



# Introduction to Databases

UNDERSTANDING DATABASE CONCEPTS, QUERYING,  
AND ORGANIZATION

# How Databases Power Everyday Apps

- ▶ Almost every app you use relies on databases!
- ▶ Examples:
  - Social Media (Twitter, Instagram) → Stores user profiles, posts, messages.
  - E-Commerce (Amazon, eBay) → Manages products, orders, user accounts.
  - Streaming Services (Netflix, Spotify) → Tracks watch history, recommendations.
  - Banking Apps → Securely store transactions, balances, and authentication.

# Basics of SQL

- ▶ SQL (Structured Query Language) is used to manage and query databases.
- ▶ Common SQL commands:
  - SELECT: Retrieve data
  - INSERT: Add new data
  - UPDATE: Modify existing data
  - DELETE: Remove data
  - CREATE TABLE: Define a new table
  - JOIN: Combine data from multiple tables

# Organizing a Database

- ▶ -Normalization: Reduce redundancy
- ▶ -Indexing: Improve query speed

# Organizing a Database

-Before Normalization

order_id	customer_name	customer_email	product_name	quantity
1	Alice	alice@email.com	Laptop	1
2	Bob	bob@email.com	Phone	2
3	Alice	alice@email.com	Keyboard	1

-After Normalization

## Customers Table

customer_id	customer_name	customer_email
1	Alice	alice@email.com
2	Bob	bob@email.com

## Orders Table

order_id	customer_id	product_name	quantity
1	1	Laptop	1
2	2	Phone	2
3	1	Keyboard	1

# Organizing a Database

Before indexing

```
SELECT * FROM users WHERE email =  
'alice@email.com';
```

After indexing

```
CREATE INDEX idx_email ON users(email);  
  
SELECT * FROM users WHERE email =  
'alice@email.com';
```

# Sharding & Scaling

- ▶ Horizontal Scaling (Sharding)
- ▶ Vertical Scaling (More CPU/RAM)

# Database Backups & Recovery

- ▶ Importance of backups
- ▶ Incremental vs. full backups



# User Authentication Using SQL

- ▶ How does a website verify your login?

- Authentication flow:

1. User enters their email & password.

2. The app queries the database:

```
```sql
```

```
SELECT * FROM users WHERE email = 'user@example.com';
```

```
```
```

3. The stored (hashed) password is compared to the entered password.

4. If they match, access is granted.

- ▶ Security considerations:

- **\*\*Hash passwords\*\*** using bcrypt, SHA-256, etc.

# Hands-on Activity: Querying a Database

- ▶ Upload the CSV file to SQLite Online.
- ▶ Run the first query that I sent to discord because it won't work without it....

# Hands-on Activity: Querying a Database

- ▶ Because a new table was made called users, we can delete the old one

# Hands-on Activity: Querying a Database

- ▶ Because a new table was made called users, we can delete the old one

```
DROP TABLE sample_database;
```

# Hands-on Activity: Querying a Database

- ▶ To view the whole table

# Hands-on Activity: Querying a Database

- ▶ To view the whole table

```
SELECT * FROM users;
```

# Hands-on Activity: Querying a Database

- ▶ To see only names in the table

# Hands-on Activity: Querying a Database

- ▶ To see only names in the table

```
SELECT name FROM users;
```



# Hands-on Activity: Querying a Database

- ▶ To see users that are more than age 25

# Hands-on Activity: Querying a Database

- ▶ To see users that are more than age 25

```
SELECT * FROM users WHERE age > 25;
```

# Hands-on Activity: Querying a Database

- ▶ To see users that signed up after 2023-06-01

# Hands-on Activity: Querying a Database

- ▶ To see users that signed up after 2023-06-01

```
SELECT * FROM users WHERE signup_date > '2023-06-01';
```