

TECH BOOTCAMP

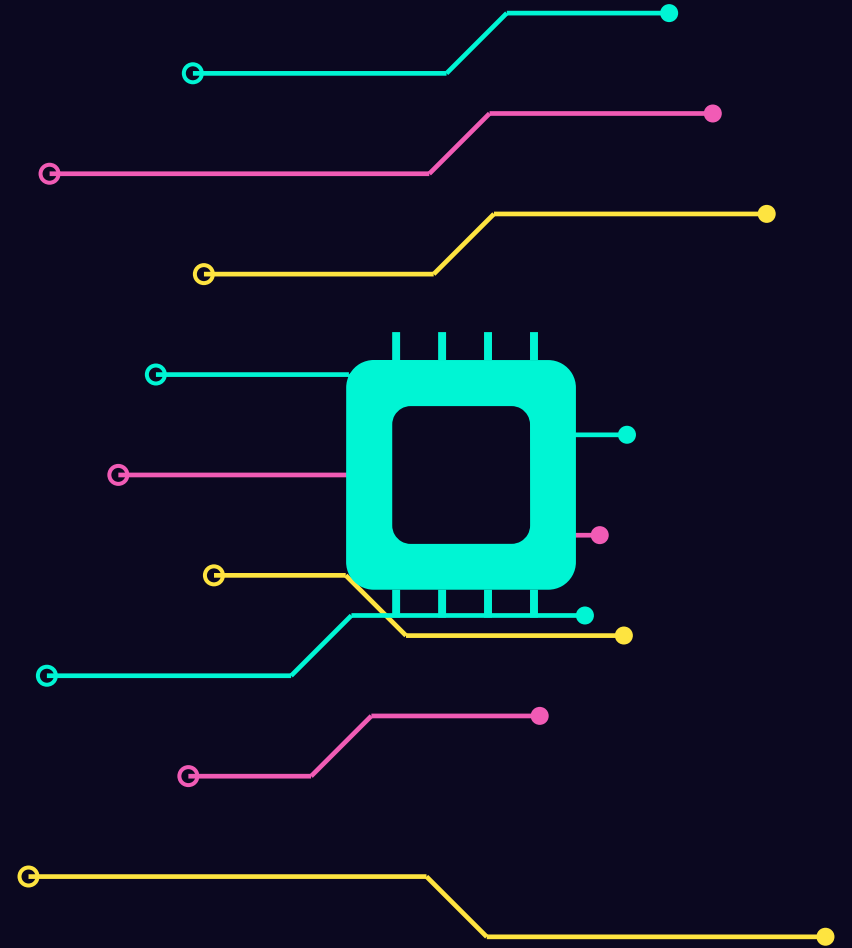
AI & Cloud Bootcamp

Modules from fundamentals to deployment

PRESENTED BY
Presenter Name

MODULES
6

EDITION
2026



Course map

- 01 Edge AI on Devices
- 02 Data Science Essentials
- 03 Web Development in 2026
- 04 Quantum-Safe Cryptography
- 05 Blockchain Explained
- 06 Cloud Computing 101





EDGE AI ON DEVICES

01



Running AI models on phones and sensors

Module 1 · Edge AI on Devices

Overview

- Edge AI runs models directly on devices
- It cuts latency and keeps data private
- Used in phones, cameras, cars and factories
- Small, efficient models make it possible

KEY IDEA

Quantised models can be four times smaller.

How it works



Train

In the cloud



Compress

Prune and quantise



Deploy

To the device



Update

Over the air

Core components



Models

Small, efficient networks.



Hardware

NPU and microcontrollers.



Tooling

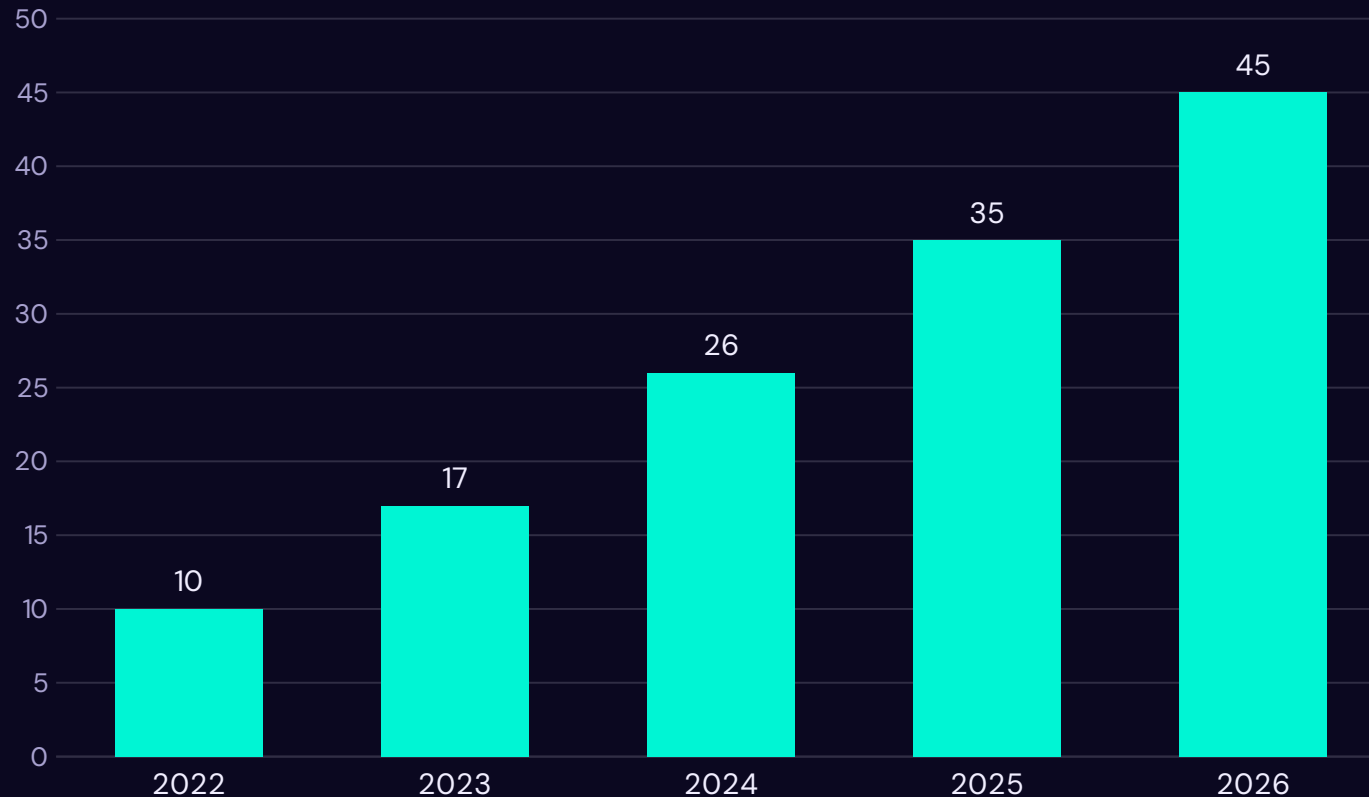
Converters and runtimes.

Tools and options

Layer	Popular choices	Notes
Runtimes	Mobile and embedded runtimes	Optimised inference
Hardware	NPU, GPU, MCU	Match to power budget
Techniques	Quantisation, pruning	Smaller, faster models

Generic examples; pick what fits your project.

Adoption over time



Adoption keeps rising as tools mature (sample data).

Sample data for illustration — replace with your own figures.

Use cases



Image 1

Smartphones

Photo and voice features



Image 2

Security

On-camera detection



Image 3

Industry

Defect inspection

Benefits and risks

Benefits

- ✓ Low latency
- ✓ Privacy by design
- ✓ Works offline

Risks

- Limited compute
- Model updates are harder
- Battery impact

Best practices



Measure first

Profile on real devices.



Compress smartly

Check accuracy after.



Plan updates

Version your models.

Getting started roadmap

Week 1

Learn

Core concepts and a tutorial

Week 3–4

Pilot

One real use case

Week 2

Prototype

Small proof of concept

Month 2

Scale

Standards, monitoring, training

DATA SCIENCE ESSENTIALS

02



From raw data to decisions

Module 2 · Data Science Essentials

Overview

- Data science turns data into insight and predictions
- It combines statistics, programming and domain knowledge
- Used in business, health, sports and government
- Clean data and clear questions matter most

KEY IDEA

Start with the question, not the algorithm.

How it works



Ask

Define the question



Collect

Gather and clean data



Model

Analyse and predict



Share

Visualise and decide

Core components



Data

Sources, quality and storage.



Models

Statistics and machine learning.



Communication

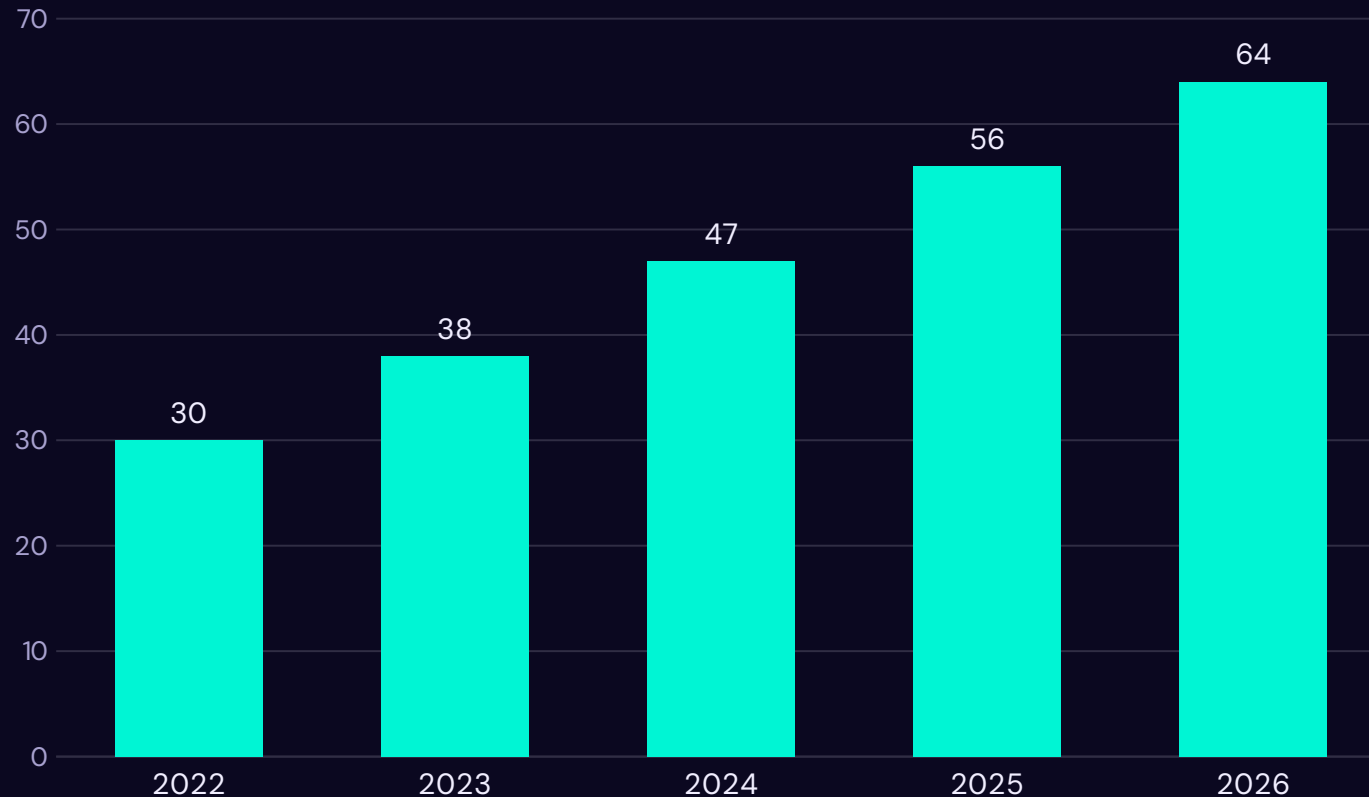
Charts and stories.

Tools and options

Layer	Popular choices	Notes
Languages	Python, R, SQL	Python is most common
Tools	Notebooks, BI dashboards	Share results easily
Libraries	pandas, scikit-learn	Free and open source

Generic examples; pick what fits your project.

Adoption over time



Adoption keeps rising as tools mature (sample data).

Sample data for illustration — replace with your own figures.

Use cases



Image 1

Retail

Demand forecasting



Image 2

Health

Risk prediction



Image 3

Sports

Performance analysis

Benefits and risks

Benefits

- ✓ Better decisions
- ✓ Finds hidden patterns
- ✓ Automates reporting

Risks

- Bad data, bad results
- Privacy concerns
- Skills shortage

Best practices



Clean first

80% of work is preparation.



Keep it simple

Simple models are easier to trust.



Tell a story

One chart, one message.

Getting started roadmap

Week 1

Learn

Core concepts and a tutorial

Week 3–4

Pilot

One real use case

Week 2

Prototype

Small proof of concept

Month 2

Scale

Standards, monitoring, training

WEB DEVELOPMENT IN 2026

03



Modern tools for fast, accessible websites

Module 3 · Web Development in 2026

Overview

- Websites today are apps as much as pages
- Frameworks handle routing, data and rendering
- Performance and accessibility are essential
- Static and server rendering are back in favour

KEY IDEA

**Ship less JavaScript;
your users will thank
you.**

How it works



Design

Mobile-first layouts



Build

Components and pages



Test

Speed and accessibility



Deploy

Global edge hosting

Core components



Front end

HTML, CSS, components.



Back end

APIs and databases.



Hosting

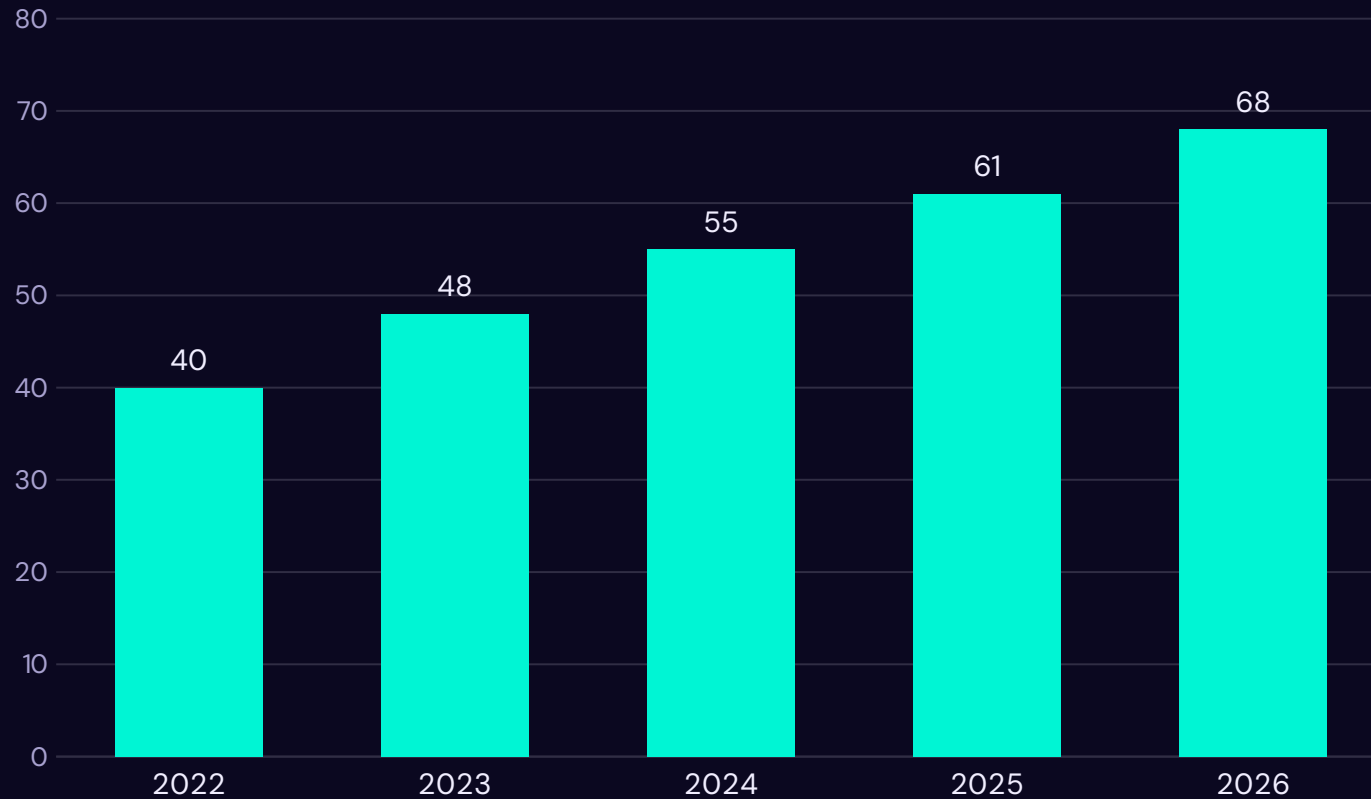
CDNs and edge functions.

Tools and options

Layer	Popular choices	Notes
Frameworks	Component frameworks	Pick one and go deep
Styling	Utility CSS, design tokens	Consistent UI
Hosting	Static and edge platforms	Fast everywhere

Generic examples; pick what fits your project.

Adoption over time



Adoption keeps rising as tools mature (sample data).

Sample data for illustration — replace with your own figures.

Use cases



Image 1

Business sites

Fast, findable pages



Image 2

Web apps

Dashboards and tools



Image 3

E-commerce

Stores that convert

Benefits and risks

Benefits

- ✓ Reach every device
- ✓ Instant updates
- ✓ Huge ecosystem

Risks

- Tooling churn
- Performance pitfalls
- Accessibility often ignored

Best practices



Mobile first

Design for small screens.



Measure speed

Check core web vitals.



Use semantic HTML

Accessibility by default.

Getting started roadmap

Week 1

Learn

Core concepts and a tutorial

Week 3–4

Pilot

One real use case

Week 2

Prototype

Small proof of concept

Month 2

Scale

Standards, monitoring, training

QUANTUM-SAFE CRYPTOGRAPHY

04



Protecting data in the quantum era

Module 4 · Quantum-Safe Cryptography

Overview

- Most encryption relies on hard maths problems
- Future quantum computers could break some of them
- New post-quantum algorithms resist quantum attacks
- Migration takes years, so planning starts now

KEY IDEA

**Data stolen today
could be decrypted
later.**

How it works



Inventory

Find where crypto is used



Assess

Rank systems by risk



Migrate

Adopt new algorithms



Test

Verify and monitor

Core components



Algorithms

Lattice and hash-based schemes.



Standards

New published standards.



Agility

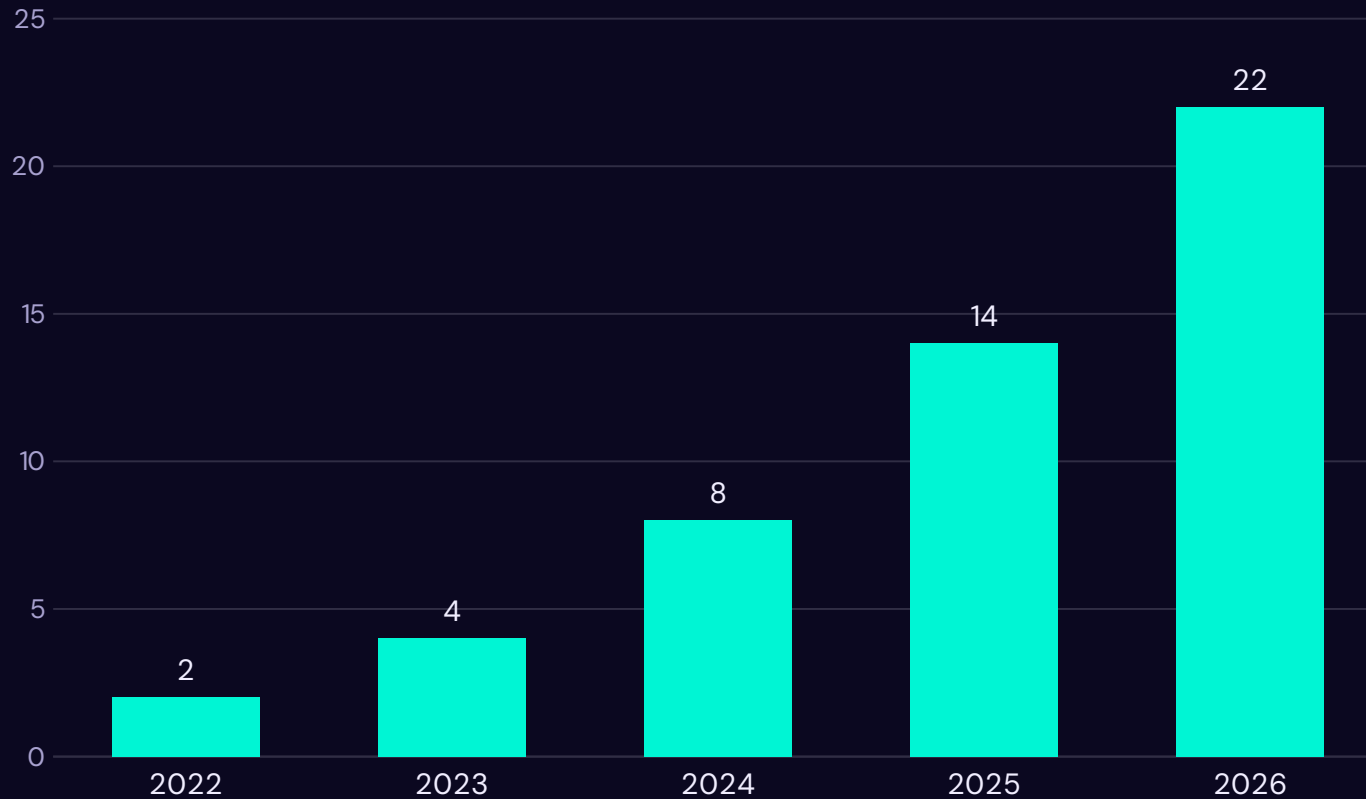
Swap algorithms easily.

Tools and options

Layer	Popular choices	Notes
Key exchange	Post-quantum KEMs	Replace older methods
Signatures	Lattice and hash-based	For certificates
Hybrid	Classical + PQ together	Safe transition

Generic examples; pick what fits your project.

Adoption over time



Adoption keeps rising as tools mature (sample data).

Sample data for illustration — replace with your own figures.

Use cases



Image 1

Banking

Secure transactions



Image 2

Government

Long-term secrets



Image 3

Health

Patient records

Benefits and risks

Benefits

- ✓ Future-proof security
- ✓ Standards available
- ✓ Hybrid mode eases transition

Risks

- Larger keys
- Performance overhead
- Complex migration

Best practices



Map your crypto

Know every use.



Be crypto-agile

Design for change.



Start pilots

Test hybrid modes.

Getting started roadmap

Week 1

Learn

Core concepts and a tutorial

Week 3–4

Pilot

One real use case

Week 2

Prototype

Small proof of concept

Month 2

Scale

Standards, monitoring, training

BLOCKCHAIN EXPLAINED

05



Ledgers, blocks and real use cases beyond crypto

Module 5 · Blockchain Explained

Overview

- A blockchain is a ledger copied across many computers
- New records are grouped into blocks and linked
- Tampering is hard because every copy must agree
- Uses include supply chains, certificates and payments

KEY IDEA

Not every problem needs a blockchain — a database is often enough.

How it works



Core components



Ledger

Append-only shared record.



Consensus

How nodes agree on the truth.



Smart contracts

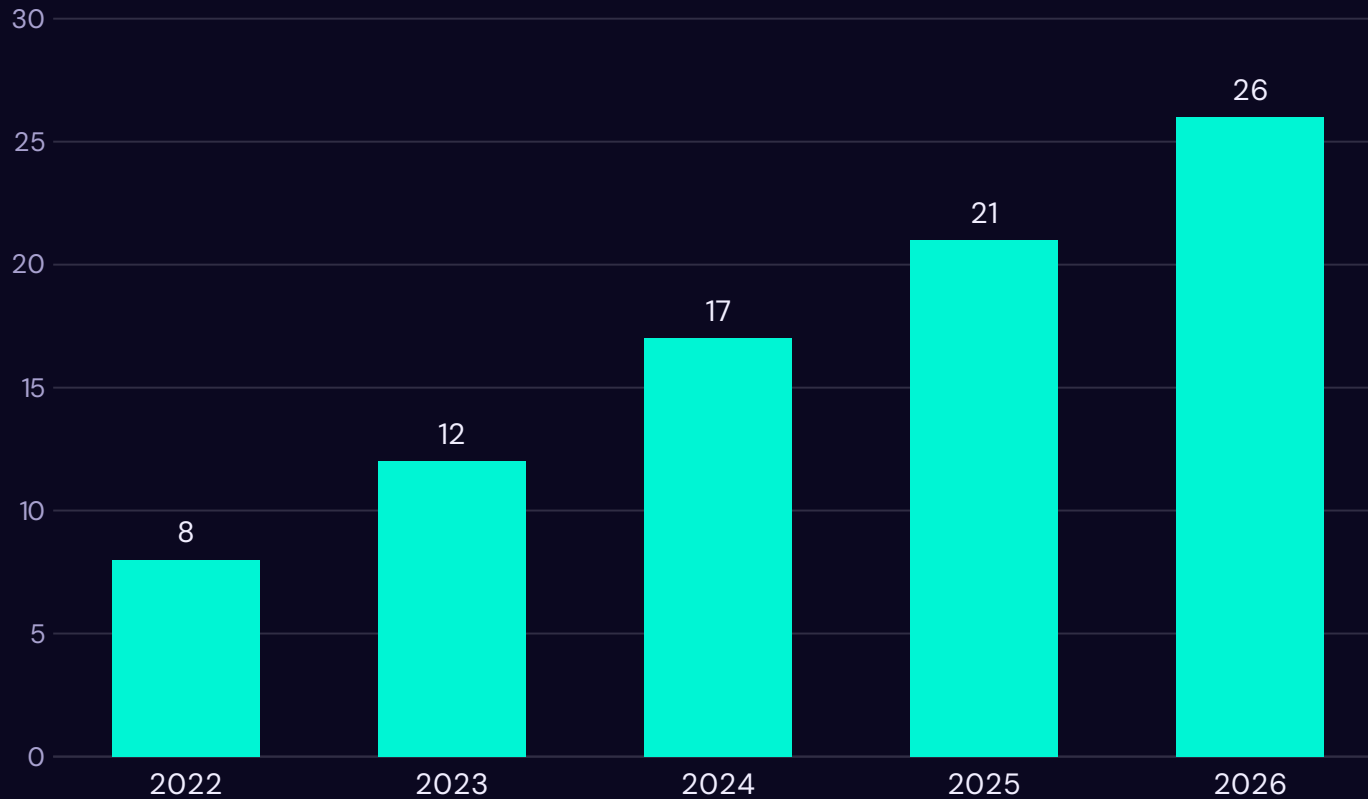
Code that runs on the chain.

Tools and options

Layer	Popular choices	Notes
Public chains	Open networks	Anyone can join
Private chains	Permissioned networks	Known participants
Wallets	Key management apps	Protect private keys

Generic examples; pick what fits your project.

Adoption over time



Adoption keeps rising as tools mature (sample data).

Sample data for illustration — replace with your own figures.

Use cases



Image 1

Certificates

Tamper-proof degrees



Image 2

Supply chain

Track goods end to end



Image 3

Land records

Transparent ownership

Benefits and risks

Benefits

- ✓ Tamper-evident records
- ✓ No single point of control
- ✓ Transparent audit trail

Risks

- Energy and cost concerns
- Hard to fix mistakes
- Regulation still evolving

Best practices



Start small

Pilot one use case.



Mind privacy

Keep personal data off-chain.



Plan governance

Who decides upgrades?

Getting started roadmap

Week 1

Learn

Core concepts and a tutorial

Week 3–4

Pilot

One real use case

Week 2

Prototype

Small proof of concept

Month 2

Scale

Standards, monitoring, training



CLOUD COMPUTING 101

06



From servers in a room to services on demand

Module 6 · Cloud Computing 101

Overview

- Cloud computing delivers servers, storage and software over the internet
- Organisations rent capacity instead of buying hardware
- Services scale up and down with demand
- It enables faster launches and global reach

KEY IDEA

Most savings come from switching off what you are not using.

How it works

01

Choose

IaaS, PaaS or SaaS

02

Deploy

Apps and data to the
cloud

03

Scale

Auto-scale with demand

04

Optimise

Monitor cost and
performance

Core components



Compute

Virtual machines and containers.



Storage

Object, block and file storage.



Networking

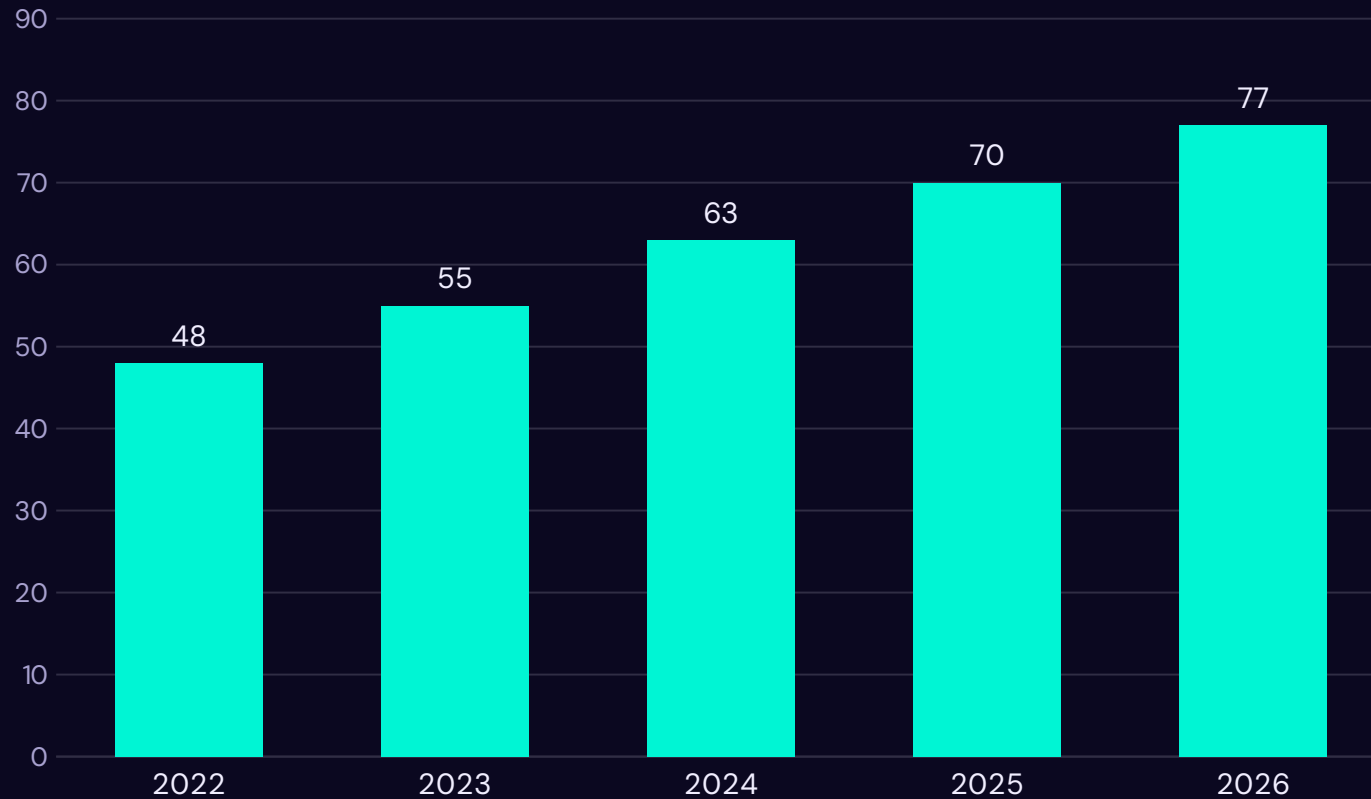
Load balancers, CDNs, VPNs.

Tools and options

Layer	Popular choices	Notes
Compute	VMs, containers, serverless	Match to workload
Data	Managed databases	Backups included
Ops	Monitoring, IaC	Automate everything

Generic examples; pick what fits your project.

Adoption over time



Adoption keeps rising as tools mature (sample data).

Sample data for illustration — replace with your own figures.

Use cases



Image 1

Websites

Hosting that scales with traffic



Image 2

Analytics

Big data on demand



Image 3

Backup

Off-site disaster recovery

Benefits and risks

Benefits

- ✓ No upfront hardware cost
- ✓ Scale in minutes
- ✓ Global availability

Risks

- Costs can creep up
- Vendor lock-in
- Needs new skills

Best practices



Tag resources

Know who owns what.



Set budgets

Alerts before overspend.



Automate

Infrastructure as code.

Getting started roadmap

Week 1

Learn

Core concepts and a tutorial

Week 3–4

Pilot

One real use case

Week 2

Prototype

Small proof of concept

Month 2

Scale

Standards, monitoring, training

Thank you

Questions and discussion.

Presenter Name

hello@yourdomain.in