or avoid — you pay only when you consume resources, and only for the amount you consume.

TABLE 1.4 The six advantages of cloud computing

ADVANTAGE	THE OLD PROBLEM	WHAT THE CLOUD CHANGES
1 · Trade capital expense for variable expense	Heavy investment in data centers and servers before you know how they'll be used	Pay only when you consume, only for the amount consumed; adopt new applications in minutes; reduced maintenance frees focus for core business goals
2 · Massive economies of scale	Your purchasing power is limited to your own scale	Usage from hundreds of thousands of customers is aggregated in the cloud, so AWS achieves higher economies of scale — and passes savings on as lower pay-as-you-go prices
3 · Stop guessing capacity	Capacity decisions made before deployment → expensive idle resources or limited capacity	Access as much or as little as you need, and scale up and down as required with only a few minutes' notice
4 · Increase speed and agility	Weeks between wanting resources and having them	New IT resources are a click away — minutes, not weeks — so the cost and time to experiment and develop drop dramatically
5 · Stop spending money running and maintaining data centers	Racking, stacking, and powering servers consumes the IT budget and attention	Focus on projects that differentiate your business — and on your own customers — instead of on infrastructure
6 · Go global in minutes	Serving distant customers means building distant infrastructure	Deploy your application in multiple AWS Regions around the world with a few clicks: lower latency and a better customer experience at minimal cost

MEMORIZE THE SIX

- Trade capital expense for variable expense
- Benefit from massive economies of scale
- Stop guessing capacity
- Increase speed and agility
- Stop spending money on running and maintaining data centers
- Go global in minutes

COMMON PITFALL

The module's sample exam question lives here: *why is AWS more economical than traditional data centers for applications with **varying** compute workloads?* The keyword is **varying** — the answer is that EC2 instances can be **launched on demand when needed**. Monthly billing, administrative access, and permanently running peak-sized fleets are all wrong: the last one literally recreates the idle-capacity problem the cloud eliminates.

1.3 Introduction to Amazon Web Services (AWS)

First, the generic term: a **web service** is any piece of software that makes itself available over the internet (or a private intranet) and uses a standardized format — **XML or JSON** — for the request and response of an **API** interaction. It is

not tied to any operating system or programming language, it's self-describing via an interface definition file, and it's discoverable.

Amazon Web Services (AWS) is a secure cloud platform that offers a broad set of global cloud-based products — delivered over the internet, so you get on-demand access to compute, storage, network, database, and other IT resources plus the tools to manage them. Resources can be provisioned and launched in minutes; the environment can be reconfigured, scaled up or down automatically to match usage patterns and optimize spending, or shut down temporarily or permanently. Billing becomes an **operational expense instead of a capital expense**. AWS services are designed to work together like **building blocks**: assemble them into sophisticated, scalable solutions, then adjust as needs change.

Services fall into **categories** — the module's category map shows 23 of them, from Compute, Storage, Database, and Networking & Content Delivery to Analytics, Machine Learning, IoT, Robotics, Blockchain, and even Satellite. You pick services across categories to build a solution.

> **WORKED EXAMPLE: A SIMPLE DATABASE APPLICATION**
>
> Customers send data to your **Amazon EC2** instances (compute category). The EC2 servers batch the data in one-minute increments and add an object per customer to **Amazon S3** (storage). A nonrelational database, **Amazon DynamoDB**, powers the application — for example, building an index so you can find all objects a given customer generated over a certain period. The whole stack runs inside an **Amazon VPC** (networking). Four categories, one solution: that's the building-block model in action.

Which service you choose depends on **your business goals and technology requirements** — the same problem often has several valid AWS answers. Compute alone illustrates the range:

TABLE 1.5 Choosing a compute service by use case

YOU WANT TO...	CHOOSE
Keep complete control over your AWS computing resources	Amazon EC2
Run code without managing or provisioning servers	AWS Lambda
Have your web applications deployed, managed, and scaled for you	AWS Elastic Beanstalk
Use a lightweight cloud platform for a simple web application	Amazon Lightsail
Run hundreds of thousands of batch workloads	AWS Batch
Run AWS infrastructure inside your on-premises data center	AWS Outposts
Implement a containers or microservices architecture	Amazon ECS, Amazon EKS, AWS Fargate
Migrate an on-premises server-virtualization platform to AWS	VMware Cloud on AWS

TABLE 1.6 Services this course covers, by category

CATEGORY	CORE SERVICES
Compute	Amazon EC2, AWS Lambda, AWS Elastic Beanstalk, Amazon EC2 Auto Scaling, Amazon ECS, Amazon EKS, Amazon ECR, AWS Fargate
Security, Identity & Compliance	AWS IAM, Amazon Cognito, AWS Shield, AWS Artifact, AWS KMS
Storage	Amazon S3, Amazon S3 Glacier, Amazon EFS, Amazon EBS
Database	Amazon RDS, Amazon DynamoDB, Amazon Redshift, Amazon Aurora
Networking & Content Delivery	Amazon VPC, Amazon Route 53, Amazon CloudFront, Elastic Load Balancing
Management & Governance	AWS Trusted Advisor, CloudWatch, CloudTrail, Well-Architected Tool, AWS Auto Scaling, AWS CLI, AWS Config, Management Console, AWS Organizations
Cost Management	AWS Cost & Usage Report, AWS Budgets, Cost Explorer

THREE WAYS TO INTERACT WITH AWS

- **AWS Management Console** — a rich graphical interface to the majority of AWS features (brand-new features occasionally launch before all their capabilities reach the console).
- **AWS Command Line Interface (AWS CLI)** — a suite of utilities launched from command scripts in Linux, macOS, or Windows; ideal for discrete commands and repeatable scripts.
- **Software development kits (SDKs)** — packages for popular languages (Java, Python, and others) that let you call AWS from your own code, up to deploying and monitoring complex systems entirely through code.
- All three are built on a **common REST-like API** — the foundation of AWS.

COMMON PITFALL

"Which compute service?" questions are answered by matching the **requirement keyword**: *complete control* → EC2 · *no server management* → Lambda · *deploys and scales web apps for you* → Elastic Beanstalk · *containers* → ECS/EKS/Fargate · *AWS hardware on premises* → Outposts. Don't default to EC2 just because it's famous.

1.4 Moving to the AWS Cloud – the AWS Cloud Adoption Framework (AWS CAF)

Cloud adoption doesn't happen instantly. An organization is **people, processes, and technology**, and all three must align for a migration to succeed. The cloud changes how technology is obtained, used, and managed — and how organizations budget and pay for it — so fundamental changes must be discussed across the entire organization, with support from stakeholders inside *and outside* IT. Leaders must understand the current state, the target state, and the transition needed between them. The **AWS Cloud Adoption Framework (AWS CAF)** exists for exactly this: guidance and best practices that help organizations identify gaps in skills and processes and build a comprehensive approach to cloud computing — across the organization and throughout the IT lifecycle — to accelerate successful adoption.

At the highest level, the CAF organizes guidance into **six areas of focus, called perspectives**, spanning people, processes, and technology. Each perspective consists of a set of **capabilities** — distinct responsibilities owned or managed by functionally related stakeholders. Capabilities are used to identify which areas of the organization need

attention; the gaps found become **prescriptive work streams** that support the cloud journey. In general, **Business, People, and Governance** focus on *business* capabilities, while **Platform, Security, and Operations** focus on *technical* capabilities.

TABLE 1.7 The six CAF perspectives

PERSPECTIVE	FOCUS	STAKEHOLDERS	CORE CONCERN
Business	Business	Business managers, finance managers, budget owners, strategy stakeholders	Ensure IT is aligned with business needs and IT investments trace to demonstrable business results; build the business case and prioritize adoption initiatives
People	Business	Human resources, staffing, people managers	Evaluate organizational structures, roles, and new skill and process requirements; prioritize training, staffing, and organizational changes to build an agile organization
Governance	Business	CIO, program managers, enterprise architects, business analysts, portfolio managers	Align IT strategy and goals with business strategy and goals, so the organization maximizes the business value of IT and minimizes business risks
Platform	Technical	CTO, IT managers, solutions architects	Understand and communicate the nature of IT systems and their relationships; describe the target-state architecture in detail; principles and patterns for new cloud solutions and for migrating on-premises workloads
Security	Technical	CISO, IT security managers, IT security analysts	Meet security objectives for visibility, auditability, control, and agility; structure the selection and implementation of security controls
Operations	Technical	IT operations managers, IT support managers	Define how day-to-day, quarter-to-quarter, and year-to-year business is conducted; identify the process changes and training needed for successful cloud operations

SECTION TAKEAWAYS

- Cloud adoption requires a thoughtful, deliberate strategy and alignment across the whole organization — people, process, and technology.
- The AWS CAF helps organizations develop efficient and effective plans for the adoption journey.
- Guidance is organized into six perspectives; each perspective is a set of capabilities owned by key stakeholders.
- Identifying capability gaps produces prescriptive work streams for the cloud journey.

COMMON PITFALL

Expect stakeholder-matching questions: **CIO → Governance**, **CTO → Platform**, **CISO → Security**, **HR → People**. And keep the halves straight — Business, People, Governance are the *business* perspectives; Platform, Security, Operations are the *technical* ones. Governance sitting on the business side is the counterintuitive one the exam likes.

Module summary

- Cloud computing is the on-demand delivery of compute power, database, storage, applications, and other IT resources via the internet with pay-as-you-go pricing.
- The cloud turns infrastructure into software: flexible, provisioned and terminated on demand, elastically scaled, and treated as temporary and disposable — no more guessing peak capacity.
- Service models trade control for convenience: IaaS (building blocks, most control) → PaaS (provider runs hardware and OS) → SaaS (completed product, you just use it).
- Deployment models: cloud (fully in the cloud), hybrid (cloud connected to on-premises), and on-premises "private cloud" (essentially legacy IT with virtualization).
- Six advantages, by name: capex → variable expense · massive economies of scale · stop guessing capacity · increase speed and agility · stop running data centers · go global in minutes.
- AWS is a secure cloud platform of global, building-block services organized into categories; choose services by business goals and technology requirements (EC2 vs. Lambda vs. Elastic Beanstalk, etc.).
- You interact with AWS three ways — Management Console, AWS CLI, and SDKs — all built on a common REST-like API.
- The AWS CAF organizes adoption guidance into six perspectives — Business, People, Governance (business) + Platform, Security, Operations (technical) — each a set of capabilities owned by functionally related stakeholders.

Review questions

1. Recite the definition of cloud computing, and name the two properties that make it "cloud."

Answer: The on-demand delivery of compute power, database, storage, applications, and other IT resources via the internet with pay-as-you-go pricing. The two hallmarks: resources arrive on demand (not procured in advance) and you pay as you go (only for what you consume).

2. What does "infrastructure as software" change about capacity planning?

Answer: Hardware forces you to provision against a guessed theoretical maximum peak — overestimate and resources sit idle, underestimate and you can't meet demand. Software infrastructure is provisioned and terminated on demand and scales elastically in minutes, so capacity tracks actual demand and resources become temporary and disposable.

3. A team wants to deploy its own web application code but manage no servers, OS patching, or scaling. Which service model is that, and which AWS compute service fits?

Answer: PaaS — the provider manages the underlying hardware and operating systems while you focus on deploying and managing the application. AWS Elastic Beanstalk fits: it deploys, manages, and scales web applications for you (AWS Lambda if you want to run code with no servers to think about at all).

4. Distinguish the three deployment models, and identify which one delivers the fewest cloud benefits.

Answer: Cloud: the application runs entirely in the cloud. Hybrid: cloud-based resources connect to existing infrastructure not in the cloud, most commonly on-premises. On-premises ("private cloud"): virtualization and resource management in your own data center — it offers dedicated resources but few cloud benefits, making it largely legacy IT.

5. Explain the chain from customer aggregation to your bill.

Answer: Usage from hundreds of thousands of customers is aggregated in the cloud, letting AWS achieve far higher economies of scale than any single company; those savings are passed on as lower pay-as-you-go prices — a lower variable cost than you could achieve on your own.

6. Which of the six advantages does "deploy your application in multiple AWS Regions with a few clicks" illustrate, and what does the end user notice?

Answer: Go global in minutes. End users get lower latency and a better experience because the application runs geographically closer to them — achieved simply and at minimal cost.

7. Map these stakeholders to their CAF perspectives: CISO, CTO, CIO, and HR managers.

Answer: CISO → Security (visibility, auditability, control, agility). CTO → Platform (describe systems and target-state architecture). CIO → Governance (align IT strategy with business strategy). HR → People (training, staffing, organizational change). Remember: Business/People/Governance are business perspectives; Platform/Security/Operations are technical.

8. In the sample exam question, why is "EC2 instances can be launched on-demand when needed" correct, while "monthly billing" and "permanently running enough instances for peak" are wrong?

Answer: The question keys on *varying* compute workloads — the economy comes from paying only when you consume. Monthly billing isn't the pricing advantage, administrative access is about control rather than cost, and permanently running a peak-sized fleet recreates exactly the idle-capacity waste of a traditional data center.