Stateless APIs

Definition: A stateless API does not retain any information about the client or its previous requests between calls. Each request from the client contains all the necessary information the server needs to process it.

Characteristics:

1. **Independent Requests**: Every request is self-contained and independent. There's no reliance on past interactions.
2. **No Server-Side Session**: The server does not store any session data about the client.
3. **Simplified Scalability**: Statelessness makes it easier to scale the application horizontally because any server can handle any request without needing to synchronize session data.
4. **Common Protocol**: REST (Representational State Transfer) APIs are typically stateless.

.

Examples:

- A REST API that requires the client to send authentication credentials (e.g., an API key or token) with every request.
- API endpoints like GET https://dog.ceo/api/breeds/list/all that return the same response regardless of prior interactions.
- An e-commerce site using a REST API where each request includes the user's token for authentication.

Stateful APIs

Definition: A stateful API maintains information (state) about the client between requests. The server keeps track of the client's interactions to provide context for subsequent requests.

Characteristics:

1. **Session-Based Interaction**: The server relies on session data to handle requests. Clients might use cookies, tokens, or a session ID to identify themselves.
2. **Server Dependency**: Clients often depend on the same server or cluster to ensure their state persists.
3. **Complex Scalability**: Scaling stateful systems requires mechanisms like session replication or sticky sessions to ensure the client's state is accessible.
4. **Use Cases**: Stateful APIs are often used in scenarios requiring multiple steps or transactions (e.g., banking systems, shopping carts).

.

1. **Examples**:

- A login-based system where the client logs in, and the server remembers their session for subsequent actions like `POST /checkout`.
- A WebSocket-based API that maintains a continuous connection between client and server.
- An online banking application where a user's session is maintained to securely complete multi-step transactions like fund transfers.