# Chapter 6

## ? Short Questions with Answers

### Q1: Who developed the Bourne Shell in Unix/Linux?

- Developed by **Stephen Bourne** at AT&T Bell Labs.
- Introduced in UNIX Version 7 (1977).
- Example: Known as the **sh shell**.

### Q2: What is the file name of the Bourne Shell executable in Unix/Linux?

- The file name is **/bin/sh**.
- It is the location of the Bourne Shell binary.
- Example: Running `/bin/sh` starts the shell.

### Q3: What is the default prompt of the Bourne Shell in Unix/Linux?

- Default prompt is **$** for normal users.
- For root user, it is **#**.
- Example: `$ echo hello` shows shell prompt with command.

### Q4: What is a Unix/Linux shell?

- A **command-line interpreter** between user and OS.
- Executes commands and scripts.
- Example: `ls` → shell interprets and displays files.

### Q5: What is the difference between a shell and a terminal?

- **Terminal:** Program that opens text interface.
- **Shell:** Interpreter running inside terminal.
- Example: Terminal is the "window," shell is the "command engine."

### Q6: What is the primary function of a shell?

- To read, interpret, and execute user commands.
- Provides environment for running scripts.
- Example: Shell runs `cp file1 file2` to copy files.

### Q7: What is a shell script?

- A text file with multiple shell commands.
- Automates tasks by executing sequentially.
- Example: `script.sh` containing `echo "Hello"`

### Q8: How do shells manage files and directories?

- Provide commands like `ls`, `cd`, `cp`, `mv`.
- Allow creating, removing, and modifying files.
- Example: `mkdir newdir` creates directory.

### Q9: What is input/output redirection in shells?

- Redirects standard input/output to files.
- `>` writes output to file, `<` takes input from file.
- Example: `ls > files.txt` saves file list to `files.txt`.

### Q10: What is job control in shells?

- Allows managing background/foreground processes.
- Commands: `jobs`, `fg`, `bg`.
- Example: Run `sleep 30 &` → job goes to background.

### Q11: How can you execute a Bourne Shell script in Unix/Linux?

- Make script executable: `chmod +x script.sh`.
- Run script with `./script.sh` or `sh script.sh`.
- Example: `./backup.sh` runs backup script.

### Q12: What is the difference between single quotes and double quotes in the Bourne Shell?

- **Single quotes (‘ ’):** Prevent variable expansion.
- **Double quotes (“ ”):** Allow variable expansion.
- Example: `echo '$USER'` prints `$USER`, `echo "$USER"` prints username.

### Q13: How can shells be used for networking tasks?

- Provide commands like `ping`, `ftp`, `ssh`, `scp`.
- Used for remote login, file transfer, connectivity checks.
- Example: `ping google.com` checks internet connection.

### Q14: What is a shell script?

- (Same as Q7, repeated in list).
- Script with shell commands executed in sequence.
- Example: `for i in 1 2 3; do echo $i; done`

### Q15: What is the purpose of the "case" statement in the Bourne Shell?

- Provides multi-way decision making.
- Similar to `switch` in C.
- Example:

```
case $day in
  Mon) echo "Start of week";;
  Fri) echo "Weekend soon";;
esac
```

## Long Questions with Answers

### Q1: Explain the functions of shell.

- **Command Interpretation:** Reads and executes commands.
- **Program Execution:** Runs scripts and applications.
- **I/O Redirection:** Redirects input/output to files or devices.
- **Job Control:** Manages background and foreground tasks.
- **Environment Control:** Manages variables like `PATH`.
  - *Example: Running `ls > files.txt` saves output to a file.*

### Q2: What are the different types of quotes in Bourne Shell scripts in Unix/Linux, and how do they affect variable expansion?

- **Single Quotes (‘ ’):** Prevent variable and command expansion.
  - Example: `echo '$HOME'` → prints `$HOME`.
- **Double Quotes (“ ”):** Expand variables and commands.
  - Example: `echo "$HOME"` → prints `/home/user`.
- **Back Quotes (``):** Execute commands inside quotes.
  - Example: `echo `date`` → prints current date.

### Q3: What is the Bourne Shell in Unix/Linux, and what are its features?

- Developed by **Stephen Bourne** at Bell Labs.
- Standard shell in early UNIX (file: `/bin/sh`).

- **Features:**
  - Command execution.
  - Input/output redirection.
  - Variables and control structures (if, case, loops).
  - Script automation.
    - *Example: Writing a script to backup files using* `sh`.

## Q4: What are the steps involved in writing a Bourne Shell script in Unix/Linux, and how can you execute it?

1. **Create a file** with `.sh` extension.
2. **Write script** inside file starting with `#!/bin/sh`.
3. **Make file executable**: `chmod +x script.sh`.
4. **Run script**: `./script.sh` or `sh script.sh`.
   - *Example: `script.sh` →

```
#!/bin/sh
echo "Hello World"
```

Execution: `./script.sh` prints *Hello World*.