ps4_Leemingsawat_Kongpob

Kongpob Lee

2023-02-03

Part I: Concepts & Definitions

Question 1: What hidden directory is created whenever a git repository is created?

Answer:

.git/ directory (contains all information needed for git to work)

Question 2: Describe what git objects are, what they are identified by, and where they are stored.

Answer:

git objects are stored inside the .git/objects directory - so basically git repository = collection of objects and each objects are identified with their own hash (40 digits identification ID). Moreover, git objects can be categorized into 4 categories, which are blob, tree, commit, and tag.

Question 3: List the 4 types of git objects and define each of them.

Answer:

- blob = used to store file data – generally a file, like text
- tree: = same as directory – it references a bunch of other trees and blobs (same as files and sub-directories)
- commit = objects hold metadata for each change introduce in the repository (including: author, date, and commit)
- tag = object assigns as arbitrary human-readable name to a specify object usually a commit – typically references to a particular commit

---

Part II: Exploring git objects

Question 2

Answer: git init

Question 3

Answer 1 (Command to redirect text into 'problemset4.R'): echo "Kongpob Leemingsawat" > problemset4.R

Answer 2 (Command to print contents of 'problemset4.R'): git diff

Question 4

Answer 1 (Command to add 'problemset4.R'): git add problemset4.R

Answer 2 (Command to find hash ID for 'problemset4.R'): git hash-object problemset4.R

Output: da4217c2131b64adaaf9a2fd509456b38cf8eee4

Question 5

Answer 1 (Command to find content of the blob object): git cat-file -p da4217c

Output: Kongpob Leemingsawat

Answer2 (Command to find type of the blob object): git cat-file -t da4217c

Output: blob

Answer 3 (Command to find size of the blob object): git cat-file -s da4217c

Output: 21

Question 6

Answer 1: git commit -m "add problemset4.R"

Answer 2: git log

Answer 3: 493873390990ea658a477fedbffc65746dca5c4e

Question 7

Answer 1: git cat-file -p $(git rev-parse HEAD) - [git cat-file -p 493873390990ea658a477fedbffc65746dca5c4e also works]

Output:

tree 58effdd7ccd17456d9a0df3e23467e3eafa58d72 author KongpobLeemingsawat kleemingsawat713@gmail.com 1675470467 -0800 committer KongpobLeemingsawat kleemingsawat713@gmail.com 1675470467 -0800

add problemset4.R

Answer 2: git cat-file -t 493873390990ea658a477fedbffc65746dca5c4e

Output: commit

Answer 3: git cat-file -s 493873390990ea658a477fedbffc65746dca5c4e

Output: 214

END
