BaseWsInstance

August 17, 2026

BaseWsInstance

Kind: Interface

Source: `packages/websockets/adapters/ws-adapter.ts`

Part of: Websockets

`BaseWsInstance` defines the minimal WebSocket-like contract required by the WS adapter. Implementations must support event subscription through `on` and expose a `close` function so the adapter can listen for lifecycle events and terminate connections consistently.

Properties

Property	Type
`on`	`(event: string, callback: Function) => void`
`close`	`Function`

Diagram

```
graph LR
  Adapter[WS Adapter] --> Instance[BaseWsInstance]
  Instance --> On["on(event, callback)"]
  Instance --> Close["close()"]
  On --> Events[Connection and message events]
  Close --> Cleanup[Connection cleanup]
```

Usage

```
import type { BaseWsInstance } from './ws-adapter';

function registerConnection(socket: BaseWsInstance) {
  socket.on('message', (data: unknown) => {
```

```
    console.log('Received message:', data);
  });

  socket.on('close', () => {
    console.log('Connection closed');
  });

  // Close the connection when it is no longer needed.
  socket.close();
}
```

AI Coding Instructions

- Treat `BaseWsInstance` as an adapter boundary; only rely on `on` and `close` when writing transport-agnostic WebSocket code.
- Register event listeners with the expected event names supported by the underlying WebSocket implementation.
- Ensure `close` is called during shutdown, error handling, or resource cleanup paths.
- Avoid depending on implementation-specific socket APIs unless the adapter interface is explicitly extended.

Relationships

- IMPORTS → `INestApplicationContext`
- IMPORTS → `WebSocketAdapter`
- IMPORTS → `WsMessageHandler`
- IMPORTS → `isFunction`
- IMPORTS → `NestApplication`