Database Fundamentals, Compliance, and IAM

🧩 1. Introduction to Databases

A **database** is an organized collection of data stored and managed electronically.
It allows you to store, retrieve, modify, and delete data efficiently.

🔹 Example:

Think of a **library**:

- Books = data
- Shelves = tables
- Librarian = Database Management System (DBMS)

You can easily find, update, or borrow books because everything is structured — same idea applies to databases.

🧱 2. What is a Database Management System (DBMS)?

A **DBMS** is software used to manage databases.
It ensures data consistency, integrity, and security.

🔸 Common DBMS Examples:

- **Relational Databases:** MySQL, PostgreSQL, Oracle, SQL Server
- **NoSQL Databases:** MongoDB, Cassandra, DynamoDB
- **Cloud Databases:** Amazon RDS, Azure SQL Database, Google Cloud SQL

🧮 3. Types of Databases

1. Relational Database (RDBMS):

- Data stored in **tables (rows and columns)**
- Tables are related using **primary and foreign keys**
- Uses **SQL (Structured Query Language)**

Example:
A “Customers” table linked to an “Orders” table.

CustomerID Name Email

1 John john@email.com

OrderID CustomerID Amount

101 1 2500

Here, CustomerID is the link between both tables.

---

2. NoSQL Database:

- Flexible schema (no fixed structure)
- Good for big data and real-time analytics
- Types: Document, Key-Value, Graph, Column-based

Example:
MongoDB stores data as JSON documents:

```
{
  "name": "John",
  "email": "john@email.com",
  "orders": [101, 102]
}
```

---

3. In-Memory Databases:

- Store data in RAM for ultra-fast access
- Example: **Redis**, **Memcached**

---

4. Cloud Databases:

- Managed services hosted by cloud providers
- Examples:
    - AWS RDS, DynamoDB
    - Azure SQL Database, Cosmos DB
    - Google Cloud Firestore, BigQuery

These handle **backups, scaling, patching, and security** automatically.

🧠 4. Core Concepts of Databases

Concept	Description	Example
Table	Collection of related data	Users table
Row (Record)	One complete entry	User #1
Column (Field)	Attribute of data	Name, Email
Primary Key	Unique identifier	User ID
Foreign Key	Reference to another table	Order's CustomerID
Query	Command to interact with data	SELECT * FROM users;
Schema	Structure/blueprint of DB	Tables, columns, constraints

---

🔐 5. Database Security and Compliance

Databases often store sensitive information such as:

- Personal data (PII)
- Financial records
- Credentials
- Healthcare data

Hence, **security and compliance** are critical.

⚙️ 5.1 Database Security Practices

Practice	Description
Encryption	Protect data at rest (storage) and in transit (network)
Access Control	Limit who can read/write data
Auditing	Track changes and access logs
Backup & Recovery	Regular data backups to prevent data loss

Practice	Description
Patching & Updates	Keep DB software up-to-date
Network Security	Use firewalls, VPCs, and private endpoints

📜 6. Database Compliance Standards

Compliance means following **industry laws and regulations** for handling data securely.

Here are key standards:

Compliance	Full Form	Focus Area	Example Use Case
GDPR	General Data Protection Regulation	EU data privacy	User consent, right to delete data
HIPAA	Health Insurance Portability and Accountability Act	Healthcare data	Hospitals, clinics
PCI DSS	Payment Card Industry Data Security Standard	Financial transactions	E-commerce storing card info
SOX	Sarbanes-Oxley Act	Financial reporting integrity	Public companies
ISO 27001	International Security Standard	Information security management	Enterprise compliance
FedRAMP	Federal Risk and Authorization Management Program	US government cloud data	Gov agencies using cloud

---

🛡️ 6.1 Common Compliance Measures in Databases

- **Data Masking:** Hiding sensitive fields (e.g., credit card numbers)
- **Role-based Access:** Only specific users see or edit certain tables
- **Audit Trails:** Keeping logs of who accessed or modified data
- **Encryption Keys Management:** Using services like AWS KMS, Azure Key Vault
- **Retention Policies:** Automatically deleting old or irrelevant data

---

👥 7. Identity and Access Management (IAM) in Databases

🔑 What is IAM?

IAM (Identity and Access Management) controls *who can access what* in a database or cloud system.

📚 Example:

A company database has three users:

- **Admin:** Full control (create, delete, modify tables)
- **Developer:** Can query and insert data
- **Analyst:** Read-only access

IAM ensures each person only does what they are authorized to.

⚙️ 7.1 IAM Components

Component	Description	Example
Identity	The user or service	Developer account
Authentication	Verify identity	Password, MFA, SSO
Authorization	Assign permissions	SELECT, INSERT access
Roles & Policies	Group permissions logically	"ReadOnly" role
Auditing	Track usage & changes	Login logs

☁️ 7.2 IAM in Cloud Databases

Platform	IAM Example
AWS RDS	IAM roles to control DB access, integrate with AWS IAM users
Azure SQL DB	Azure AD authentication, role assignments
GCP Cloud SQL	IAM policies + service accounts for database management

---

💡 Example: Azure SQL IAM Integration

1. **Azure AD user** logs into SQL Database
2. Azure verifies identity
3. User role defines permissions:
    - Reader: SELECT only
    - Contributor: INSERT, UPDATE, DELETE
    - Owner: All permissions
4. All actions logged in **Azure Monitor** for compliance

🧩 8. IAM Best Practices for Databases

1. **Use Least Privilege Principle:**
   Give users minimum access needed for their role.
2. **Implement MFA (Multi-Factor Authentication):**
   Protect login with password + token.
3. **Rotate Access Keys Regularly**
4. **Use Role-Based Access Control (RBAC)** instead of hardcoded credentials.
5. **Enable Logging & Auditing** to track activities.
6. **Integrate with Centralized Identity Provider (like Entra ID, Okta, AWS IAM).**

📊 9. Real-World Example

Scenario:
An e-commerce company uses AWS RDS for its orders database.

Role	Responsibility	Permissions
Admin	Manage structure, users	Full Access
Developer	Create new APIs	Read + Write
Data Analyst	Generate reports	Read Only

Security Controls:

- Database encrypted using AWS KMS
- Backups stored in S3 with IAM policy restriction

- Audit logs sent to CloudWatch
- Access only via company VPN

This ensures both **security** and **compliance (PCI DSS)**.

📜 10. Summary

Area	Key Points
Database	Organized system for storing and managing data
DBMS	Software to interact with database
Compliance	Legal standards to secure sensitive data
IAM	Controls user identities and permissions
Best Practice	Encrypt data, audit access, use least privilege

---

🚀 Next Steps for Beginners

1. Learn basic **SQL** commands (SELECT, INSERT, UPDATE, DELETE).
2. Practice with **MySQL** or **PostgreSQL** locally or in cloud (AWS RDS, Azure SQL).
3. Explore **database encryption** and **user management**.
4. Understand **IAM setup** in AWS or Azure portals.
5. Review compliance basics for your industry (GDPR, HIPAA, PCI DSS, etc.).