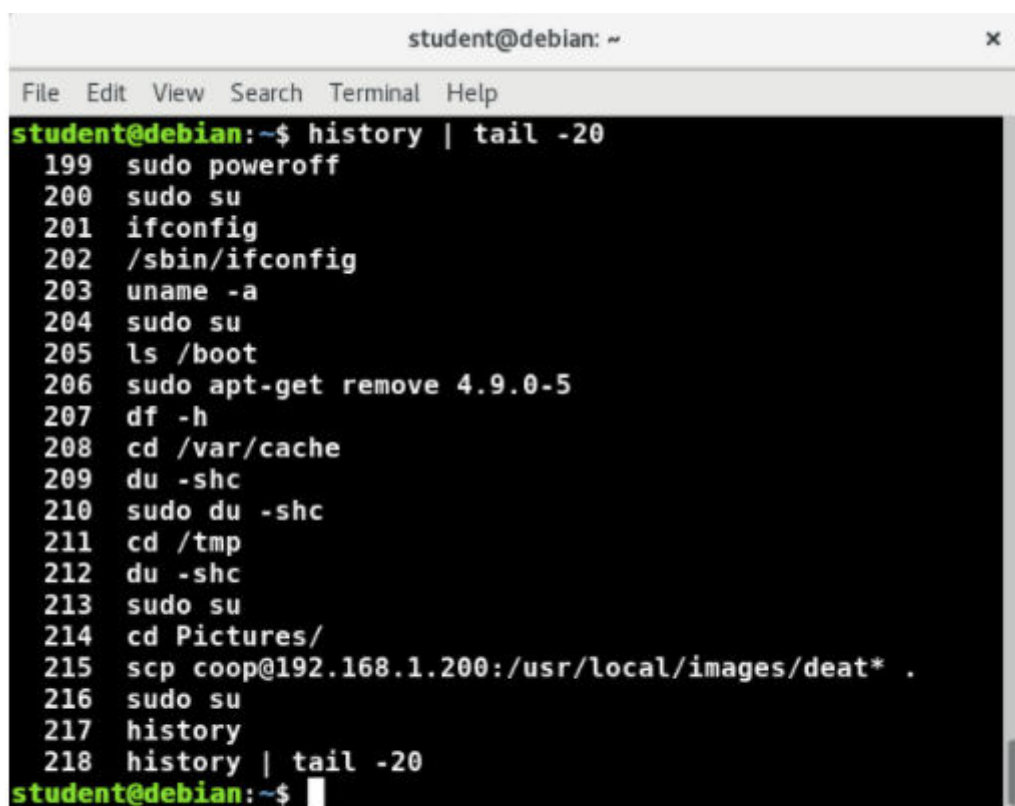


# Recalling Previous Commands

## Recalling Previous Commands:

1. **bash** keeps track of previously entered commands and statements in a **history buffer**.
2. You can recall previously used commands simply by using the **Up** and **Down** cursor keys.
3. To view the list of previously executed commands, you can just type **history** at the command line. The list of commands is displayed with the most recent command appearing last in the list.
4. This information is stored in **~/.bash\_history**.
5. If you have multiple terminals open, the commands typed in each session are not saved until the session terminates.



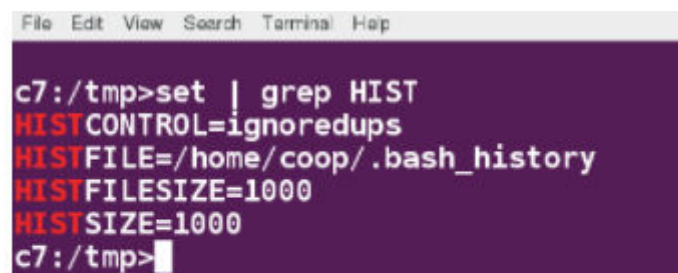
```
student@debian: ~  
File Edit View Search Terminal Help  
student@debian:~$ history | tail -20  
199 sudo poweroff  
200 sudo su  
201 ifconfig  
202 /sbin/ifconfig  
203 uname -a  
204 sudo su  
205 ls /boot  
206 sudo apt-get remove 4.9.0-5  
207 df -h  
208 cd /var/cache  
209 du -shc  
210 sudo du -shc  
211 cd /tmp  
212 du -shc  
213 sudo su  
214 cd Pictures/  
215 scp coop@192.168.1.200:/usr/local/images/deat* .  
216 sudo su  
217 history  
218 history | tail -20  
student@debian:~$
```

## Recalling Previous Commands

## Using History Environment Variables:

1. Several associated environment variables can be used to get information about the history file.
  - **HISTFILE** - The location of the history file.
  - **HISTFILESIZE** - The maximum number of lines in the history file (default 500).
  - **HISTSIZE** - The maximum number of commands in the history file.
  - **HISTCONTROL** - How commands are stored.
  - **HISTIGNORE** - Which command lines can be unsaved.

2. For a complete description of the use of these environment variables, see **man bash**.

A terminal window with a dark purple background and light blue text. The window title bar shows 'File Edit View Search Terminal Help'. The terminal content shows the command 'c7:/tmp>set | grep HIST' and its output: 'HISTCONTROL=ignoredups', 'HISTFILE=/home/coop/.bash\_history', 'HISTFILESIZE=1000', and 'HISTSIZE=1000'. The prompt 'c7:/tmp>' is visible at the bottom.

```
c7:/tmp>set | grep HIST
HISTCONTROL=ignoredups
HISTFILE=/home/coop/.bash_history
HISTFILESIZE=1000
HISTSIZE=1000
c7:/tmp>
```

## Using History Environment Variables

### Finding and Using Previous Commands:

1. Specific keys to perform various task:

Key	Usage
Up/Down arrow keys	Browse through the list of commands previously executed
!! (Pronounced as bang-bang)	Execute the previous command
CTRL-R	Search previously used commands

2. If you want to recall a command in the history list, but do not want to press the arrow key repeatedly, you can press **CTRL-R** to do a reverse intelligent search.
3. As you start typing, the search goes back in reverse order to the first command that matches the letters you have typed. By typing more successive letters, you make the match more and more specific.
4. The following is an example of how you can use the **CTRL-R** command to search through the command history:

```
$ ^R - (This all happens on 1 line)
$ (reverse-i-search)'s': sleep 1000 - (Searched for 's'; matched "sleep")
$ sleep 1000 - (Pressed Enter to execute the searched command)
$
```

## Executing Previous Commands:

1. The table describes the syntax used to execute previously used commands:

Syntax	Task
!	Start a history substitution
!\$	Refer to the last argument in a line
!n	Refer to the n <sup>th</sup> command line
!string	Refer to the most recent command starting with string

2. All history substitutions start with !. When typing the command: `ls -l /bin /etc /var`, !\$ will refer to `/var`, the last argument to the command.
3. Here are more examples:
  - `$ history`
    1. `echo $SHELL`
    2. `echo $HOME`
    3. `echo $PS1`
    4. `ls -a`
    5. `ls -l /etc/ passwd`
    6. `sleep 1000`
    7. `history`
  - `$ !1` (Execute command #1 above)  
`echo $SHELL`  
`/bin/bash`
  - `!sl` (Execute the command beginning with "sl")  
`sleep 1000`  
`$`

## Keyboard Shortcuts:

1. You can use keyboard shortcuts to perform different tasks quickly.
2. The table lists some of these keyboard shortcuts and their uses.
3. Note the case of the "hotkey" does not matter, e.g. doing **CTRL-a** is the same as doing **CTRL-A**.

Keyboard Shortcut	Task
CTRL-L	Clears the screen
CTRL-D	Exits the current shell
CTRL-Z	Puts the current process into suspended background
CTRL-C	Kills the current process
CTRL-H	Works the same as backspace
CTRL-A	Goes to the beginning of the line
CTRL-W	Deletes the word before the cursor
CTRL-U	Deletes from beginning of line to cursor position
CTRL-E	Goes to the end of the line
Tab	Auto-completes files, directories, and binaries