CLOUD COMPUTING ESSENTIALS

A practical guide to cloud services, architecture, security, and cost

LEARNSAPPEAL | ORIGINAL COURSE

Understand the building blocks behind modern cloud services.

Learn how cloud resources are delivered, connected, protected, scaled, monitored, and paid for.

Before you begin

Everyday computer and web knowledge is enough. Examples use provider-neutral terms so you can apply the ideas across different platforms. This course explains concepts and planning; it does not ask you to create paid services. If you try a provider console later, check its current prices and remove resources when you finish.

Course map: Cloud concepts and building blocks (Modules 1-6), then design, operate, and improve a solution (Modules 7-12). Includes comparisons, exercises with answers, project work, and a glossary.

How to use this course

Read each topic and apply it to a simple service such as a course catalog. Ask what needs to run, what data it stores, who can access it, how people reach it, and what should happen if a component fails. Cloud products change often; learn the idea first, then check the provider documentation for exact options and prices.

By the end, you can: distinguish service and deployment models; explain regions and zones; choose compute, storage, and networking building blocks; describe shared security responsibilities; plan for reliability, monitoring, and cost; and sketch a small cloud design.

Course map

- 1. What cloud computing means
- 2. Service models: IaaS, PaaS, and SaaS
- 3. Public, private, hybrid, and community clouds
- 4. Regions, zones, and edge locations
- 5. Compute: virtual machines, containers, and serverless
- 6. Storage and databases
- 7. Networking and content delivery
- 8. Identity, security, and shared responsibility
- 9. Reliability, scaling, and disaster recovery
- 10. Monitoring, automation, and cost
- 11. Cloud architecture patterns
- 12. Capstone: host a learning catalog

CLOUD CONCEPTS AND BUILDING BLOCKS

1. What cloud computing means

Cloud computing provides network access to configurable resources that can be provisioned and released when needed. Instead of buying every server upfront, a team can request capacity from a provider and pay according to a service model. The provider operates physical facilities and many underlying systems; the customer still chooses how to configure and use resources. Cloud computing changes how resources are obtained, but does not remove the need for design and operations.

At a glance

```
User -> Internet -> Cloud service -> Response
```

Check your understanding

- Why can cloud resources be useful for a changing workload?
- Does using a cloud provider remove all operations work?

Suggested answers

1. Capacity can be requested or adjusted as demand changes.
2. No. Customers still configure, secure, monitor, and operate their workloads.

2. Service models: IaaS, PaaS, and SaaS

Infrastructure as a Service (IaaS) gives access to fundamental computing resources such as virtual machines, networks, and storage. Platform as a Service (PaaS) provides a managed platform for building or running applications. Software as a Service (SaaS) provides a finished application to use. In general, moving toward SaaS means the provider operates more of the stack, while the customer configures the service and manages its data and access.

At a glance

```
IaaS: rent building blocks; you manage OS and app.
PaaS: deploy app; provider manages runtime platform.
SaaS: use complete application; configure users and data.
```

Check your understanding

- Which model gives most control over an operating system?
- Which is a finished application?
- What customer responsibilities remain with SaaS?

Suggested answers

1. IaaS.
2. SaaS.
3. Account access, data, configuration, and appropriate use.

3. Public, private, hybrid, and community clouds

A public cloud is offered to customers over a provider■s shared infrastructure. A private cloud is dedicated to one organization. A hybrid cloud connects distinct environments, such as private infrastructure and public cloud services. A community cloud supports organizations with shared needs. These describe how cloud environments are arranged; the best choice depends on requirements, skills, cost, and policy rather than a label alone.

At a glance

```
Choice depends on: control needs, existing systems,
regulatory constraints, staffing, connectivity, and cost.
```

Check your understanding

- What makes a hybrid approach hybrid?
- Does private cloud automatically mean more secure?
- Name one reason to connect existing systems to a public cloud.

Suggested answers

1. It connects different environments such as private systems and public services.

2. No. Security depends on design, configuration, operations, and people.
3. To add capacity or use a managed service while retaining existing systems.

4. Regions, zones, and edge locations

A region is a geographic area where a provider operates cloud resources. A region may contain multiple zones designed as separate deployment areas. Using more than one zone can improve resilience to a local failure, but the application must be designed to use them. Edge locations place some services closer to users. Choose locations based on latency, data residency, service availability, and recovery needs; exact meanings differ by provider.

At a glance

```
User -> nearby edge cache -> region -> zone A / zone B
```

Check your understanding

- Why might an application use multiple zones?
- What should you consider when choosing a region?

Suggested answers

1. To reduce dependence on one zone and improve availability.
2. User latency, data-location rules, service availability, and recovery objectives.

5. Compute: virtual machines, containers, and serverless

A virtual machine is a software-defined computer with an operating system. A container packages an application and its dependencies while sharing a host operating-system kernel. Serverless services let a provider manage more of the underlying runtime and may charge by requests or execution. Serverless does not mean there are no servers; it means teams manage less infrastructure. Choose based on workload, control, portability, and operations effort.

At a glance

```
Request -> load balancer -> app instances
                         -> managed function
```

Check your understanding

- Which option usually gives the most operating-system control?
- What does a container package?
- Does serverless mean no servers exist?

Suggested answers

1. A virtual machine.
2. An application and its dependencies.
3. No. The provider manages the underlying servers.

6. Storage and databases

Object storage holds files such as images, backups, and course PDFs. Block storage provides disk-like volumes for virtual machines. File storage offers shared folders. Relational databases organize structured data with tables and relationships; NoSQL databases use other data models suited to particular access patterns. Plan retention, access, backup, encryption, and recovery before storing important data.

At a glance

```
Course PDF -> object storage
Catalog metadata -> database
Application files -> compute service
```

Check your understanding

- Where might a course PDF fit?
- What does a relational database organize?
- Why test restoring stored data?

Suggested answers

1. Object storage.

2. Structured records in related tables.
3. To verify that backups can be recovered and are usable.

DESIGN, OPERATE, AND IMPROVE

7. Networking and content delivery

Cloud networking connects resources with virtual networks, subnets, routes, and security rules. Public endpoints can be reached from the internet; private endpoints are limited to approved networks. DNS maps names to service addresses. A content delivery network (CDN) caches suitable content closer to users. Expose only what needs to be public and review network rules as the design changes.

At a glance

```
Browser -> DNS -> CDN (optional) -> public endpoint
                                    -> private app -> database
```

Check your understanding

- Why keep a database endpoint private?
- What does DNS help a user do?
- When can a CDN help?

Suggested answers

1. To limit direct exposure and allow access only from approved services.
2. Reach a service using a readable domain name.
3. When content can be cached closer to users.

8. Identity, security, and shared responsibility

Identity and access management (IAM) determines who or what can perform actions. Use individual identities, multi-factor authentication, least privilege, and managed secrets. Cloud security is shared: the provider protects parts of the service and infrastructure; customers protect their data, identities, configurations, and workloads according to the service model. The exact boundary varies by product, so check provider documentation.

At a glance

```
Identity -> policy -> allowed action -> resource
Prefer: named roles, least privilege, MFA, audit logs.
```

Check your understanding

- What does least privilege mean?
- Who is responsible for customer data and access configuration?
- Why check responsibility for each managed service?

Suggested answers

1. Grant only the access needed for a task.
2. The customer.
3. The boundary changes with service choices and provider agreements.

9. Reliability, scaling, and disaster recovery

Reliability means a service works as expected over time. Scaling changes capacity to handle demand. Horizontal scaling adds instances; vertical scaling increases the size of a resource. Design for component failures with health checks, backups, and tested recovery steps. Recovery Time Objective (RTO) is how quickly service should return; Recovery Point Objective (RPO) is how much recent data loss is acceptable.

At a glance

```
More users -> scale out app instances
Zone failure -> redirect traffic and restore service
RTO: recovery time goal | RPO: data-loss window
```

Check your understanding

- What is the difference between scaling out and up?
- What does RTO describe?

• Why rehearse recovery?

Suggested answers

1. Out adds instances; up increases the size of a resource.
2. The target time to restore service.
3. To find gaps before a real outage.

10. Monitoring, automation, and cost

Monitoring turns metrics, logs, and traces into operational visibility. Metrics show values over time; logs record events; traces follow a request across components. Alerts should signal an actionable problem. Automation and infrastructure as code make environments repeatable. Cloud bills can include compute time, storage, requests, licenses, and data transfer. Budgets, tags, and cleanup routines help avoid surprise charges.

At a glance

```
Measure: request success, latency, errors, storage use
Tag: owner, project, environment
Alert: threshold + owner + response action
```

Check your understanding

• What does a trace help follow?

• Why tag resources?

• Name two cloud cost categories.

Suggested answers

1. One request across multiple services.
2. To identify ownership and group spending.
3. Compute, storage, requests, data transfer, or licenses.

11. Cloud architecture patterns

A workload is the set of resources that deliver a service. Start with needs: users, data sensitivity, expected traffic, availability, budget, and recovery. A simple three-part pattern may use a web front end, application service, and database. Add a CDN for static content, a queue for work that can run later, or a cache for frequently used data only when needs justify them. Keep a clear path for deployment, monitoring, backup, and rollback.

At a glance

```
Users -> CDN / web front end -> app service -> database
                                    -> queue -> worker
Logs + metrics + alerts across the workload
```

Check your understanding

• Why begin with requirements?

• When might a queue help?

• Why not add every cloud service at the start?

Suggested answers

1. They guide trade-offs and prevent unnecessary complexity.
2. To process work asynchronously or absorb bursts.
3. Extra services add cost, failure modes, and operational burden.

12. Capstone: host a learning catalog

Design a cloud-hosted course catalog that serves a website and downloadable PDFs. Sketch the request path, choose where static files and metadata live, identify what must be public, and keep administrative access private. Plan backups, monitoring, a budget alert, and a recovery process. Compare one simple design with a higher-availability option and explain the trade-offs. Do not create paid resources unless you understand the charges and have permission.

At a glance

```
Learner -> DNS -> CDN -> static site + PDF storage
Admin -> MFA -> private management access
Logs/alerts -> owner; backup -> tested restore
```

Check your understanding

- What should be public in the catalog?
- Name one control for administrator access.
- Why compare a simple and a resilient design?

Suggested answers

1. The website and intended course files.
2. MFA, least privilege, and private management access.
3. To make cost, complexity, and availability trade-offs clear.

Architecture workshop

1. Choose a service model

For a static learning site, compare hosting files on a virtual machine, using a managed web host, and adopting a complete hosted site platform. Note who patches the operating system, who deploys the site, and who manages access.

2. Plan for an outage

For a site with downloadable course material, list what must recover first, how often files change, acceptable downtime, where backups live, and how restoration will be tested.

3. Review a cloud bill

Group resources by project and owner. Find idle compute, unused storage, and high data transfer. Propose one change and describe how you will verify that user experience and reliability remain acceptable.

Design checklist
Are requirements written down? Are public and private paths clear? Are identities limited? Can the service recover? Can an owner understand alerts and costs? Is each design choice justified?

Glossary

Term	Meaning
Availability zone	A separate deployment area within a cloud region.
CDN	A network that can cache suitable content nearer to users.
Cloud region	A geographic area containing provider cloud locations.
Container	A packaged application and its dependencies sharing a host kernel.
IaaS	Infrastructure as a Service.
IAM	Identity and access management.
PaaS	Platform as a Service.
RPO	Recovery Point Objective: the acceptable data-loss window.
RTO	Recovery Time Objective: the target time to restore service.
SaaS	Software as a Service.
Scalability	The ability to adjust capacity as workload changes.
Shared responsibility	Security duties split between provider and customer.
Virtual machine	A software-defined computer with its own operating system.
Workload	The resources and processes that deliver a service.

Further study

Cloud products and prices change. Use provider documentation for current service details, regional availability, and rates. The NIST definition gives a provider-neutral foundation.

- NIST. *The NIST Definition of Cloud Computing, SP 800-145.* https://csrc.nist.gov/pubs/sp/800/145/final
- AWS. *Well-Architected Framework pillars.* https://docs.aws.amazon.com/wellarchitected/latest/framework/the-pillars-of-the-framework.html
- Google Cloud. *Geography and regions.* https://docs.cloud.google.com/docs/geography-and-regions
- Microsoft Learn. *Shared responsibility in the cloud.* https://learn.microsoft.com/en-us/azure/security/fundamentals/shared-responsibility

Course completion checklist

I can explain service and deployment models; describe regions, zones, and cloud building blocks; sketch a request path; identify provider and customer security duties; plan for recovery and monitoring; estimate cost categories; and justify a simple cloud design.