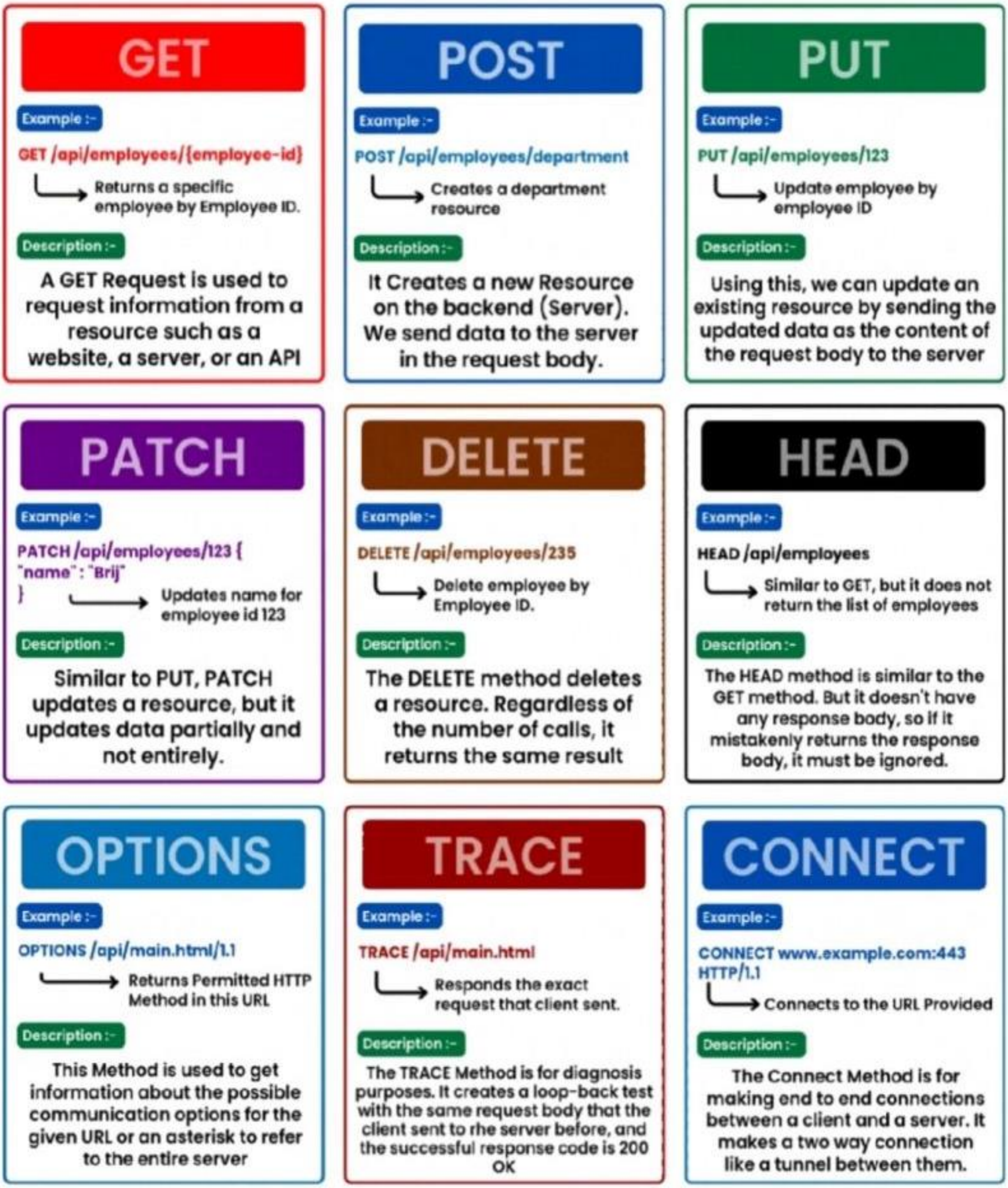
HTTP METHODS



GET

Example :-

GET /api/employees/{employee-id}

Returns a specific employee by Employee ID.

Description :-

A GET Request is used to request information from a resource such as a website, a server, or an API

POST

Example :-

POST /api/employees/department

Creates a department resource

Description :-

It Creates a new Resource on the backend (Server). We send data to the server in the request body.

PUT

Example :-

PUT /api/employees/123

Update employee by employee ID

Description :-

Using this, we can update an existing resource by sending the updated data as the content of the request body to the server

PATCH

Example :-

PATCH /api/employees/123 {
"name" : "Brij"
}

Updates name for employee id 123

Description :-

Similar to PUT, PATCH updates a resource, but it updates data partially and not entirely.

DELETE

Example :-

DELETE /api/employees/235

Delete employee by Employee ID.

Description :-

The DELETE method deletes a resource. Regardless of the number of calls, it returns the same result

HEAD

Example :-

HEAD /api/employees

Similar to GET, but it does not return the list of employees

Description :-

The HEAD method is similar to the GET method. But it doesn't have any response body, so if it mistakenly returns the response body, it must be ignored.

OPTIONS

Example :-

OPTIONS /api/main.html/1.1

Returns Permitted HTTP Method in this URL

Description :-

This Method is used to get information about the possible communication options for the given URL or an asterisk to refer to the entire server

TRACE

Example :-

TRACE /api/main.html

Responds the exact request that client sent.

Description :-

The TRACE Method is for diagnosis purposes. It creates a loop-back test with the same request body that the client sent to rhe server before, and the successful response code is 200 OK

CONNECT

Example :-

CONNECT www.example.com:443 HTTP/1.1

Connects to the URL Provided

Description :-

The Connect Method is for making end to end connections between a client and a server. It makes a two way connection like a tunnel between them.