**Request - Response Best Practices**

**Request in REST:**

**===============**

**- To make a valid Request, the client needs to include 4 things:**

**1. URL (Uniform Resource Locator)**

**2. Method**

**3. List of Headers**

**4. Body**

**URL:**

**===**

**URLs became an easy way for the client to tell the server which things it wants to interact, called resources.**

**Method:**

**=======**

**The method request tells the server what kind of action the client wants the server to take. The 4 most commonly used or seen in APIs are:**

**1. GET -- Asks the server to retrieve a resource**

**2. POST -- Asks the server to create a new resource**

**3. PUT -- Asks the server to edit/update an existing resource**

**4. DELETE -- Asks the server to delete a resource**

**Headers:**

**=======**

**Headers provide meta-information about a request.**

**Body:**

**====**

**The request body contains the data the clients wants to send the server.**

**Ex: URL --** [**http://abc.com**](http://abc.com/)

**Method -- POST**

**Headers -- user-agent**

**Body -- data**

**Response in REST:**

**================**

**The server response with a status code. Status codes are 3 digit numbers**

**Ex: Status Code -- 200(OK)**

**Headers -- content-type:text**

**Body -- data**

**Data Formats:**

**============**

**A well-designed format is dictated by what makes the information the easiest for the intended audience to understand. The most common formats found APIs are**

**1. JSON**

**2. XML**

**JSON -- It is very simple format that has two pieces -- ‘Key’ and ‘Value’ Ex: {“name”:”api”}**

**XML -- It provides few building blocks. The main block is called node. XML always starts with root node, inside that will have child nodes.**

**How Data Formats are used in HTTP:**

**==================================**

**Using Headers we can inform the server what information we are sending to it and what we are expecting in return.**

**Content:type : When the clients send the content-type its saying what format the data is.**

**Accept: The Accept header tells the server what data-format it is able to accept.**