GET is used to request data from a specified resource. To send a GET request, a client needs to specify the URL of the resource it wants to retrieve. The request is then sent to the server, which processes the request and sends the requested data back to the client.

POST sends data to a server to create or update a resource. For example, it is often used to submit an HTML form to a server. To send a POST request, a client needs to specify the URL of the resource to which it wants to send data and the data itself. The request is then sent to the server, which processes the request and sends a response back to the client.

The POST method is often used to submit forms or upload files to a server.

Difference-

- When using GET, data parameters are included in the URL and visible to everyone. However, when using POST, data is not displayed in the URL but in the HTTP message body.
- GET is less secure because the URL contains part of the data sent. On the other hand, POST is safer because the parameters are not stored in web server logs or the browser history.
- GET requests can be cached and remain in the browser history, while POST requests cannot. This means GET requests can be bookmarked, shared, and revisited, while POST requests cannot.
- GET method is limited to a maximum number of characters, while the POST method has no such limitation. This is because the GET method sends data through the resource URL, which is limited in length, while the POST method sends data through the HTTP message body, which has no such limitation.
- GET method supports only string data types, while the POST method supports different data types such as string, numeric, binary,

Content negotiation is the process of selecting the best resource for a response when multiple resource representations are available. Content negotiation happens when a client specifies the media type it wants as a response to the request `Accept` header. By default, ASP.NET Core Web API returns a JSON formatted result and it will ignore the browser Accept header.

HTTPS is HTTP with encryption and verification. The only difference between the two protocols is that HTTPS uses TLS (SSL) to encrypt normal HTTP requests and responses, and to digitally sign those requests and responses. As a result, HTTPS is far more secure than HTTP.