**REQUEST**

**Request Message:**

**CRLF (Carriage Return Line Feed) -** contains three values which are separated by spaces

**Request CRLF**

1. Method
2. Request Target
3. Protocol Version

**Process**:

* + - 1. Message Body
      2. Empty Line
      3. Payload

\*HTTP 1.0 has at least one required header which is request header

\*Message headers: General, Request, Entity Header

\*Payload - found after the header, blank line means empty payload

**Request Methods**

1. Get – retrieves, transfers a selected resource into the message body aka entity. Commonly used method.
2. Head – identical to get but doesn’t get the resource. It has no payload, only headers and metadata.
3. Post – non-safe and non-idempotent method. Request that the target resource perform resource-specific processing of the representation enclosed in the message body of the request.

* Typically used in HTML form data.
* Any data that is being submitted is already in the payload aka message body and the information will not be in the URL.

\*Search-friendly URL – doesn’t contain any query.

1. Put – create or replace the state of the target resource with the state defined by the representation enclosed in the request message payload i.e. target resource.

* Web servers, by default, do not allow put request since it may affect the information.

1. Delete – removes the association between the target resource and its current functionality. This is only a logical delete and does not necessarily delete the true resource.
2. Options – allows you to query a particular resource on the server itself. Its response is usually other request methods.

* This is another way to validate a resource

\*A request may be an asterisk (\*), in which case the request applies to the server in general rather than a specific resource.

1. Trace – echoes back to the user or client what is happening to the request or received request message. It can manipulate the message.

* Typically used for testing, diagnosing the request, response chain and troubleshooting.

\*Response chain – set of nodes from client to server.

1. Connect – establishes the request of a tunnel to the destination origin server. It is commonly used to create an end-to-end virtual connection.

\*Request Method Resource: RFC 7231, section 4: Request methods

\*Link Rot – linking to different pages but these pages aren’t controlled by you so some links may not exist anymore and if not updated, the website will degrade due to dead links. Solution: Update.

**Method Properties:**

1. Safe Methods -
2. Idempotent Methods
3. Cacheable Methods

**RESPONSE**

**Response Message:**

**Response CRLF:**

1. Protocol Version
2. Status Code
3. Reason Phrase

Example: HTTP/1.1 200 OK

**Process**:

1. Message Body
2. Empty Line
3. Payload