Assignment 5

Submit Assignment

Due Mar 22 by 11:55pm **Points** 100 **Submitting** a file upload **File Types** zip

CS-554 Assignment 5

Inter-Process Communication

For this lab, you will make an API similar to Assignment 4. This time you will implement GET, PUT, POST, and DELETE routes with data retrieved from a second process. You will also continue to practice your usage of redis and async / await usage.

You will promisify the redis-client for use, and you will use all of its methods with the use of the *await*keyword.

For this lab, you will use a worker to do **all** database-like operations (Create, Read, Update, and Delete). Normally, you would only pass writes off to the worker.

You will not have to utilize a cache for this assignment

You will need to run two programs!

Running 2 Processes with your npm start command

For your `npm start` command, simply do something like:

```
"start": "node worker/index.js && node server/index.js
```

In order to run both at once.

For the sake of this assignment, please make only one package that has dependencies for both the worker and server.

Your worker

You will make a file, **worker.js**, that performs the following:

1. When the worker starts up, it will **download** the dummy data (https://gist.githubusercontent.com/philbarresi/5cf15393d245b38a2d86ce8207d5076c/raw/d529fb474c1af347702ca4d7b992256237fa2819/lab5.json) and store it in memory, similar to last week. **NOTE**: you must **download** this data, not hardcode it.
2. The worker should respond to a request to provide the person listed at the supplied ID
3. The worker should respond to a request to create a person with the supplied information; the response should have the newly created person data
4. The worker should respond to a request to update a person with the supplied information; the response should have the

updated person data

5. The worker should respond to a request to delete a person with the supplied ID; the response should simply indicate that the deletion was successful.

Your Routes

You will only have 4 routes!

`GET /api/people/:id`

This route will publish a message to request a person from the worker, and render JSON of the person (or of an error, should once occur)

`POST /api/people`

This route will publish a message to request that the worker creates a person, and render JSON of the person created (or of an error, should once occur)

`DELETE /api/people/:id`

This route will publish a message to request that the worker deletes a person, and render JSON stating that the operation completed (or of an error, should once occur)

`PUT /api/people/:id`

This route will publish a message to request that the worker updates a person, and render JSON of the person updated (or of an error, should once occur)

General Requirements

1. Remember to submit your `package.json` file but **not** your `node_modules` folder.
2. Remember to check for errors, everywhere!