**Socket.io**

**const server = http.createServer(app);**

* This line creates an HTTP server using Node.js's built-in **http** module. The server is configured to handle requests using the Express.js application you created in the previous step (**app**). This is how you link your Express app to the HTTP server.

**const io = new SocketIO(server);**

* This line initializes a Socket.IO server and associates it with your HTTP server (**server**). Socket.IO is a library that allows for real-time, bidirectional communication between the server and connected clients using WebSockets and other transport mechanisms. The **io** object is an instance of the Socket.IO server.

**io.on('connection', socket => {**

* This line sets up an event handler for the 'connection' event. The 'connection' event is a built-in event in Socket.IO and is automatically triggered when a client establishes a connection to the server.

**socket => {**

* This is the callback function that will be executed when a client connects to the server. It takes a **socket** object as its parameter. The **socket** object represents the connection between the server and the connected client.

Difference Between Socket.io and io.on

**socket.on** and **io.on** are both event handling methods provided by the Socket.IO library, but they are used in different contexts and have different purposes:

1. **socket.on**:
   * **socket.on** is used to set up event listeners on an individual socket connection, representing a specific client's connection to the server.
   * When you use **socket.on**, you are setting up event handlers for events that are specific to a single client. These events pertain to communication between the server and that particular client.
   * For example, you might use **socket.on** to handle events like 'message' when a client sends a message to the server or 'disconnect' when a client disconnects from the server.

**io.on**:

* **io.on** is used to set up event listeners on the global Socket.IO server object.
* When you use **io.on**, you are setting up event handlers for events that are global to the entire Socket.IO server. These events typically pertain to server-level events and broadcasting messages to all connected clients.
* For example, you might use **io.on** to handle events like 'connection' when a new client connects to the server or 'broadcast' when you want to send a message to all connected clients.

In summary, **socket.on** is used to set up event handlers for specific client connections, whereas **io.on** is used for server-level event handling and broadcasting messages to all connected clients. They serve different purposes in managing real-time communication in a Socket.IO application.