1. How Do Big Systems Store Data?

   Large-scale applications like Instagram, Uber, and Amazon utilize a combination of SQL and NoSQL databases to manage their vast and diverse data needs.

   Instagram uses PostgreSQL (a relational database) to store user data, photos, and metadata. To scale, it uses sharding, which splits data across multiple servers.

   Uber uses MySQL for structured data and Cassandra and Redis (NoSQL) for real-time analytics and caching. This hybrid model supports both transactional accuracy and high-speed operations.

   Amazon combines Amazon Aurora (a MySQL/PostgreSQL-compatible SQL database) with DynamoDB (NoSQL) to handle structured and unstructured data at scale.

   SQL Server Use Case: While not common among big tech companies, SQL Server is used in enterprise settings like banks and healthcare. Its strong transactional support and integration with Microsoft tools make it ideal for secure, structured data environments.

2. Do Video Games Use Databases?

   Yes! Online and multiplayer video games rely heavily on databases to store player data.

   Games like PUBG and Fortnite use backend databases for player profiles, match histories, in-game purchases, and progress tracking. Many use NoSQL databases for fast, scalable access.

   Relational Database Example: Some Massively Multiplayer Online Games (MMOGs) use SQL databases for account management, inventory systems, and billing—where data integrity and complex queries are required.

3. Can a Database Work Without the Internet?

   Yes—databases can operate fully offline.

Offline Use Cases: Systems in military bases, submarines, airplanes, and remote outposts often run databases locally, without internet access.
SQL Server Offline Capability: Microsoft SQL Server can be installed and run offline, ideal for high-security or isolated environments. It operates within local networks and supports air-gapped deployments.

4. What Makes Online Stores So Fast?

   E-commerce platforms like Amazon and Noon use advanced database strategies to stay fast and reliable.

   Hybrid Databases: They combine SQL for transactions (like orders and payments) and NoSQL for flexible data (like product listings and user sessions).

   Cloud Infrastructure: Using cloud databases allows them to scale automatically, handling millions of products and thousands of simultaneous users.

   Challenges: They must handle huge product volumes, real-time search, and ensure data consistency across global regions.

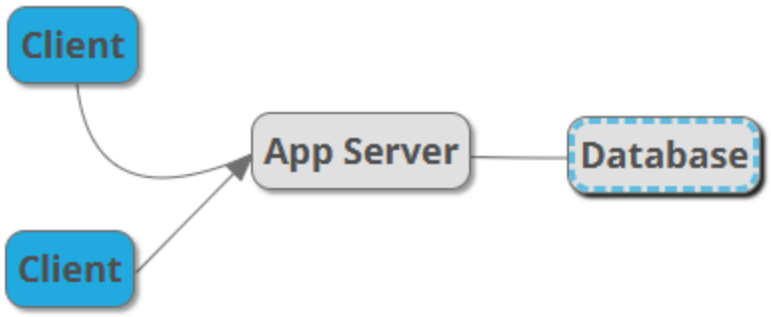
5. How Do Banks Protect Their Data?

   Banks use multiple layers of protection in their database systems.

   Security Features: Include encryption, user access controls, data masking, and auditing to track access and changes.

   Common Systems: Many banks use Oracle or SQL Server because they meet strict compliance and security standards, such as PCI-DSS and GDPR.

6. Draw Me the System



❖ Client Tier (Presentation Layer):

This is what users interact with directly

Examples: Web browsers, mobile apps, desktop applications

Responsibilities: Displays information and collects user input

❖ Application Server Tier (Logic Tier):

The "brains" of the operation

Contains the core business logic and application functionality

Processes requests from clients and retrieves/stores data from the database

Examples: Node.js server, Java EE application, .NET application

❖ Database Server Tier (Data Tier):

Responsible for data storage and retrieval

Ensures data integrity and security

Examples: MySQL, PostgreSQL, MongoDB, Oracle

Runs on a separate server for security and performance