EXTRA CREDIT REPORT - [Tutorial: Build RESTful APIs with Node and Express | Mosh](https://www.youtube.com/watch?v=pKd0Rpw7O48&amp=&t=2626s)

Most if not all applications these days follow the client/server architecture. The client or app itself must communicate with the server using the HTTP protocol. Rest or Representational State Transfers is essentially a convention to build these HTTP services. We can use simple HTTP protocol principles to create, read, update and delete data – referred to as CRUD operations.

Standard HTTP Methods are Get, Post, Put and Delete. Get requests responds with an array of objects, Put requests update information, Delete deletes information and Post adds a new piece of information.

We handled two HTTP Get requests in the tutorial, one which returned a full list of courses, and a second that implemented a new endpoint by adding an id parameter at the end of the path. After doing that we had routes to return both the full array and a second for a specific course. We used HTTP Post requests in order to add a new course to the collection of courses. To handle a Put request, we added a new route handler “app.put,” that looked up a course, returned a 404 error if not existing, validated the request and then updated and returned the course. For the final CRUD operation we implemented a Delete request with the “app.delete” method.

The tutorial also went over some debugging and provided additional best practices for cleaner, less cluttered code. In this section, Mosh shared that the proper way to exit a route handler is by returning the response and using a “;” to exit the function. This same technique was used in validating the request as well. You can also incorporate the response “return” into the middle of one line of code, eg. directly after the if (), making the line cleaner and removing the unneeded code block.