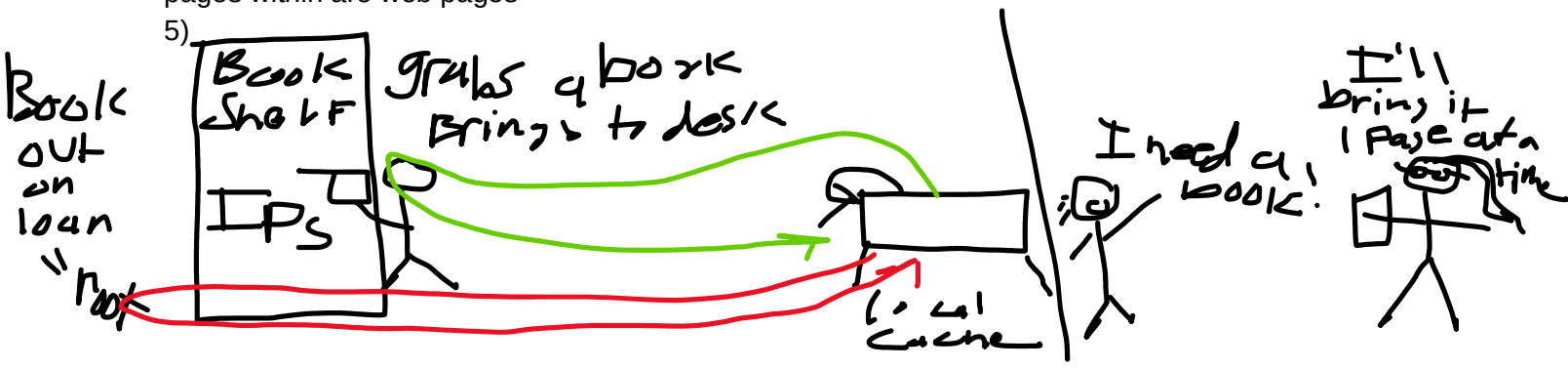
Topic 1: The Internet and the World Wide Web

1) A really big network (set of computers that can intercommunicate)
2) A connected system of public webpages accessible through the internet
3) a) computers linked either physically or wirelessly
   b) computers that store large amounts of data that can be accessed by user
   c) specialized computer meant to connect other computers together (limit amount of connections needed
   d) packets are tiny groups of information. Request usually have a large number and if they are corrupted or lost, they can easily be resent or fixed
4) The internet is a library and each book is a computer. Some books are servers. And the pages within are web pages
5)



Topic 2: IP Addresses and Domains

1) IP address is the actual unique name to send a request to a server. Domain name is the nickname that humans use to refer to it.

2) 104.22.13.35

3) For safety and better control requests
4) checks the local cache, router cache, ips cache, and root in that order to connect it to the correct IP address

Topic 3: How a web page loads into a browser

Steps Scrambled	Steps in Correct Order	Why did you put this step in this position?
Example: Here is an example step	*Here is an example step*	*- I put this step first because ____* *- I put this step before/after ____ because ____*
Request reaches app server	Initial request (link clicked, URL visited)	Need a request before data will be sent.
HTML processing finishes	Request reaches app server	Request has to be given to server before it knows to send it back.

App code finishes execution	Browser receives HTML, begins processing	HTML will only process after it has been received after server get it's request.
Initial request (link clicked, URL visited)	App code finishes execution	App code is in the HTML so html has to start processing, but can't have finished.
Page rendered in browser	HTML processing finishes	Can't display in browser until html is completely read
Browser receives HTML, begins processing	Page rendered in browser	Nothing left, but to show all data processed.

Topic 4: Requests and Responses

Part A: GET /

1) h1 tag jurrni, h2 tag journaling your journies
2) text content html
3) yes, h1 tag and h2 tag looked like headers
4) yes, it only has text an html

Part B: GET /entries

1) We will see the objects from the entries variable
2) An array of objects
3) yes, it refers the entries variable
4) application/json,

Part C: POST /entry

1) requesting and then pushing a new object to the entry array and incrementing global id.
2) date and content. Strings
3) {"date" : "July 19th", "content" : "Yippee"}
4) localhost:4500/entry
5) our changes with the new object.
6) Application/json
7) yes, we changed it in hopes it would change to what we had.
8) yes. It uses json, but I don't understand exactly why.

Further study

1.Curl httpbin.org

2.Curl httpbin.org/anything

3.curl -i -X POST httpbin.org/anything

4.curl --data '{"value":"panda"}' httpbin.org/anything

5.curl google.com/robots.txt

6.curl -H "User-Agent: elephant" httpbin.org/anything

7.curl -X "DELETE" httpbin.org/anything

8.curl -i httpbin.org/anything

9.curl -X POST -H "Content-type: application/json" -d '{"value": "panda"}'

https://httpbin.org/anything

10.curl -X POST -H "Content-type: application/json" https://httpbin.org/anything

11.curl -H "Accept-Encoding: gzip" httpbin.org/anything