

Name:- Anmol Sharma

Roll NO:- 2024101148

Q1)

a) If we push without pulling our teammate's changes, bit will reject our push due to diverging histories. So for resolving this we can use `git pull--rebase` after this we can push again `git push`.

b) `git add`.

c) `git checkout testing`

d) ~~`git reset -- Soft HEAD ~ 1`~~

d) `git reset -- Soft HEAD ~ 1`

82)

a) `head -n 5 data.txt`

b) `sed 's/HTML/CSS/g' input.txt`

c) `sed -n '5,10p' report.txt`

d)

```
# !/bin/bash
```

```
if [ -f "$1" ]; then
```

```
    wc -l < "$1"
```

```
else
```

```
    echo "File not found!"
```

```
fi
```

83)

a)

```
SELECT name, salary
FROM employees
WHERE department = 'Engineering' AND salary > 50000;
```

b)

```
SELECT department, AVG(salary) AS avg_salary
FROM employees
GROUP BY department
ORDER BY avg_salary DESC;
```

c)

```
db.employees.updateOne(
```

```
  { name: "John Doe" },
```

```
  { $inc: { salary: 10000 } }
```

```
);
```


d) db.orders.aggregate([

{ \$match: { status: "delivered" } },

{ \$group: { _id: null, totalRevenue: { \$sum: "\$total-price" } } }

]);

8.4)
a) - 'a' - Append mode: adds data to the end of the file without erasing content.

'x' - Exclusive creation: creates a new file and raises an error if the file already exists.

'b' - Binary mode: - Opens the file in binary format for reading/writing bytes.

't' - Text mode: - Opens the file in text format (default mode for string data).

b) Using readlines() is inefficient for large files because it loads the entire file into memory at once, which can cause high memory usage or crashes if the file is very large.

More efficient method.

with open("large-file.txt") as file:

for line in file:

process(line)

Q5)

Ans a)

A.ndim = 2

A.Size = $4 \times 5 = 20$

b)

$A[::-1, ::2]$

$[::-1]$ reverse the rows (flips the matrix vertically)

$::2$ selects every second column (0th, 2nd, 4th columns)

```
array([[0.14, 0.13, 0.71],  
       [0.91, 0.64, 0.55],  
       [0.43, 0.67, 0.24],  
       [0.25, 0.98, 0.72]])
```

c) $fig, axes = plt.subplots(2, 2)$

$axes[0,0].plot([1, 2, 3], [4, 4, 9])$

$axes[0,0].set_title("line plot")$

$axes[0,1].plot_scatter([1, 2, 3], [3, 5, 2])$

$axes[0,1].set_title("Scatter plot")$

$axes[1,0].bar(['A', 'B', 'C'], [5, 2, 7])$

$axes[1,0].set_title("Bar plot")$

$axes[1,1].hist([1, 2, 2, 3, 3, 3, 4])$

$axes[1,1].set_title("Histogram")$

$plt.tight_layout()$

$plt.show()$

Q6)

a) "post"
min = "18"

b)

c) center;
100vw;

Q7)

a) let - block-scoped, reassignable

e.g.: let count = 5;

count = 10; // Valid

const :- block-scoped, not reassignable

e.g.: const PI = 3.14;

PI = 3.14159; // Error: Assignment to constant variable

var :- function-scoped, reassignable

e.g.: var name = "Alice";

name = "Bob"; // Valid

b) addEventListener

~~getElementById~~ ~~document~~ ~~Box~~

style.backgroundColor