

BASH THAM KHẢO NHANH

Lệnh, scripting, pipe, chuyển hướng, quản lý tiến trình

Cơ Bản

echo & Điều Hướng

```
echo "Hello, World!" # print text
pwd # print working directory
cd /path/to/dir # change directory
cd .. # go up one level
cd - # go to home directory
cd - # go to previous directory
```

Liệt Kê & Tạo

```
ls # list files
ls -la # long format, show hidden
ls -lh # human-readable sizes
mkdir mydir # create directory
mkdir -p a/b/c # create nested directories
```

Sao Chép, Di Chuyển & Xóa

```
cp file.txt copy.txt # copy file
cp -r dir/ backup/ # copy directory recursively
mv old.txt new.txt # rename / move
rm file.txt # delete file
rm -r dir/ # delete directory recursively
rm -rf dir/ # force delete (no prompt)
```

Biến & Mở Rộng

Biến

```
name="Alice" # assign (no spaces!)
echo "$name" # variable expansion
echo "${name}_file" # braces for clarity
readonly PI=3.14 # constant
unset name # delete variable
```

Biến Đặc Biệt

```
$_ Tên script
$1 $2 ... Tham số vị trí
 $# Số lượng tham số
 @ Tất cả tham số (từng từ riêng biệt)
 * Tất cả tham số (một chuỗi duy nhất)
 $* Trạng thái thoát của lệnh trước
 $$ ID tiến trình hiện tại
 $! PID của tiến trình nền cuối cùng
```

Thay Thế Lệnh & Số Học

```
files=$(ls) # capture output
today=$(date +%Y-%m-%d) # command substitution
count=$((5 + 3)) # arithmetic: 8
echo $(10 / 3) # integer division: 3
echo $(10 % 3) # modulo: 1
```

Thao Tác Chuỗi

```
${#str} Độ dài chuỗi
${str:0:5} Chuỗi con (offset:length)
${str/old/new} Thay thế kết quả đầu tiên
${str/old/new} Thay thế tất cả kết quả
${str^^} Chữ hoa
${str,,} Chữ thường
```

Điều Kiện

if / elif / else

```
if [[ "$name" == "Alice" ]]; then
  echo "Hi Alice"
elif [[ "$name" == "Bob" ]]; then
  echo "Hi Bob"
else
  echo "Who are you?"
fi
```

Toán Tử Kiểm Tra

```
-eq -ne Số nguyên bằng / không bằng
-lt -gt Số nguyên nhỏ hơn / lớn hơn
-le -ge Số nguyên nhỏ hơn/lớn hơn hoặc bằng
== != Chuỗi bằng / không bằng
-z "$str" Chuỗi rỗng
-n "$str" Chuỗi không rỗng
-f file File tồn tại và là file thường
-d dir Thư mục tồn tại
-e path Đường dẫn tồn tại (bất kỳ loại)
-r -w -x Đọc được / ghi được / thực thi được
&& || AND / OR logic
```

Vòng Lặp

Vòng Lặp for

```
for fruit in apple banana cherry; do
  echo "$fruit"
done
```

```
for f in *.txt; do
  echo "File: $f"
done
```

Vòng Lặp for Kiểu C

```
for ((i=0; i<5; i++)); do
  echo "$i"
done
```

Vòng Lặp while

```
count=0
while [[ $count -lt 5 ]]; do
  echo "$count"
  ((count++))
done
```

Điều Khiển Vòng Lặp

```
break Thoát khỏi vòng lặp
continue Bỏ qua sang lần lặp tiếp theo
```

Hàm

Định Nghĩa & Gọi

```
greet() {
  echo "Hello, $1!" # $1 = first arg
  return 0 # exit status
}
greet "Alice" # Hello, Alice!
```

Biến Cục Bộ & Giá Trị Trả Về

```
add() {
  local sum=$(( $1 + $2 ))
  echo "$sum" # "return" via stdout
}
result=$(add 3 5) # capture: 8
```

Pipe & Chuyển Hướng

Pipe

```
ls -l | grep ".txt" # pipe output
cat log | sort | uniq # chain commands
cmd1 | tee out.txt # pipe + save to file
```

Chuyển Hướng

```
cmd > file Chuyển hướng stdout (ghi đè)
cmd >> file Chuyển hướng stdout (nối thêm)
cmd < file Chuyển hướng stdin từ file
cmd 2> file Chuyển hướng stderr
cmd 2>&1 Chuyển hướng stderr vào stdout
cmd &> file Chuyển hướng stdout + stderr
cmd << EOF Here document (nhập liệu inline)
/dev/null Bỏ qua output: 'cmd > /dev/null'
```

Thao Tác File

Xem File

```
cat file.txt # print entire file
head -n 10 file.txt # first 10 lines
tail -n 10 file.txt # last 10 lines
tail -f log.txt # follow (live updates)
less file.txt # paginated viewer
```

Đếm & Tìm Kiếm

```
wc -l file.txt # count lines
wc -w file.txt # count words
wc -c file.txt # count bytes
find . -name "*.txt" # find by name
find . -type d # find directories
find . -mtime -7 # modified in last 7 days
```

Lệnh File Khác

```
touch file Tạo file / cập nhật timestamp
stat file Metadata file (kích thước, ngày)
file img.png Nhận diện loại file
diff a.txt b.txt So sánh hai file
sort file.txt Sắp xếp các dòng
uniq Xóa các dòng trùng lặp liên kề
cut -d: -f1 Trích xuất trường theo dấu phân tách
tr 'a-z' 'A-Z' Chuyển đổi / thay thế ký tự
```

Xử Lý Văn Bản

grep

```
grep "error" log.txt # search for pattern
grep -i "error" log.txt # case-insensitive
grep -r "TODO" src/ # recursive search
grep -n "func" file.go # show line numbers
grep -c "error" log.txt # count matches
grep -v "debug" log.txt # invert match
```

sed

```
sed 's/old/new/' file # replace first per line
sed 's/old/new/g' file # replace all
sed -i 's/old/new/g' file # edit in place
sed -n '5,10p' file # print lines 5-10
sed '/pattern/d' file # delete matching lines
```

awk

```
awk '{print $1}' file # print first field
awk -F: '{print $1}' file # custom delimiter
awk 's3 > 100' file # filter by field value
awk '{sum+=$1} END{print sum}' file # sum column
```

Quyền Hạn

chmod

```
chmod 755 script.sh # rwxr-xr-x
chmod +x script.sh # add execute
chmod -w file.txt # remove write
chmod u+x,g-w file # user +exec, group -write
```

Tham Khảo Quyền Hạn

```
r (4) Đọc
w (2) Ghi
x (1) Thực thi
u / g / o Người dùng / Nhóm / Khác
755 Chủ sở hữu: rw, Nhóm/Khác: r-x
644 Chủ sở hữu: rw, Nhóm/Khác: r--
```

Quyền Sở Hữu

```
chown user file.txt # change owner
chown user:group file.txt # change owner + group
chown -R user:group dir/ # recursive
```