**Senpiper Tasks**

1. Find the count of log statements in the attachements (as access.log.zip)file “access.log” with successful response (status code 200).

Solution: grep " 200 " access.log | wc –l

The number of log statements with a 200 status code is 18

* Grep “200” filters lines that contain “200”
* wc –l counts the number of matching lines.

1. Write a command to find the files which contain words DEBUG, ERROR, and INFO in any directory of the filesystem.

Solution: grep -rilE "DEBUG|ERROR|INFO" /path/to/search/

* R - recursive
* I – ignore case
* L-list files

1. What would be the sed command to convert the string input “AbICdZEt3Gh4s...." to “a-bc-de-fg-hi-j…”?

Solution: echo "AbICdZEt3Gh4s...." | sed 's/[A-Z]/\L&/g; s/[0-9]/-/g; s/t//g'

* s/[A-Z]/\L&/g → Converts uppercase letters to lowercase.
* s/[0-9]/-/g → Replaces digits with -.
* s/t//g → Removes t from the string.

1. Write a script/program to return if the opening and closing brace are complete, otherwise false.

Solution: Python Code:

def are\_braces\_balanced(s):

stack = []

braces = {'{': '}'}

for char in s:

if char in braces:

stack.append(char)

elif char == '}':

if not stack or braces[stack.pop()] != char:

return False

return len(stack) == 0

* Uses a stack to check if braces are correctly paired.

1. Write steps to create and publish a docker image to the docker repository.

Solution:

1. Create a Dockerfile:

FROM nginx:latest

COPY index.html /usr/share/nginx/html/index.html

EXPOSE 80

CMD ["nginx", "-g", "daemon off;"]

2. Build and Run the Image:

docker build -t my-custom-nginx .

docker run -d -p 8080:80 my-custom-nginx

3. Push to Docker Hub:

docker login

docker tag my-custom-nginx your-dockerhub-username/my-custom-nginx

docker push your-dockerhub-username/my-custom-nginx

* FROM nginx:latest → Uses the official nginx image.
* COPY index.html ... → Adds a custom web page.
* docker build → Creates the image.
* docker run → Starts the container.

1. Given the following snippet of nginx server configuration and write a location block request “/api” that will include the proxied server: <http://localhost:8000/welcome/home>.

Solution: location /a {

proxy\_pass http://localhost:8000/welcome/home;

proxy\_set\_header Host $host;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

}

* proxy\_pass → Forwards requests to another service.
* proxy\_set\_header → Preserves request headers.