Question 1 of 15

What is the correct order of magnitude for operations per second that a modern in-memory cache can handle?

1 10,000 ops/sec

2 1,000,000 ops/sec

3 100,000 ops/sec

4 1,000 ops/sec

Correct!

Modern in-memory caches like Redis can handle over 100,000 operations per second per instance. Understanding this capability helps avoid premature scaling decisions - many systems that seem to need distributed caching can actually run on a single high-performance cache instance.

Question 2 of 15

Which storage type provides sub-millisecond data access latency?

1 Magnetic disk

2 Network storage

3 In-memory cache

4 SSD storage

Correct!

In-memory caches like Redis provide sub-millisecond latency by keeping data in RAM, while SSD storage typically provides 5-30ms latency, and other storage types are significantly slower.

Question 3 of 15

Which is NOT typically the first bottleneck in modern application servers?

1 CPU utilization

2 Network bandwidth

3 Memory capacity

4 Response latency

Incorrect.

Modern application servers have 64-512GB of RAM standard, making memory capacity rarely the first constraint. CPU utilization typically becomes the bottleneck before memory, network, or latency limits are reached.

Question 4 of 15

Modern memory-optimized servers can handle terabytes of data in memory.

1 True

2 False

Correct!

Current memory-optimized instances like the Amazon EC2 U7i High Memory can provide up to 24TB of RAM, allowing entire large datasets to be kept in memory for ultra-fast access patterns that were impossible just a few years ago.

Question 5 of 15

Which factor BEST indicates when database sharding becomes necessary?

1 Using cloud hosting

2 Dataset approaching 50+ TiB

3 More than 1000 users

4 Having multiple tables

Correct!

Modern single database instances can handle up to 64+ TiB and tens of thousands of transactions per second. Sharding becomes necessary when approaching these actual hardware limits, not arbitrary user counts or architectural choices.

Question 6 of 15

A 400GB dataset requires cache sharding for optimal performance.

1 True

2 False

Correct!

Not necessarily. Modern memory-optimized cache instances can handle up to 1TB of data. A 400GB dataset fits comfortably within a single cache instance, avoiding the complexity of sharding while maintaining excellent performance.

Question 7 of 15

What causes engineers to over-engineer systems during design?

1. Using outdated hardware constraints ✓
2. Following security best practices
3. Implementing proper monitoring
4. Writing clean code

Correct!

When engineers use hardware assumptions from 2015-2020, they dramatically underestimate modern capabilities, leading to unnecessarily complex distributed solutions where simple architectures would suffice.

Question 8 of 15

A system processing 5,000 writes per second needs message queue buffering.

1 True

2 False

Correct!

Modern well-tuned database instances can handle 10-20k writes per second. At 5,000 WPS, the database can handle the load directly without requiring message queue complexity for buffering.

Question 9 of 15

When optimizing for sub-millisecond response times, which approach works best?

1 Database indexing

2 SSD optimization

3 Network tuning

4 In-memory caching

Correct!

Sub-millisecond response times require in-memory storage to avoid disk I/O entirely. SSDs provide 5-30ms latency, which is too slow for sub-millisecond requirements.

Question 10 of 15

What is the correct order of magnitude for storage capacity that a single modern database instance can handle?

1 10 TB

2 1 TB

3 100 GB

4 100 TB

Correct!

Modern single database instances can handle up to 64+ TiB of storage, with some configurations supporting even more. This represents a massive increase from older systems and means many applications don't need database sharding until they reach truly massive scale.

Question 11 of 15

Which scenario does NOT require database sharding?

1. 2TB dataset with simple queries ✓
2. Cross-region user base
3. Geographic data distribution
4. Backup window constraints

Correct!

A 2TB dataset with simple queries can be handled by a single modern database instance. Geographic distribution, cross-region requirements, and operational concerns like backup windows are valid reasons for sharding.

Question 12 of 15

Message queues with sub-5ms latency can be used in synchronous request flows.

1 True

2 False

Correct!

Modern high-performance message queues achieve 1-5ms end-to-end latency, making them fast enough to use within synchronous APIs while gaining benefits of reliable delivery and decoupling.

Question 13 of 15

For a system with 10 million businesses at 1KB each, which storage approach is most appropriate?

1 Multiple cache layers

2 Single database instance

3 Microservice architecture

4 Distributed database cluster

Correct!

10 million businesses at 1KB each equals only 10GB of data. Even accounting for indexes and related data, this easily fits within a single modern database instance without requiring distributed complexity.

Question 14 of 15

A single optimized application server instance typically supports approximately how many concurrent connections?

1 1,000 connections

2 100,000 connections

3 1,000,000 connections

4 10,000 connections

Correct!

Modern application servers with optimized configurations can handle over 100,000 concurrent connections per instance. This capability means that connection limits are rarely the first bottleneck - CPU utilization typically becomes the constraint before running out of connection capacity.

Question 15 of 15

What is the typical network latency for communication within a single cloud region?

1 20-50 milliseconds

2 5-10 milliseconds

3 Under 1 millisecond

4 1-2 milliseconds

Correct!

Within a single cloud region, network latency typically ranges from 1-2 milliseconds. This predictable low latency enables reliable distributed system design and real-time communication between services in the same region.