

MongoDB Administration

Course Duration ~60 Hours · 8 Weeks · Live Online · Lab-Heavy · You Build, Shard, Break and Recover Real Clusters

Curriculum reviewed: July 2026 — vector search, Queryable Encryption and the release cadence are all moving; this syllabus is checked against the official MongoDB documentation, release notes and lifecycle pages

Document Model

Data Modelling

Aggregation

Indexing & Explain

Replica Sets

Sharding

Queryable Encryption

PITR

Atlas & Vector Search

In short: Model it right, scale it wide, and get it back when it breaks. This is the document-database side of the data layer — MongoDB is the most widely adopted document database and the default wherever the data is document-shaped, the schema evolves fast, or the application maps naturally onto nested objects. Free content teaches CRUD and basic queries; it does not teach the two things that actually decide whether a MongoDB project succeeds. First, **data modelling for the document model**: the single biggest source of failed projects is people modelling documents like relational tables, so embedding versus referencing, designing around access patterns rather than normal forms, the document size limit and unbounded-array anti-patterns, schema design patterns and the schema-versioning pattern get a headline module. Second, **operating it in production**: replica sets and elections, sharding and the discipline of choosing a shard key you cannot easily change, indexing strategy, reading explain output, backup and point-in-time recovery, security and monitoring. It is a 2026 syllabus, not a 2016 one: native vector search is now first-class and has been extending beyond the managed cloud service towards self-managed editions, so storing operational data and embeddings together is a real architectural option — quantised vectors, recall-versus-latency tuning and hybrid search get real teaching time, cross-linked to our GenAI and AI Agent courses. Queryable Encryption is taught precisely, including which query types are production-ready and which are still preview. On AI: it writes MongoDB fluently and ships exactly the mistakes that only appear at scale — the value is judgment about modelling, indexing and scale, and there is a block on using AI tooling without shipping its errors. 8 weeks, ~60 hours, lab-heavy — you build, shard, break and recover real clusters.

Duration	8 weeks · ~60 hours · live + hands-on	Format	Live online · lab-heavy · you will build, shard, break and recover real clusters
Prerequisite	Basic command line comfort and familiarity with some programming or query language. No prior MongoDB or NoSQL experience required.	Curriculum	9 modules, from the document model to a defended production cluster
Certification	Mapped primarily to the MongoDB Associate Database Administrator credential, with the associate developer and data-modeler paths as adjacent options	Ideal for	Fresh IT graduates, backend developers using MongoDB who were never taught to model or operate it, relational DBAs adding document skills, DevOps/SRE inheriting MongoDB, and AI engineers using vector search

Anyone Can Insert a Document. The Job Is the Cluster Underneath.

Anyone can insert a document. Designing documents around how the application actually reads them, and keeping a sharded cluster fast, available and recoverable when it holds real data — that is the job.

Free MongoDB content is abundant and teaches CRUD and basic queries well. What it does not teach are the decisions that are hard to reverse, so those are foregrounded here:

- **Data modelling for the document model** — the single biggest source of failed MongoDB projects is people modelling documents like relational tables. Embedding versus referencing, designing around access patterns rather than normal forms, and the schema-versioning pattern for data that has to keep evolving. A headline module, not an appendix.
- **Unbounded array growth** — documents that grow without limit until they collide with the maximum document size, long after the design decision that caused it
- **Aggregation pipelines that ignore indexes** — a pipeline that is correct on a thousand documents and scans the whole collection on ten million

- **Compound-index field order** — the ordering that decides whether an index is used at all, taught as a method with explain output as the evidence
- **Shard keys you cannot easily change** — the highest-consequence decision in a MongoDB deployment, and the one most often made casually
- **Backups nobody has restored** — point-in-time recovery rehearsed as a drill with a measured recovery time, because an untested backup is not a backup

Document-database mistakes are often invisible until scale, and expensive after it.

AI Writes MongoDB Fluently — And Ships the Mistakes That Only Appear at Scale

AI writes MongoDB queries and schema code fluently, and produces exactly the things that hurt in production: schema designs that model documents like relational tables and miss the point of the document model entirely, unbounded array growth that eventually runs into the document size limit, aggregation pipelines that ignore indexes, missing or wrongly ordered compound indexes, and shard key choices that cannot be undone. None of these fail immediately. They fail at scale, when the data is already there.

The value this course builds is **judgment about data modelling, indexing and operating at scale** — what an assistant cannot supply, because it cannot see your access patterns, your document growth, your index usage or your recovery objectives. A short, practical block covers checking what AI proposes against explain output, real document sizes and the queries your application actually issues, before any of it reaches production.

MongoDB or PostgreSQL? Not a War — a Choice

Progmatrix runs both this course and a PostgreSQL Administration course, and the honest engineering position is that **neither database "wins."** They suit different problems, most large organisations run both, and knowing when each is the right tool is itself a senior skill.

- **PostgreSQL Administration** — the relational and SQL world: strong consistency, joins across normalised data, a rich extension ecosystem, and the default for transactional systems where the schema is stable and the relationships matter.
- **MongoDB Administration** — the document and NoSQL world: flexible schema, horizontal scale by design, and the default where the data is document-shaped, the schema evolves fast, or the application maps naturally onto nested objects.
- **Take both if you can.** Relational DBAs adding document skills, and MongoDB engineers who have never operated a relational system, are both more employable for it. The operational fundamentals rhyme; the modelling and scaling decisions do not.

You will not find "NoSQL versus SQL" framing here. That argument was settled by the industry running both, and repeating it would only date the course.

A 2026 Syllabus, Not a 2016 One

- **MongoDB as an AI database** — native vector search is a first-class capability and, in the current release line, has been extended beyond the managed cloud service towards self-managed editions. Storing operational data and embeddings together in one system is now a real architectural option. Combined with quantised vectors, which trade a controlled amount of recall for substantially lower memory use, this makes MongoDB a credible retrieval-augmented-generation backend. Cross-linked to our Generative AI & LLM Engineering, AI Agent Development and Prompt Engineering courses — the same converged-workload story the PostgreSQL course tells from the relational side.
- **Queryable Encryption** — running queries against encrypted fields without the server ever seeing plaintext, a strong position for regulated industries. Equality and range queries are supported for production use; additional string query types have been arriving through public preview and are explicitly not for production until they reach general availability. You learn where that line currently sits, how to check it yourself against the official documentation, and how to design so a preview feature never becomes a production dependency by accident.
- **The release cadence, and what it means for upgrade planning** — a major release on a multi-year cycle with a long support lifecycle, plus more frequent minor releases between them, and those minor releases are now downloadable for self-managed Community and Enterprise editions rather than being cloud-only. That is how search, vector search and the newer encryption query types are reaching self-hosted deployments. The trade-offs are real: minor releases must be upgraded sequentially, the previous minor release stops receiving patches once a new one lands, and not every feature is available on that track. Upgrade planning is taught against the official lifecycle and release notes rather than against version numbers that go stale.

- **Query-engine work in the current line** — including cost-based plan selection becoming the default fallback during query planning, plus time-series and resharding improvements. What that means for you: plan choice is more automatic, so explain output matters more, not less. We deliberately quote no headline performance percentage — published figures for the same release vary widely by workload and source, and a number you cannot reproduce on your own workload is not useful to you.
- **The managed cloud service** as the dominant way MongoDB is now run. It automates provisioning, scaling, patching, backup and index advice. It does not automate your data model, your shard key, your index strategy, your security posture or your recovery objectives — those stay yours, which is precisely why this course exists.

Hands-on Projects & Portfolio

Data-Modelling Workshop Redesign a relational schema for the document model around real access patterns, and justify every embed-versus-reference decision.	Aggregation Pipeline Build Turn a hard reporting requirement into an efficient, index-using pipeline.
Index & Query Optimisation Take a slow workload and fix it with evidence from explain output, before and after.	Replica Set & Induced Failover Build the set, fail it over under load, recover it and document what happened.
Sharded Cluster Exercise Shard a growing collection, choose and defend a shard key, and observe balancing.	Security Hardening to a Baseline Authentication, RBAC, TLS and encryption — with a justification for each choice.
Disaster Recovery Drill Point-in-time restore to just before a destructive change, with a measured recovery time.	Vector Search Implementation Semantic retrieval running alongside operational data in the same system.
Capstone A single node taken to a sharded, secured, monitored, recoverable production cluster — and every decision defended.	

Tools & Stack

MongoDB · mongosh · MongoDB Compass · Aggregation Pipeline · Replica Sets · Sharding · MongoDB Atlas · Atlas Vector Search · Queryable Encryption · mongodump / mongorestore · Ops Manager · Database Tools · MongoDB Kubernetes Operator · Prometheus · Grafana · Docker · a driver in Python or Node.js · Linux

Certification Alignment

- Mapped primarily to the **MongoDB Associate Database Administrator** credential — the administration-focused certification this syllabus most directly prepares you for
- The **MongoDB Associate Developer** and **MongoDB Associate Data Modeler** paths are adjacent options; the modelling, querying and aggregation modules here overlap substantially with both. An **Associate Atlas Administrator** path also exists for the managed-service side.
- Certification names, structure and format change — **verify the current line-up on MongoDB's own certification pages before you register**. This syllabus deliberately states no exam format, question count or duration, because those details move.
- Presented as preparation and alignment only — no official partnership or accreditation implied, and no exam outcome guaranteed
- An honest note: this field hires primarily on **demonstrated operational ability** — a cluster you have modelled, sharded, broken and recovered speaks louder than any certificate, which is why this course produces exactly that evidence

Career Outcomes

MongoDB is the most widely adopted document database and a default choice for applications with flexible or fast-evolving schemas, and its move into native vector search has extended its relevance into AI application backends — so the skill now spans operational and AI workloads. Document-database administration is a distinct specialisation from relational DBA work: the operational instincts transfer, but the modelling and scaling decisions do not, which is why the two are complementary rather than interchangeable. And database operational fundamentals — modelling, indexing, scaling, recovery — are unusually durable against technology churn.

MongoDB Database Administrator · Database Administrator (NoSQL) · Database Reliability Engineer · Data Platform Engineer · Backend Engineer owning the data layer · DevOps/SRE with database ownership · Cloud Database Engineer

Who This Course Is For

- Final-year students and fresh IT graduates targeting database or backend roles
- **Backend developers using MongoDB who were never taught to model or operate it properly** — the most common and most consequential gap this course closes
- Relational DBAs adding document-database skills — complementary to what you already know, not a replacement for it
- DevOps, SRE and platform engineers inheriting MongoDB responsibility
- Data engineers who need operational depth beyond writing queries
- **AI and application engineers using MongoDB vector search** who need to run it properly
- Anyone preparing for a MongoDB administration certification

Curriculum — 9 Modules

Module 1: The Document Model & Architecture

5 Hours

YOU CONFIGURE

A working deployment you connect to and inspect — and the mental model of how documents are actually stored that makes every later modelling and indexing decision obvious

TOPICS

- Documents, collections and BSON; how the document model differs from the relational model, and when that difference is a genuine advantage versus a trap
- The storage engine and how data is held on disk; the mongod process and what it does
- Installing a deployment, connecting with the shell and the GUI client, and orienting yourself in a running system

Assessment: Quiz + Lab

Module 2: Data Modelling — the Module That Decides Success

8 Hours

YOU DESIGN

A document model built around how the application actually reads the data — with every embed-versus-reference decision justified, and the growth behaviour of each document understood before there is data in it

TOPICS

- Embedding versus referencing, and the criteria that decide between them; modelling around access patterns rather than normal forms
- One-to-many and many-to-many relationships in a document world; the document size limit and unbounded-array anti-patterns — the failure that arrives long after the decision that caused it
- Established schema design patterns; the schema-versioning pattern for evolving data without downtime; schema validation to enforce structure where you want it

Assessment: Quiz + Data-Modelling Workshop

Module 3: Querying & Aggregation**7 Hours****YOU BUILD**

An aggregation pipeline that answers a hard reporting requirement *and* uses indexes instead of scanning the whole collection

TOPICS

- The query language, projections and operators; updates including array and nested-document updates
- The aggregation pipeline in depth — stages, grouping, joins across collections, transformations, and the pipeline patterns worth knowing by heart
- Writing aggregations that use indexes: stage order, what can be pushed down, and where a pipeline quietly stops being efficient

Assessment: Quiz + Lab

Module 4: Indexing & Query Performance**8 Hours****YOU TUNE**

A slow workload taken apart with a systematic method — the cause identified from explain output, the fix applied, and the improvement proved with evidence rather than folklore

TOPICS

- Index types — compound, multikey, text and wildcard — and how compound-index field order decides whether an index gets used at all
- Covered queries; index intersection; reading explain output as the core diagnostic skill
- The cost of over-indexing on write performance and storage; a repeatable method for fixing a slow query

Assessment: Quiz + Optimisation Lab

Module 5: Replication & High Availability**6 Hours****YOU RECOVER**

Service continuity through a node failure — a rehearsed failover under load, treated as a drill rather than a diagram

TOPICS

- Replica sets, primary election and automatic failover; adding and removing members safely
- Read and write concerns and exactly what they trade off; read preferences and when each is appropriate
- The oplog, replication lag, and diagnosing a secondary that is falling behind

Assessment: Quiz + Failover Drill

Module 6: Sharding & Horizontal Scale**7 Hours****YOU SCALE**

A growing collection out across a sharded cluster — with a shard key you chose deliberately, defended out loud, and understand you cannot easily change later

TOPICS

- When to shard and — just as important — when not to; shard keys and the discipline of choosing one you cannot easily change; hashed versus ranged strategies
- Chunks and the balancer; targeted versus scatter-gather queries and why the shard key determines which you get; zone sharding
- Resharding and what it costs; the operational realities of running a sharded cluster day to day

Assessment: Quiz + Sharded Cluster Exercise

Module 7: Security & Compliance**6 Hours****YOU SECURE**

A deployment hardened to a defensible baseline — authentication, least-privilege roles, TLS and encryption — with a written justification for every control

TOPICS

- Authentication and authorisation; role-based access control and least privilege; network hardening and TLS
- Encryption at rest; auditing; field-level and client-side encryption
- **Queryable Encryption** — querying encrypted data without the server seeing plaintext, with an accurate account of which query types are production-ready and which are still in preview, and how to check that yourself against the official documentation

Assessment: Quiz + Hardening Lab

Module 8: Operations, Backup & Recovery**7 Hours****YOU RECOVER**

A database to a moment just before a destructive change — a graded point-in-time restore with a measured recovery time, because an untested backup is not a backup

TOPICS

- Monitoring the metrics that actually matter; capacity planning; connection management and pooling
- The standard backup tools; point-in-time recovery; tested restore drills with measured recovery objectives; upgrade planning across the release cadence
- Diagnosing the incidents that really happen — a runaway query, a lagging secondary, a full disk, an unbalanced shard

Assessment: Quiz + Graded Restore Drill

Module 9: Atlas, AI Workloads & Capstone**6 Hours****YOU DELIVER**

One deployment taken from a single node to a sharded, secured, monitored, recoverable cluster — with a vector search workload running alongside the operational data, and every decision defended

TOPICS

- The managed cloud service: what it automates — provisioning, scaling, backup, index advice — and what stays yours; the performance advisor and automated index guidance; serverless and tiering; cost control
- **Vector search for AI applications** — indexing embeddings, quantised vectors and the memory they save, recall-versus-latency tuning, and hybrid search alongside operational data
- Running MongoDB on Kubernetes with the operator; capstone: single node to sharded, secured, monitored, recoverable cluster — and defend every decision

Assessment: Capstone Project + Defence