```
Shell Script Cheat Sheet

 BASICS
#!/bin/bash        # Shebang (always at the top)
# Comment line
echo "Hello, world" # Print

 VARIABLES
NAME="John"
echo "Hi, $NAME"
NUM1=10
NUM2=5
SUM=$((NUM1 + NUM2))

 FILE & DIRECTORY
ls         # List files
pwd        # Current directory
cd /path  # Change directory
mkdir new_dir      # Make directory
rm -rf file.txt    # Delete file/dir
touch file.txt     # Create empty file

 CONDITIONS
if [ $NUM1 -gt $NUM2 ]; then
  echo "NUM1 is greater"
else
  echo "NUM2 is greater"
fi

# Strings
if [ "$NAME" = "John" ]; then echo "Hi John"; fi

Operators:
- -eq (equal), -ne (not equal)
- -gt (greater), -lt (less), -ge, -le
- -z (empty), -n (not empty)

 LOOPS
# For loop
for i in {1..5}; do
  echo "Loop $i"
done

# While loop
count=1
while [ $count -le 5 ]; do
  echo $count
  ((count++))
done

 FUNCTIONS
say_hello() {
  echo "Hello $1"
}
```

```bash
say_hello "Alice"
```

INPUT & ARGUMENTS
```bash
read -p "Enter name: " USERNAME
echo "Hello $USERNAME"

# Command line arguments
echo "First argument: $1"
echo "All args: $@"
```

FILE TESTS
```bash
FILE="file.txt"
if [ -f "$FILE" ]; then
  echo "It's a file"
fi
if [ -d "$FILE" ]; then
  echo "It's a directory"
fi
```

COMMON COMMANDS
```bash
date                 # Current date/time
whoami               # Current user
ps aux               # List processes
top / htop           # System monitor
uptime               # Load average
df -h                # Disk usage
free -h              # Memory
```

SCRIPT OPTIONS
```bash
#!/bin/bash
while getopts ":u:p:" opt; do
  case $opt in
    u) USER="$OPTARG" ;;
    p) PASS="$OPTARG" ;;
    \?) echo "Invalid option: -$OPTARG" ;;
  esac
done
```

ERROR HANDLING
```bash
set -e  # Exit on any error
command_that_might_fail || {
  echo "Command failed!"
  exit 1
}
```

BONUS TRICKS
```bash
# Redirect output
command > file.txt        # stdout
command 2> error.txt      # stderr
command >> file.txt       # append
# Combined
command > all.txt 2>&1
# Background process
long_running_command &
```