**HTTP response codes**

**1xxx->Informational**

**2xxx->Success**

200**-** Request Succeeded (GET, DELETE,PUT,PATCH)

201**-**Successfully created new record (POST)

**202 (Accepted)**:

**A 202 response is typically used for actions that take a long while to process. It indicates that the request has been accepted for processing, but the processing has not been completed.** (Ex: Receiver system REST API service need to process a batch request, it takes time to process the request, so it acknowledge the client that it received request successfully and it takes to process by returning 202 http response code)

**Idempotence**:

From a RESTful service standpoint, for an operation (or service call) to be idempotent, clients can make that same call repeatedly while producing the same result. In other words, making multiple identical requests has the same effect as making a single request.

Ex: GET/PUT/DELETE

**PUT VS POST**:

->The difference between POST and PUT is that PUT requests are idempotent. That is, calling the same PUT request multiple times will always produce the same result. In contrast, calling a POST request repeatedly have side effects of creating the same resource multiple times.

->Response from PUT is non catchable whereas response from POST is catchable

**HTTP Methods**:

GET method is for fetching the records

POST method is for creating new record

PUT and PATCH are for creation/updating an existing record, PUT overrides the existing record whereas PATCH partially updates the record.

Ex: Employee record has Job info, personal details and address ….if I want to update only address details .By using PATCH method ,we just need to send new address details ,If I want to use PUT method ,I need to send all his information (Job info, personal details and address) because it overrides the existing record with new record.

**3xxx->Redirect**

304**-**Nothing was modified by the request

**4xxx->Client-side issue**

**401-Unauthorized** (Credentials are not valid)

**403**-Credentials are valid, but **don't have permission to access the resource**

**404-Resource not found (**Wrong URI)

**405-Method Not Allowed (**A read-only resource could support only GET)

In header's

Content-Type ->application/json

**406 (Not Acceptable)**: It is when you can't send what they want

For example, a client request for data formatted as application/xml will receive a 406 response if the API is only willing to format data as application/json

**415 (Unsupported media type)**: When they send what you don't want

Content-Type, Content-Encoding

**429 Too Many Requests** - The error is used when there may be DOS attack detected or the request is rejected due to rate limiting

**5xxx->Server-side issue**

**500** (Internal server Error):

The **server has encountered a situation it doesn't know how to handle**.

**503** (**Service Unavailable**):

The server is not ready to handle the request. Common causes are a **server that is down for maintenance or that is overloaded**.

**504 (Gateway Timeout)**:

This error response is given when the server is acting as a gateway and **cannot get a response in time**. (**Ex: Receiver system service has 5 min timeout set, if it couldn't process the request in 5 minutes ,it returns error with '504' http response code back to user.**)

---------------------------------------------------------------------------------------------

Difference b/w SOAP based webservices and REST based webservices

|  |  |
| --- | --- |
| **SOAP** | **REST** |
|  | **Rest is light weight, widely used for mobile and web applications**  ->REST doesn't require XML parsing and takes less bandwidth |
| SOAP only supports **XML** | Supports both **XML** and **JSON** |
| ->It's easy to handle SOAP responses compared to REST responses, because in SOAP we don't need to read success/failure HTTP response codes and act accordingly  ->In REST ,we need to read HTTP response codes and determine whether it's success or failure ,if it's failed ,is that because of client side issue or service side issue and act accordingly.    ->SOAP has Request/Response/Fault messages (Exception are sent in fault response]  ->REST HTTP Response codes  200 series-Success  400 series-Client-side issues  500 series-Server-side issues |  |
| Authentication: **Basic** (Credentials), **Client certificate-based authentication** | Authentication: **Basic** (Credentials), **Client certificate-based authentication, OAUTH** |
| Schema used -WSDL | Schema used -EDMX or XSD file |

Webservices:

SOAP based webservices

REST based webservices