**Idempotent :**  An idempotent HTTP method is a method that can be invoked many times without the different outcomes. It should not matter if the method has been called only once, or ten times. The result should always be the same. For example, in arithmetic, adding zero to a number is an idempotent operation. The following HTTP methods are idempotent: **GET, HEAD, OPTIONS, TRACE, PUT and DELETE**. **A PATCH is not necessarily idempotent**

1. POST is **NOT** idempotent.
2. GET, PUT, DELETE are idempotent.

**POST** : So when we invoke the same POST request N times, we will have N new resources on the server. So, **POST is not idempotent**.

**PUT** : If you invoke a PUT API N times, the very first request will update the resource; the other N-1 requests will just overwrite the same resource state again and again – effectively not changing anything. So, **PUT is** **idempotent.**

**DELETE** : When you invoke N similar DELETE requests, the first request will delete the resource and the response will be 200 (OK) or 204 (No Content). Other N-1 requests will return 404 (Not Found). Clearly, the response is different from the first request, but there is no change of state for any resource on the server-side because the original resource is already deleted. **DELETE is** **idempotent.**

**GET** : GET method NEVER change the resource on the server. They are purely for retrieving the resource. So, invoking multiple requests will not have any write operation on the server, so **GET is idempotent**.