

1. What's the difference between ' ' and " " in shell?

- Single quotes ' ': Preserve the literal value of everything inside, things as variables, special characters, and command substitutions are not expanded.
- Double quotes " ": Allow variable expansion and command substitution, so special characters like \$, \, and backticks are interpreted.

2. Explain [-f filename] vs [-d dirname].

- In shell, [...] is a test command, so [-f filename] :Returns true if the file exists and is a regular file. While [-d dirname], Returns true if the file exists and is a directory.

3. Explain stdout/stderr redirection, appending vs overwrite. How can you confirm redirection using file descriptors?

- File descriptors (FDs):
 - 0 = stdin (input)
 - 1 = stdout (normal output)
 - 2 = stderr (error messages)
- Redirection
 - > : redirect stdout, overwrite file.
 - >> : redirect stdout, append to file.
- 2> : redirect stderr, overwrite file.
- 2>> : redirect stderr, append to file.
- &> : redirect both stdout and stderr.
- Examples

ls > out.txt # stdout to out.txt (overwrite)

ls >> out.txt # stdout to out.txt (append)

ls notafile 2> err.txt # stderr to err.txt

ls notafile &> both.txt # stdout + stderr to both.txt

- Confirming redirection

Check with /proc/self/fd:

```
exec 1>out.txt # redirect stdout
```

```
ls -l /proc/$$/fd
```

This will show file descriptor 1 (stdout) pointing to out.txt instead of the terminal.

4. Show an example of a for loop in bash. Then, write a simple bash calculator that does add/subtract.

- For loop example

```
“for i in 1 2 3 4 5
```

```
do
```

```
    echo "Number: $i"
```

```
done”
```

Output: prints 1 to 5.

- Simple Bash Calculator

```
“#!/bin/bash
```

```
echo "Enter first number:"
```

```
read a
```

```
echo "Enter second number:"
```

```
read b
```

```
echo "Choose operation (+ or -):"
```

```
read op
```

```
if [ "$op" == "+" ]; then
```

```
    echo "Result: $((a + b))"
```

```
elif [ "$op" == "-" ]; then
```

```
    echo "Result: $((a - b))"
```

```
else
```

```
    echo "Invalid operation"
```

```
fi”
```