TESTING LOG

- Setting Up Stage - arguments
  - `./httpserver -f kvs -m alias  localhost:8000` OK
  - `./httpserver -f kvs -m        localhost:8080` FAILED
  - `./httpserver -f kvs           localhost:8080` FAILED
  - `./httpserver -f               localhost:8080` FAILED
  - `./httpserver        -m alias  localhost:8080` FAILED
  - `./httpserver        -m        localhost:8080` FAILED
  - `./httpserver                  localhost:8080` FAILED
- Alias detection
  - **Standard**
    `./httpclient a:12345…67890:xxx` SUCCEED – 200 OK
    `./httpclient a:xxx          :yyy` SUCCEED – 200 OK
    `./httpclient s:file         :yyy` SUCCEED – 200 OK
    `./httpclient r:yyy          :fileback` SUCCEED – 200 OK

    `./httpclient s:file         :12345…67891` SUCCEED – 200 OK
    `./httpclient r:12345…67890:fileback` SUCCEED – 200 OK
  - **Lack of arguments**
    `./httpclient a:12345…67890:` FAILED – 400 Bad Request
    `./httpclient a:            :xxx` FAILED – 400 Bad Request
    `./httpclient a:            :` FAILED – 400 Bad Request
    `./httpclient a:` FAILED – 400 Bad Request
  - **existing_name not exists**
    `./httpclient a:xxx          :yyy` FAILED – 404 Not Found
  - **alias chain loop**
    `./httpclient a:12345…67890:xxx` SUCCEED – 200 OK
    `./httpclient a:xxx          :yyy` SUCCEED – 200 OK
    `./httpclient a:yyy          :xxx` SUCCEED – 200 OK
    `./httpclient s:file         :xxx` FAILED – 404 Not Found
    `./httpclient r:xxx          :fileback` FAILED – 404 Not Found
  - **Restart server**
    `./httpclient a:12345…67890:xxx` SUCCEED – 200 OK
    `RESTART SERVER`
    `./httpclient s:file         :xxx` SUCCEED – 200 OK
    `RESTART SERVER`
    `./httpclient r:xxx          :fileback` SUCCEED – 200 OK
- File Transportation - file
  - 4 KB file   SUCCEED – 200 OK
  - 4 MB file   SUCCEED – 200 OK

QUESTION: Explain the difference between fully resolving a name (to an httpname) when the name is created and the approach that you're taking for this assignment. Give an example of when it might be useful.

First of all, we can have multiple names point to one file on server which allows clients to obtain the same file in different name. This way enhances the performance of the client to create, modify or obtain the file on server.

On the other hand, httpname is a standard 40 HEX doc name to use on server. However, for actual clients in the real may not want to use such long name which has risks to type wrong or other issues. The alias method provides client a new way to better pick file name even though on the server the file name never changed. It enhances the user experience.

QUESTION: Was it easier to modify your existing KV store code from Assignment 3 as compared to getting it to work the first time?

Absolutely right. Because this time the alias KVS file doesn't have data blocks that are separated with entry blocks, the alias KVS is much easier to implement. Especially after fully understanding how the cache and KVS cooperate with each other, alias KVS only takes me a couple of hours to finish.

QUESTION: What did you learn about system design from this class? In particular, describe how each of the basic techniques (abstraction, layering, hierarchy, and modularity) helped you by simplifying your design, making it more efficient, or making it easier to design.

The most important principle that I learned is modularity. Try to divide a piece of long code into different modules with different functions. It shorts the program length and keeps the program much simpler to understand. When design the project, I started to think of how to design modules instead of long piece of code to make every line of code efficient. In addition, those modular can be changed and improved later for new functionalities.