

## ONE PAGE SUMMARY

IT & Software > Backend Development > WebSockets

# Introduction to WebSockets

WebSockets enable real-time communication between clients and servers.

## DEFINITION

WebSockets are a protocol that allows for full-duplex communication channels over a single TCP connection. They are designed for real-time web applications, enabling instant data exchange.

## KEY CONCEPTS

### Full-Duplex Communication

WebSockets allow data to be sent and received simultaneously, unlike traditional HTTP.

### Persistent Connection

Once established, a WebSocket connection remains open, reducing latency for ongoing data exchange.

### Low Overhead

WebSockets have less overhead than HTTP, making them more efficient for frequent messages.

### Event-Driven

WebSockets use events to trigger actions, allowing for responsive applications.

### Cross-Origin Communication

WebSockets can facilitate communication between different domains, enhancing application interoperability.

## EXAMPLES

- Real-time chat applications using WebSockets for instant messaging.
- Live sports updates where scores are pushed to users without refreshing the page.
- Collaborative editing tools that reflect changes made by users in real-time.

## MEMORY TIPS

- ★ Think of WebSockets as a phone call, where both parties can talk at the same time.
- ★ Remember 'WebSockets = Web + Sockets' for real-time connections.
- ★ Visualize a highway with no traffic lights for constant data flow.

## COMMON MISTAKES

- ✗ Confusing WebSockets with HTTP, thinking they operate the same way.
- ✗ Neglecting to handle connection closures and errors properly.
- ✗ Forgetting to implement security measures like WSS for encrypted connections.

## QUICK RECAP

WebSockets provide a persistent, low-latency connection for real-time data exchange. They enable full-duplex communication, making them ideal for interactive applications. Remember their efficiency and event-driven nature for effective backend development.