

# Giới thiệu về Bash Shell Script

- Bash Shell Script là một ngôn ngữ lập trình kịch bản được sử dụng trên hệ điều hành Linux và Unix.
- Nó giúp tự động hóa các tác vụ và quy trình

## 1. Tạo file bash và cấp quyền cho nó có thể thực thi

- Để tạo một file bash script ta thêm vào file ta cần tạo đuôi .sh

```
# Tạo tại thư mục đang làm việc
```

```
touch bash.sh
```

```
# Tạo thư đường dẫn đến thư mục cần tạo
```

```
touch home\duongtn1512\script\bash.sh
```

- Cấp quyền cho file có thể chạy được

```
# Cấp cho người sử hữu file (owner) quyền chạy (execute)
```

```
Chmod u+x bash.sh
```

```
# Cấp cho nhóm của người sử hữu file (group) quyền chạy (execute)
```

```
Chmod g+x bash.sh
```

```
# Cấp cho những người khác trong server (other) quyền chạy (execute)
```

```
Chmod o+x bash.sh
```

- Ta sử dụng nano hoặc vim để truy cập vào và chỉnh sửa file

```
nano bash.sh
```

```
nano home\duongtn1512\script\bash.sh
```

- Mỗi đầu dòng của file.sh ta thêm `#!/bin/bash` để file có thể thực thi lệnh bash shell được viết ở phía dưới

Với Bash Shell Script, bạn có khả năng thực hiện các tác vụ đa dạng từ sao lưu dữ liệu, xử lý tệp tin, tạo thư mục và thậm chí tương tác với hệ thống và ứng dụng khác.

## 2. Cú pháp và quy tắc cơ bản khi viết Bash shell script.

### Biến

- Biến trong Bash Shell Script là một tên được sử dụng để lưu trữ một giá trị dữ liệu

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

```
echo "Vui lòng nhập một tên:"
```

```
read name
```

```
echo "Tên bạn là ${name} "
```

### Điều kiện

- Sử dụng câu lệnh if, else, và elif để thực hiện kiểm tra điều kiện trong script.

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

```
echo "Vui lòng nhập một số:"
```

```
read number
```

```
# Kiểm tra số nhập vào là chẵn hay lẻ
```

```
if (( number % 2 == 0 )); then
```

```
    echo "Số $number là số chẵn."
```

```
else
```

```
    echo "Số $number là số lẻ."
```

```
fi
```

## Vòng lặp

- Sử dụng vòng lặp `for`, để lặp qua một danh sách hoặc thực hiện một tác vụ lặp lại.
- Các cấu trúc điều kiện trong vòng lặp và cách sử dụng lệnh `break` và `continue`.

```
#!/bin/bash
for (( i=1; i<=5; i++ )); do
    echo "Số $i"
done
```

## Tạo một script tự động

- Một script tạo người dùng nhóm người dùng tạo một đường dẫn tạo các file trong đường dẫn đó trong đó có một file chấm SH phân quyền cho file đó có thể được người dùng và nhóm người dùng mới tạo thực thi:

```
#!/bin/bash
# Tạo người dùng mới
read -p "Tên người dùng mới: " username
sudo useradd $username

# Tạo nhóm người dùng mới
read -p "Tên nhóm người dùng mới:" groupname
sudo groupadd $groupname

# Tạo thư mục cho nhóm người dùng mới
read -p "Nhập tên thư mục của nhóm người dùng mới: " directory
sudo mkdir /home/$directory

# Tạo và cấp quyền cho thư mục của người dùng và nhóm mới
sudo chown $username:$groupname /home/$directory
sudo chmod 775 /home/$directory

# Tạo file shell script và cấp quyền cho người dùng và nhóm mới
read -p "Nhập tên file shell script: " scriptname
sudo touch /home/$directory/$scriptname.sh
sudo chown $username:$groupname /home/$directory/$scriptname.sh
sudo chmod 776 /home/$directory/$scriptname.sh

# Viết nội dung cho file shell script
echo "#!/bin/bash" >> /home/$directory/$scriptname.sh
echo "Hello, World!" >> /home/$directory/$scriptname.sh

# Kết thúc thông báo
echo "Đã tạo thành công người dùng: $username, nhóm người dùng: $groupname, thư mục home: $directory và file shell script: $scriptname."
```

# Giới thiệu về CRONTAB

Crontab là một tiện ích cho phép thực hiện các tác vụ một cách tự động theo định kỳ ở chế độ nền của hệ thống. Crontab là một file chứa đựng bảng biểu (schedule) của các entries được chạy. Bằng cách sử dụng các lệnh trong linux Crontab, ta có thể tạo những task chạy vào giờ cụ thể đặt trước, như vào giờ nào trong ngày vào thứ mấy trong tuần.

cron schedule nằm ở /var/spool/cron. Crontab file không cho phép bạn tạo hoặc chỉnh sửa trực tiếp với bất kỳ trình text editor nào, trừ khi dùng lệnh Crontab.

```
# Một số lệnh thường gặp
crontab -e # tạo chỉnh sửa các crontab
crontab -l # xem các crontab đã tạo
crontab -r # xóa file crontab
```

```
# Cài đặt crontab
apt install cron
```

```
# Kiểm tra trạng thái Cron đã được cài đặt
Systemctl status cron
```

```
# Khởi động lại dịch vụ
Systemctl restart cron
```

```
# Chạy crontab tự động mỗi khi reboot
service cron start
chkconfig cron on
```

Cấu trúc crontab file có 5 trường và câu lệnh được chạy định kỳ:

```
* * * * * command to be executed
- - - - -
| | | | |
| | | | +----- day of week (0 - 6) (Sunday=0)
| | | +----- month (1 - 12)
| | +----- day of month (1 - 31)
| +----- hour (0 - 23)
+----- min (0 - 59)
```

Nếu một cột được gán ký tự \* nó có nghĩa là tác vụ sau đó sẽ được chạy ở mọi giá trị cho cột đó

Field	Giải thích	Giá trị cho phép	
-----	-----	-----	
MIN	phút	0 to 59	
HOURL	Giờ	0 to 23	

DOM	Ngày trong tháng	1-31	
MON	Tháng	1-12	
DOW	Ngày trong tuần	0-6	
CMD	Lệnh	Các lệnh có thể thực hiện	

```
# Backup dữ liệu web vào ngày 07 tháng 10 lúc 08:11 AM
11 08 10 07 * /home/framgia.vn/backup
```

VD: backup dữ liệu 2 lần trong 1 ngày

```
# Tạo một tác vụ thực hiện 2 lần trong 1 ngày
00 07,21 * * * /home/framgia.vn/backup
```

VD: từ thứ 2 đến thứ 6

```
# Tạo một tác vụ chỉ thực hiện vào các giờ cụ thể
00 09-18 * * 1-5 /home/hostingaz.info/full-backup
```

# Một số giá trị thời gian cho crontab

Keyword	Equivalent	
-----	-----	
@yearly	0 0 1 1 *	
@daily	0 0 * * *	
@hourly	0 * * * *	
@reboot	chạy lúc khởi động	

```
# Tạo một tác vụ chạy vào phút đầu tiên của năm
@yearly /home/framgia.vn/backup
```

```
# Tạo một tác vụ chạy vào phút đầu tiên của tháng
@monthly /home/framgia.vn/backup
```

```
# Tạo một tác vụ chạy khi khởi động
@reboot CMD
```