Q.2) Submit a report on the detailed study of comparison between web sockets and webRTC's.

A.2) The WebSocket is a computer communications protocol, which provides full-duplex communication channels over a single TCP connection. The WebSocket is designed to work over HTTP port 80 and 443, hence making it compatible with the HTTP protocol. Most browsers support the WebSocket protocol, including Google Chrome, Firefox, etc. Basically, WebSockets are a bidirectional mechanism for browser communication. It is used to send unsolicited messages from the server to the client (which HTTP cannot perform). It is generally implemented by using the WebSocket API in development.

- WebRTC (Or Web Real Time Communication) is a set of protocols and APIs that enable real time communication over peer-to-peer connection. WebRTC uses real time protocols to transfer video and audio over peer connection. WebRTC is open source and is supported on Chrome, Firefox, etc.
Basically, WebRTC enables audio and video communication to work inside web pages. This works without plugins.

Also, WebRTC allows access to devices of different varieties. It allows the access of the laptop microphone, camera, or screen – all in real time.
WebRTC is not just limited to audio & video – it allows sending of any arbitrary data.

* WebSockets vs WebRTC

1. WebRTC has no signalling channel to the client or server. When starting a WebRTC session, the connection to the client/server is not managed by the WebRTC. Instead, in a browser, it is usually done by WebSocket or HTTP. Hence, in a way, WebSockets complement WebRTC.

2. WebRTC and WebSockets use different protocols in the transport layer. While WebRTC works with UDP, WebSockets work with TCP. Hence, WebSocket is designed for reliable communication.

3. While both WebSockets and WebRTC allow full-duplex communication, WebSockets allow it ~~over~~ between a browser and a web server [client-and-server], and WebRTC allows it between two browsers [peer-to-peer].

4. Due to the different build in the architecture of WebRTC and WebSocket, they are used in differing scenarios. WebSockets are used for sport updates, financial tickers, multiplaya games, etc. while WebRTC are used in real-time marketing and advertising, social networking, etc.

5. WebRTC is designed for high-performance, high quality communication of video, audio, and arbitrary data. WebSockets are designed for bidirectional communication between client and server. Streaming A/V content on WebSockets is not as efficient as doing the same on WebRTC.

6. WebSockets can be very efficient if application requires frequent messages, such as notification systems. WebRTC is useful in applications where high volume of data is transported and reliability is a secondary concern and few frames can be sacrificed in favour of response time and delivering something.