**A screenshot of a computer

AI-generated content may be incorrect.**

**IoT-SenseTilt Node**  
This node represents a custom sensor input that monitors tilt or orientation data in real time. In a cloud architecture, data from IoT-SenseTilt can be sent upstream to cloud services for analytics, alerting, or long-term storage. By integrating this sensor node into Node-RED, you can easily process the incoming stream (e.g., filtering, transforming, or forwarding) before pushing it to other services.

**2. Timestamp Node**  
The Timestamp node generates a time-based message, often used as a trigger or heartbeat in flows. In a cloud context, it can schedule periodic tasks—such as polling sensors, refreshing data, or sending routine status updates to remote databases or dashboards. It ensures the system can operate autonomously, performing actions at fixed intervals without manual intervention.

**3. Samaritan Node**  
In the IoT cloud architecture, this node receives commands from the cloud (e.g., via MQTT or HTTP endpoints) or respond to sensor data (like temperature or tilt)

**4. Button Node**  
The Button node provides a user interface element for manually injecting messages into the flow. In a cloud setup, this can be used for quick tests, manual overrides, or user-driven events. For instance, an operator might click the button in the Node-RED dashboard to send a command to the fan or to log a specific event to the cloud.

**5. Slider Node**  
The Slider node is another UI element that outputs a numerical value. In an IoT solution, this can adjust settings in real time—such as fan speed, sensor thresholds, or other parameters—without requiring code changes. The updated slider value can be passed directly to connected devices or forwarded to cloud-based storage or analytics services.

**6. Chart Node**  
The Chart node visualizes data in real time. In a cloud-based scenario, it can display sensor readings (from IoT-SenseTilt) or user interactions (from the Button or Slider) on a local dashboard. It also complements remote dashboards or analytics tools by providing immediate, on-premises feedback. This helps operators quickly assess device status or verify that sensor values are within expected ranges before or alongside sending them to the cloud.

**7. Debug Node**  
Finally, the Debug node is indispensable for troubleshooting and verifying data flow. In a production cloud environment, data might travel through numerous processing steps, so being able to inspect messages at various points ensures correct functionality. Developers use the Debug node to confirm that sensor readings, user inputs, or commands to devices are formatted and routed correctly.

Overall, these nodes together form a small but illustrative portion of an IoT cloud architecture. Node-RED acts as a central integration layer, where sensor data, user inputs, and device controls converge before being routed to cloud services for storage, analytics, and further action. The combination of visual UI elements (Button, Slider, Chart), device-specific nodes (IoT-SenseTilt, Samaritan), and core Node-RED utilities (Timestamp, Debug) provides a flexible, modular foundation for building robust and scalable IoT solutions.

**A graph with a line

AI-generated content may be incorrect.**