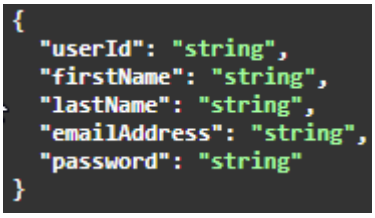
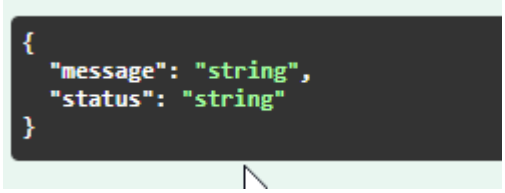
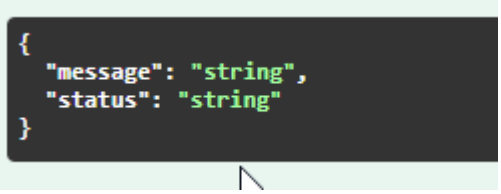
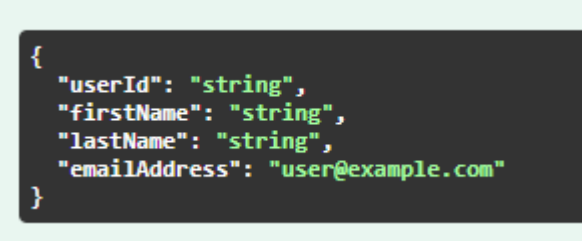
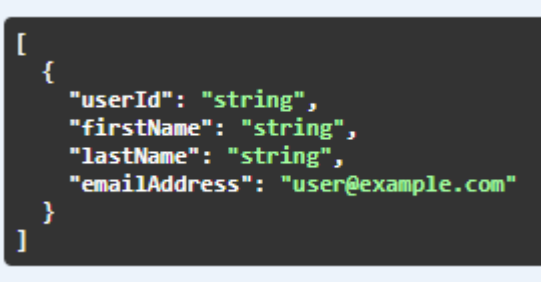
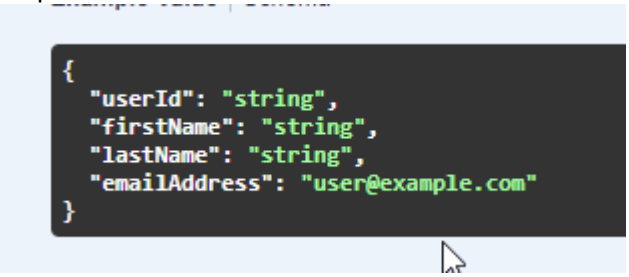
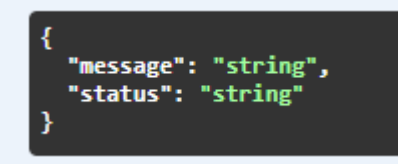
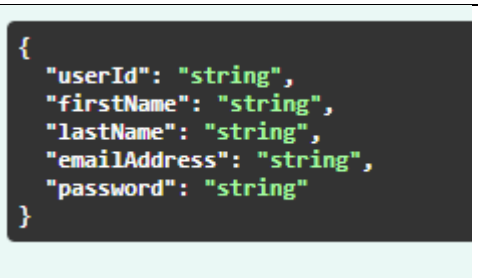
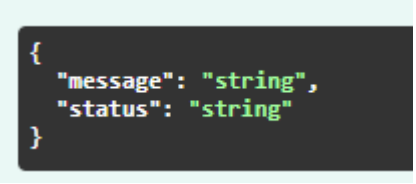
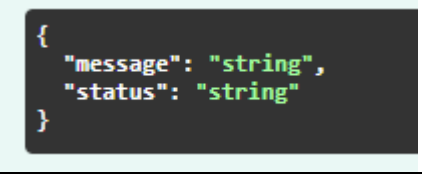
Node Express User CRUD API

This assignment is an introduction to node and express and it will serve as the beginning back end for the next assignments it is imperative that this assignment be done and done correctly so that the next assignment after this can rely on it.

Write a Node application which will listen for request (web server) on port 3000 and it should provide the following routes and functionality for each route respectively.

Method	Route	Functionality
[GET]	/	Any Get Request made to the root path of the server should return a HELP Page written in HTML and CSS which will outline the functionality of the API. It should list each method that is supported and provide an example of the input and output expected as well as any error paths or error handling that is available. An example of what this could look like is available at https://unf.josecgomez.dev/ Though it can be as simple as a table like this.
[POST]	/Users	When a POST request is made to /Users it should expect a user object as shown below  It should take this data and create a new User record and store it in an in-memory array. All the fields are required and there should be error checking in place for this. This should return a 406 UnAcceptable status along with a message  No two users can share the same user ID if the same user ID is attempted to be created it should respond with 409 Conflict status and a

		message  If everything is successful it should respond with status 201 (Created) and a copy of the object (without the password) 
[GET]	/Users	Should return a list (JSON ARRAY) of all the Users which have been created with a 200 response or just a blank array. 
[GET]	/Users/{userId}	If the user ID provided exists it should return that user object with a 200 response  If the user ID provided is not found it should return a 404 (not Found) error along with a message 
[PATCH]	/Users/{userId}	This method should expect a payload of a user record (incomplete or complete) and should update the existing user with the provided values if the user exists.

		 If the user doesn't exist it should return 404 not found along with a message  It should ignore any changes to the user ID but all other fields provided should be updated and a status 200 should be returned along with the updated user object.
[DELETE]	/Users/{userId}	This request should take the provided user id and remove it from the user array (if found) and respond with status 204 Deleted. If the user is not found it should respond with a 404 not found and appropriate message. 

Make sure the user info is stored in a User Class (OOP!) and an in-memory array.

Any requests to other Routes not outlined above should redirect to the [GET] / Route which provides the Help

The assignment must be written using TypeScript, Node and Express no other external tools or libraries are necessary

As extra credit provide an Input <HTML> form that can perform all of the above functions. (Partial credit given for each piece)
Create User
Update User
Delete User

Get User Info