

Bài 10: Lập trình Shell - Giới thiệu về biến

Nguyễn Minh Hải

Ngày 11 tháng 10 năm 2016

Nội dung

Giới thiệu (1)

- *shell script* là tệp văn bản ghi lại chuỗi lệnh mà ta muốn *shell* thực hiện
- Lập trình *shell* là viết *shell script*
- Cấu trúc một mỗi *shell script* gồm có
 - **shell keywords** các từ khóa như: *if*, *else*, ...
 - **Shell commands** Các lệnh trong shell như *pwd*, *echo*, ...
 - **Linux binary commands** Các lệnh nhị phân của Linux như *who*, *w*
 - **Text processing utilities** Các hoạt động xử lý văn bản như *grep*, *cut*
 - **Functions** Các công việc thường làm được tổ chức thành các hàm
 - **Control flows** Các cấu trúc điều khiển

Giới thiệu (2)

- Mục đích
 - Tiết kiệm công sức cho những công việc thường xuyên phải tiến hành
 - Tổ chức được những công việc phức tạp một cách có cấu trúc và rành mạch
 - Cách làm việc có tính kế thừa và mở rộng
- Một số ví dụ
 - Giám sát hệ thống tự động
 - Sao lưu dữ liệu tự động
 - Lập lịch tự động cho một số tác vụ

Giới thiệu về Bash

- Phát triển bởi Brian J. Fox vào năm 1989 cho dự án GNU nhằm thay thế Bourne Shell
- Là shell mặc định của Linux
- Tương thích với sh, có dùng lại một số tiện ích của Korn shell và C shell
- Có thể chạy trên mọi phiên bản Unix và một số hệ điều hành khác như MS-DOS, OS/2, Windows, ...

Ví dụ Hello Word

- Dùng một trình soạn thảo như vi hoặc emacs tạo một tệp hello.sh như sau

```
#!/bin/bash
echo "Hello World!"
```
- Cấp quyền thực thi cho người dùng hiện tại đối với tệp tin trên
- Chạy tệp tin với lệnh

\$./hello.sh
- Dòng đầu được gọi là Shebang, chỉ dẫn tới thư mục chứa shell thực thi chương trình này
- Dòng thứ hai là một lệnh in ra màn hình

Biến (1)

- Biến trong shell dùng để lưu dữ liệu và các thông tin cài đặt (gọi chung là thông tin)
- Khác với các ngôn ngữ lập trình khác, trong bash, biến không nhất thiết khai báo kiểu
- Biến gồm có biến môi trường và biến người dùng trong chương trình
- Biến môi trường có tên gồm toàn chữ cái in hoa gồm hai loại
 - Biến môi trường *enviromental variable* dùng để lưu các thông tin sẽ được truyền cho các shell con và các tiến trình được khởi tạo trong shell hiện tại
 - Biến shell *shell variable* dùng để lưu thông tin dùng trong shell hiện tại

Để chuyển biến shell thành biến môi trường dùng lệnh **export**, chuyển ngược lại dùng **export -n**

Biến (2)

- Cách khởi tạo đồng thời gán giá trị cho biến

```
variable_name=value
```

lưu ý không dùng dấu trắng trước và sau dấu bằng

- Khi sử dụng dùng tên biến đi kèm với dấu \$ đằng trước
- Ví dụ

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

```
var=100
```

```
echo $var
```

- Xóa một biến dùng

```
unset variable_name
```


Biến (3)

- Cách đặt tên biến
 - Khởi đầu bằng chữ cái hoặc dấu gạch ngang dưới
 - Theo sau là chữ cái, chữ số hoặc dấu gạch ngang dưới
 - Phân biệt in hoa in thường
 - Các biến môi trường được đánh dấu bằng cách tên của chúng chứa toàn chữ cái in hoa

Biến (4)

- Đọc biến từ bàn phím dùng

```
read variable_name
```

- Muốn in thông báo kèm theo dùng *-p*

```
read -p "type a number: " numbervar  
echo "your number is $numbervar"
```

- Có thể đọc nhiều biến một lúc

```
read -p "type names: " name1 name2 name3
```

- Muốn dữ liệu nhập vào dưới chế độ gõ mật khẩu

```
read -s -p "type your password: " passw
```

- Có thể đọc giá trị từ một biến khác

```
server="SP1 SP2 SP3"
```

```
read -r ns1 ns2 ns3 <<"$servers"
```

Biến *\$servers* sẽ được tách ra thành các *token* nhờ các nhận biết tách chuỗi *seperators* được lueu trong biến *\$IFS*

Biến (5)

- Khai báo biến số nguyên `declare -i x=10`
- Khai báo hằng số

```
declare -r cons1=10
```

```
readonly cons2=20
```

Các hằng số sẽ không thể bị xóa

Cấu trúc rẽ nhánh (1)

- Cấu trúc của lệnh rẽ nhánh điều kiện

```
if condition1
then
    commands
elif condition2
then
    commands
elif condition3
then
    commands ...
else
    commands
fi
```

Cấu trúc rẽ nhánh

