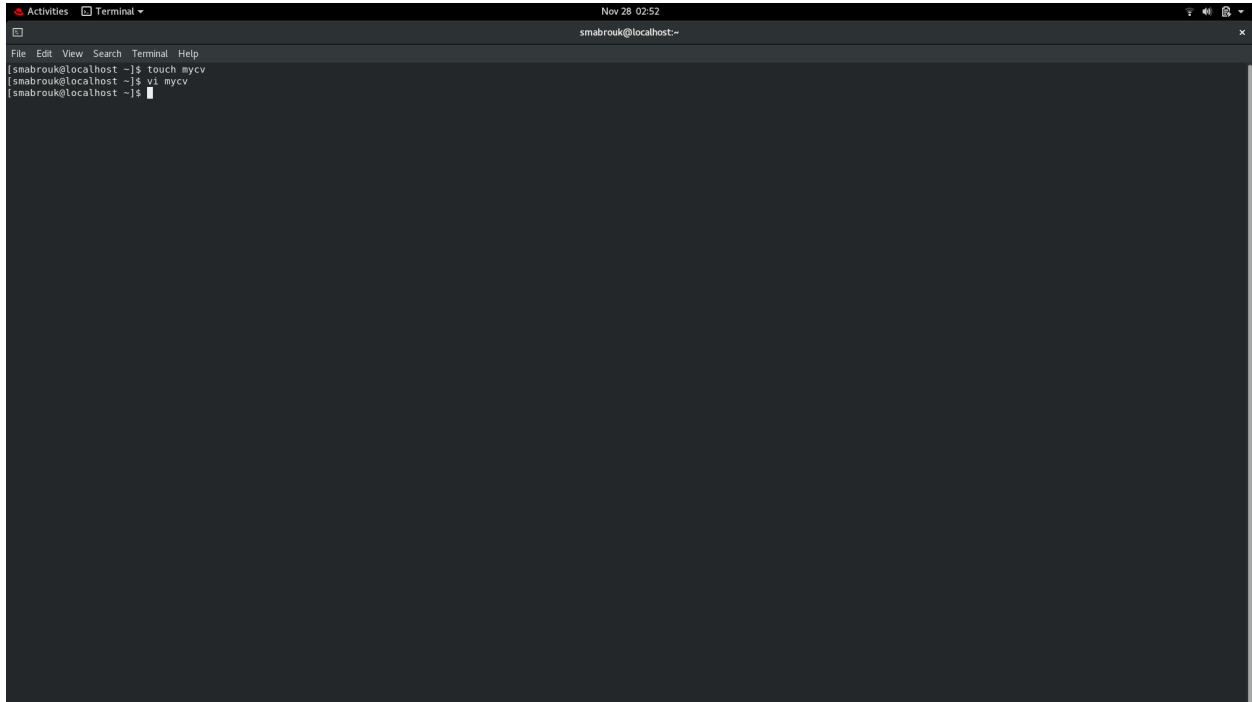


RHSA1 - Lab 3

1. Using vi write your CV in the file mycv.



```
Activities Terminal Nov 28 02:52 smabrouk@localhost:~  
File Edit View Search Terminal Help  
[smabrouk@localhost ~]$ touch mycv  
[smabrouk@localhost ~]$ vi mycv  
[smabrouk@localhost ~]$
```

2. Open mycv file using vi command then: Without using arrows state how to:

- Move the cursor down one line at time.
(:j)
- Move the cursor up one line at time.
(:k)
- Search for word age
(:/age)
- Step to line 5 (assuming that you are in line 1 and file is more than 5 lines).
(:5)
- Delete the line you are on and line 5.
(:dd then :5d)
- How to step to the end of line and change to writing mode in one-step.
(:A)

3. List the available shells in your system.

```
Nov 28 02:54
smabrouk@localhost:~$ cat /etc/shells
/bin/sh
/bin/bash
/usr/bin/sh
/usr/bin/bash
smabrouk@localhost:~$
```

4. List the environment variables in your current shell.

```
Nov 28 02:57
smabrouk@localhost:~$
File Edit View Search Terminal Help
LANG=en_US.UTF-8
GDM_LANG=en_US.UTF-8
HISTCONTROL=ignoredups
DISPLAY=:0
HOSTNAME=localhost.localdomain
COLORTERM=truecolor
USERNAME=smabrouk
XDG_VTNR=2
SSH_AUTH_SOCK=/run/user/1000/ssh-agent
XDG_SESSION_ID=2
USER=smabrouk
DESKTOP_SESSION=gnome
WAYLAND_DISPLAY=wayland-0
GNOME_TERMINAL_SCREEN=/org/gnome/Terminal/screen/5cd208a2_4d56_4c57_8cde_ce8f639a3060
PWD=/home/slabrouk
SSH_ASKPASS=/usr/libexec/openssh/gnome-ssh-askpass
HOME=/home/slabrouk
XDG_SESSION_TYPE=wayland
XDG_DATA_DIRS=/home/slabrouk:/usr/share/flatpak/exports/share:/usr/local/share:/usr/share:/var/lib/flatpak/exports/share:/usr/share/flatpak/exports/share
XDG_SESSION_DESKTOP=gnome
GJS_DEBUG_OUTPUT=stderr
MAIL=/var/spool/mail/slabrouk
VTE_VERSION=5204
SHELL=/bin/bash
TERM=xterm-256color
QT_IM_MODULE=ibus
XMODIFIERS=@=ibus
XDG_CURRENT_DESKTOP=GNOME
GNOME_TERMINAL_SERVICE=:1.96
XDG_SEAT=seat0
SHULVL=2
GDMSESSION=gnome
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
LOGNAME=smabrouk
DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus
XDG_RUNTIME_DIR=/run/user/1000
XAUTHORITY=/run/user/1000/.mutter-Xwaylandauth.SEMRMI
PATH=/home/slabrouk:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin:/usr/bin:/usr/sbin:/usr/lib/snapd/snap/bin
HISTSIZE=1000
GJS_DEBUG_TOPICS=:JS ERROR:JS LOG
SESSION_MANAGER=local/unix:/tmp/.ICE-unix/2386,unix:/tmp/.ICE-unix/2386
LESSOPEN=|/usr/bin/lesspipe.sh %s
BASH_FUNC_which%%=() { _which;
eval $(which declare) } | /usr/bin/which --tty-only --read-alias --read-functions --show-tilde --show-dot $@
}
~/usr/bin/printenv
smabrouk@localhost:~$
```

5. List all of the environment variables for the bash shell.

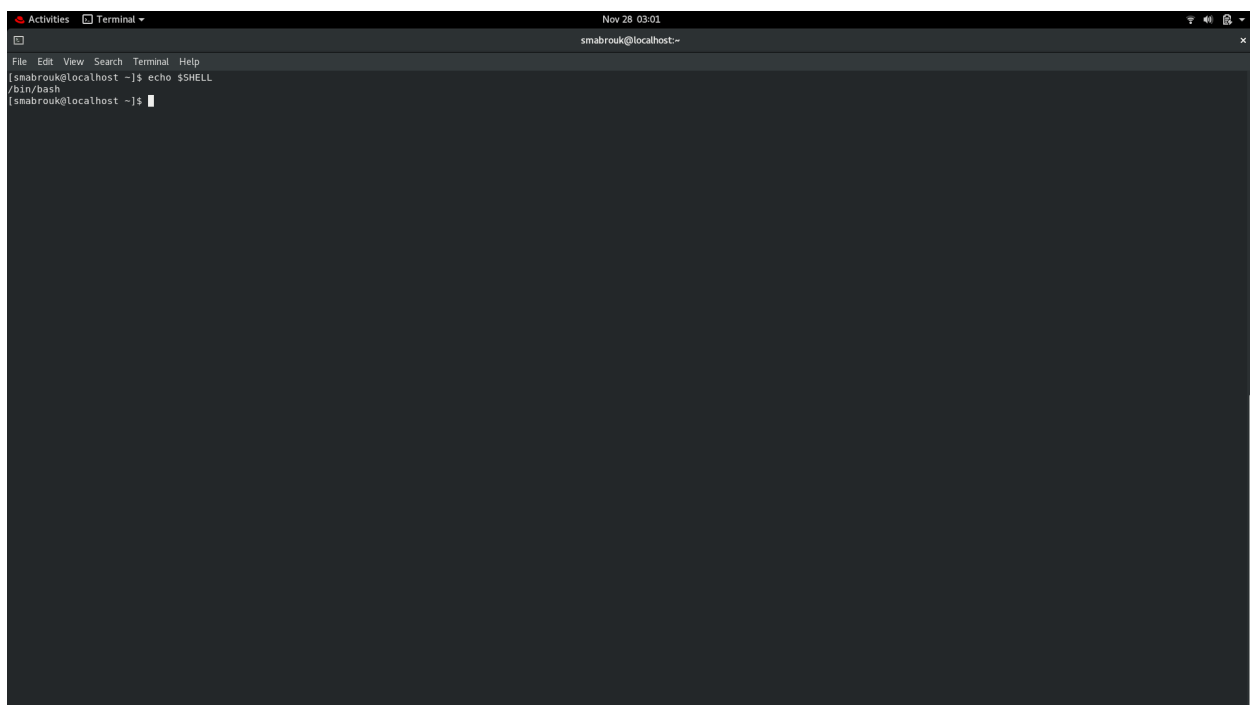


```
Nov 28 03:00
smabrouk@localhost:~$ set
The following variables are set by the shell:
BASH Expands to the full filename used to invoke this instance of bash.
BASHOPTS A colon-separated list of enabled shell options. Each word in the list is a valid argument for the -s option to the shopt builtin command (see SHELL BUILTIN COMMANDS below). The options appearing in BASHOPTS are those reported as on by shopt. If this variable is in the environment when bash starts up, each shell option in the list will be enabled before reading any startup files. This variable is read-only.
BASHPID Expands to the process ID of the current bash process. This differs from $$ under certain circumstances, such as subshells that do not require bash to be re-initialized.
BASH_ALIASES An associative array variable whose members correspond to the internal list of aliases as maintained by the alias builtin. Elements added to this array appear in the alias list; however, unsetting array elements currently does not cause aliases to be removed from the alias list. If BASH_ALIASES is unset, it loses its special properties, even if it is subsequently reset.
BASH_ARGC An array variable whose values are the number of parameters in each frame of the current bash execution call stack. The number of parameters to the current subroutine (shell function or script) executed with . or source is at the top of the stack. When a subroutine is executed, the number of parameters passed is pushed onto BASH_ARGC. The shell sets BASH_ARGC only when in extended debugging mode (see the description of the extdebug option to the shopt builtin below)
BASH_ARGV An array variable containing all of the parameters in the current bash execution call stack. The final parameter of the last subroutine call is at the top of the stack; the first parameter of the initial call is at the bottom. When a subroutine is executed, the parameters supplied are pushed onto BASH_ARGV. The shell sets BASH_ARGV only when in extended debugging mode (see the description of the extdebug option to the shopt builtin below)
BASH_CMDS An associative array variable whose members correspond to the internal hash table of commands as maintained by the hash builtin. Elements added to this array appear in the hash table; however, unsetting array elements currently does not cause command names to be removed from the hash table. If BASH_CMDS is unset, it loses its special properties, even if it is subsequently reset.
BASH_COMMAND The command currently being executed or about to be executed, unless the shell is executing a command as the result of a trap, in which case it is the command executing at the time of the trap.
BASH_EXECUTION_STRING The command argument to the -c invocation option.
BASH_LINENO An array variable whose members are the line numbers in source files where each corresponding member of FUNCNAME was invoked. ${BASH_LINENO[i]} is the line number in the source file ${BASH_SOURCE[i]} where ${FUNCNAME[i]} was called (or ${BASH_LINENO[i]} if referenced within another shell function). Use LINENO to obtain the current line number.
BASH_LOADABLES_PATH A colon-separated list of directories in which the shell looks for dynamically loadable builtins specified by the enable command.
BASH_REMATCH An array variable whose members are assigned by the =~ binary operator to the [[ conditional command. The element with index 0 is the portion of the string matching the entire regular expression. The element with index n is the portion of the string matching the nth parenthesized subexpression. This variable is read-only.
BASH_SOURCE An array variable whose members are the source filenames where the corresponding shell function names in the FUNCNAME array variable are defined. The shell function ${FUNCNAME[i]} is defined in the file ${BASH_SOURCE[i]} and called from ${BASH_SOURCE[i-1]}.
BASH_SUBSHELL Incremented by one within each subshell or subshell environment when the shell begins executing in that environment. The initial value is 0.
BASH_VERSIONINFO A read-only array variable whose members hold version information for this instance of bash. The values assigned to the array members are as follows:
  BASH_VERSIONINFO[0] The major version number (the release).
  BASH_VERSIONINFO[1] The minor version number (the version).
  BASH_VERSIONINFO[2] The patch level.
  BASH_VERSIONINFO[3] The build version.
  BASH_VERSIONINFO[4] The release status (e.g., beta1).
  BASH_VERSIONINFO[5] The value of MACHTYPE.
BASH_VERSION Expands to a string describing the version of this instance of bash.
Manual page bash(1) line 427 (press h for help or q to quit)
```

6. What are the commands that list the value of a specific variable?

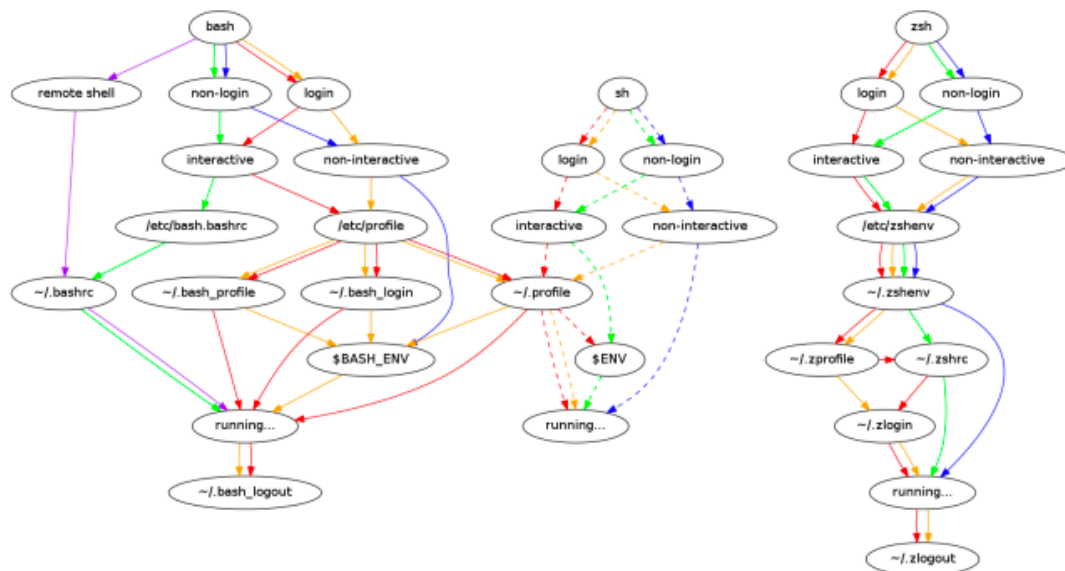
echo <variable_name>
printenv <variable_name>

7. Display your current shell name.

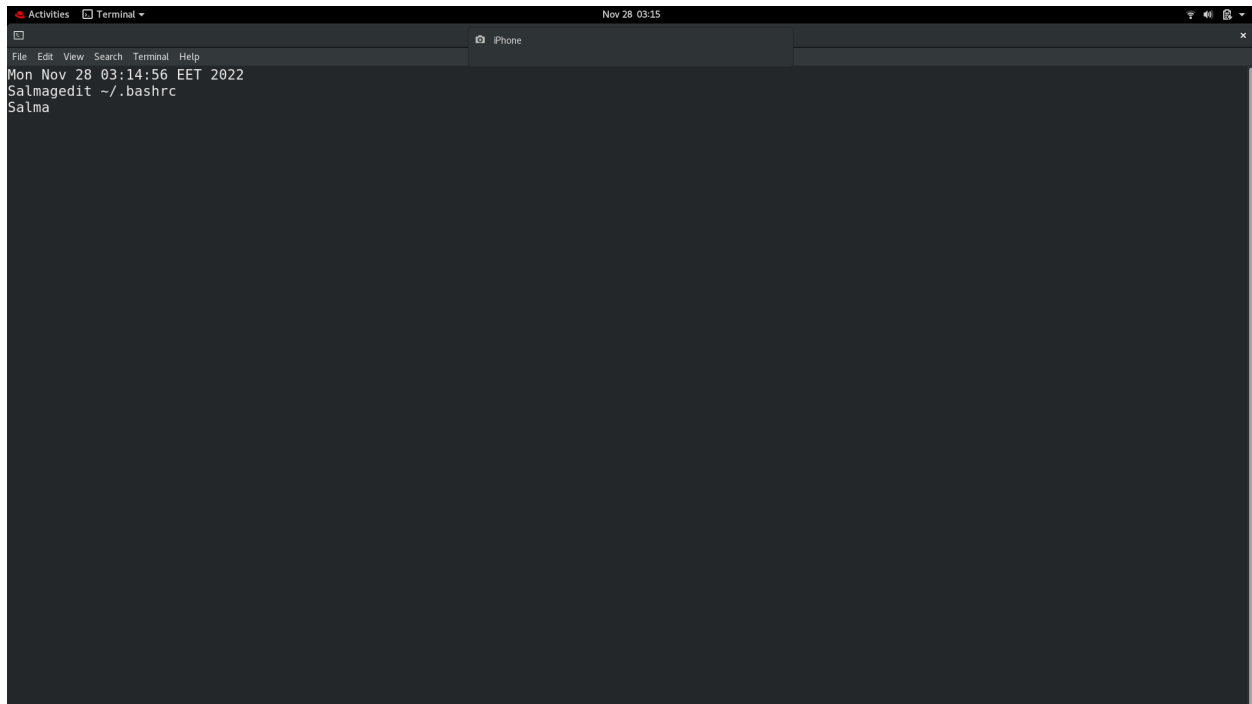


```
Nov 28 03:01
smabrouk@localhost:~$ echo $SHELL
/bin/bash
smabrouk@localhost:~$
```

8. State the initialization files of: sh, ksh, bash.



9. Edit in your profile to display date at login and change your prompt permanently.

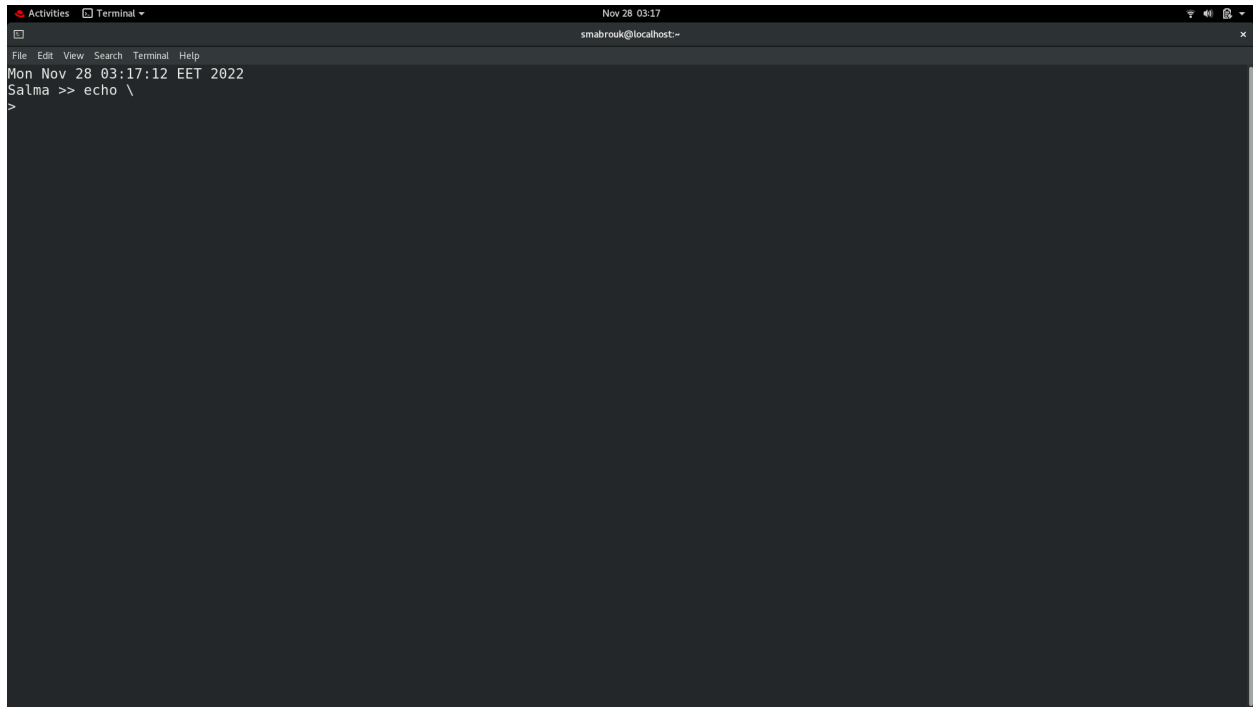


10. Execute the following command :

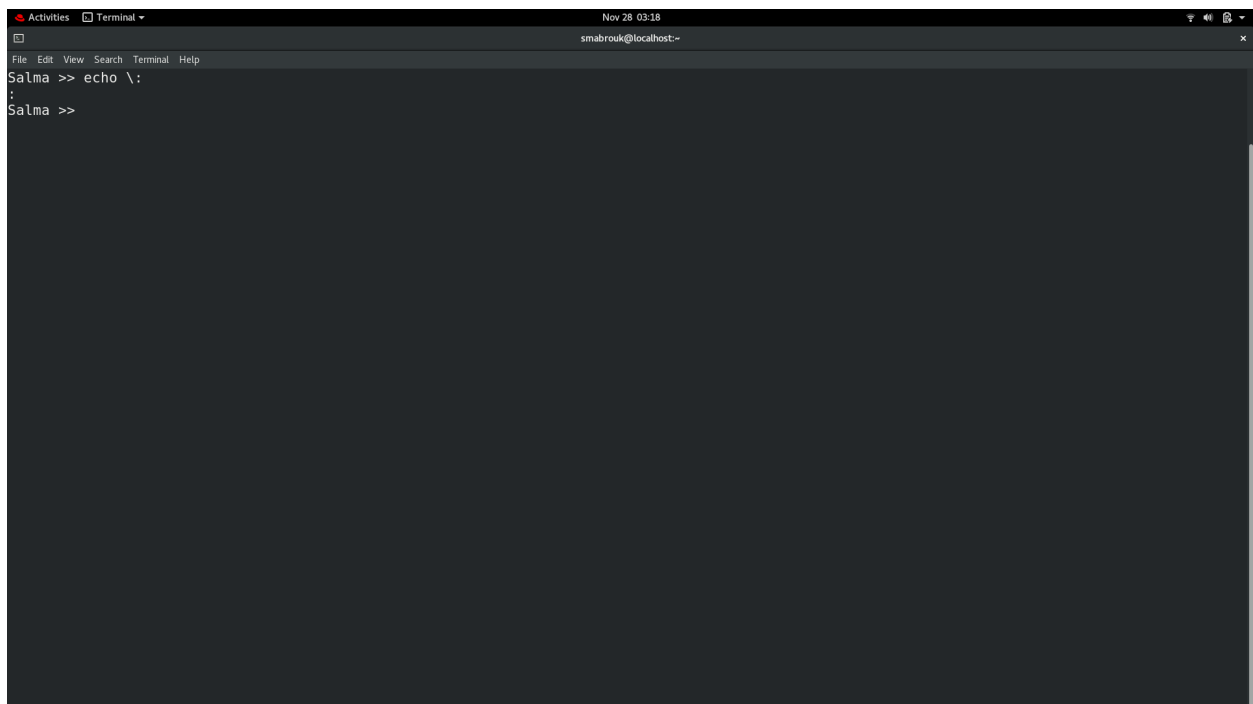
`echo \` then press enter

What is the purpose of `\` ?

Notice the prompt `>` what is that? and how can you change it from `>` to `:`.

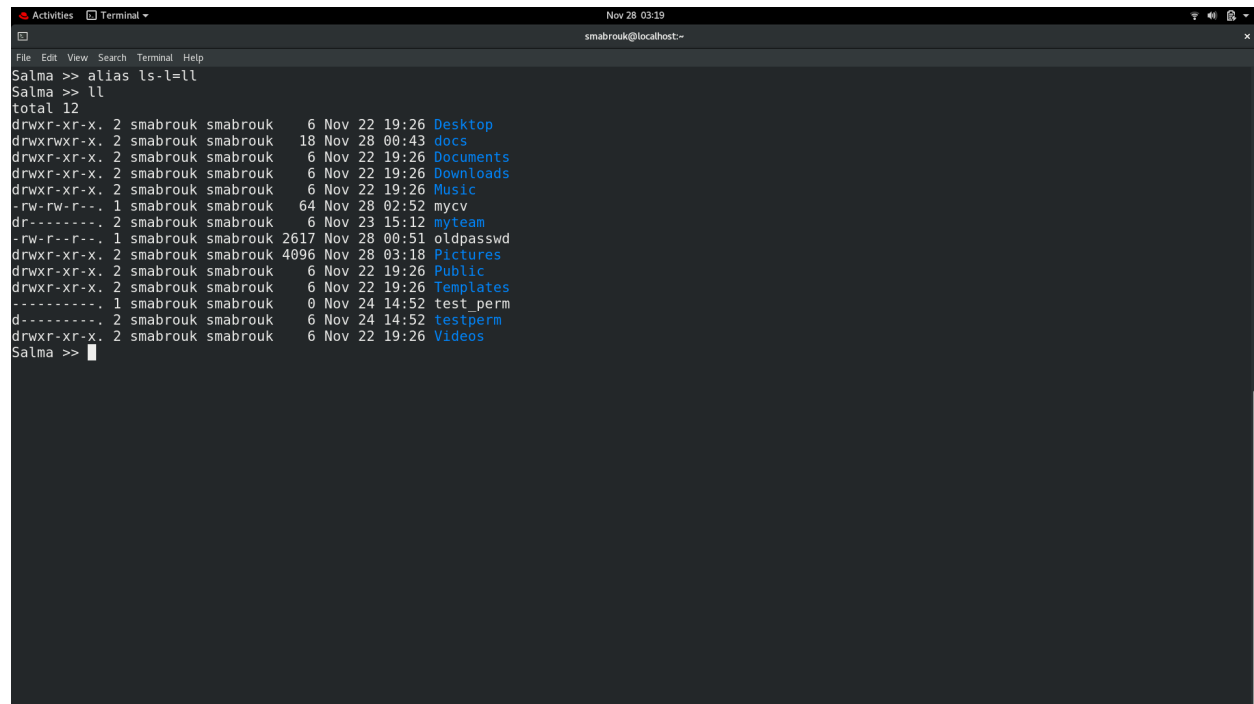


```
Mon Nov 28 03:17:12 EET 2022
Salma >> echo \
>
```



```
Salma >> echo \:
:
Salma >>
```

11. Create a Bash shell alias named ls for the “ls -l” command



A terminal window titled "Terminal" with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Nov 28 03:19, smabrouk@localhost). The user "Salma" enters the command `alias ls-l=ll` to create an alias. Then, they enter `ll`, which displays a long listing of files and directories in the current directory. The output shows permissions, owner, group, size, date, and filename for various system and user directories.

```
Salma >> alias ls-l=ll
Salma >> ll
total 12
drwxr-xr-x. 2 smabrouk smabrouk  6 Nov 22 19:26 Desktop
drwxrwxr-x. 2 smabrouk smabrouk 18 Nov 28 00:43 docs
drwxr-xr-x. 2 smabrouk smabrouk  6 Nov 22 19:26 Documents
drwxr-xr-x. 2 smabrouk smabrouk  6 Nov 22 19:26 Downloads
drwxr-xr-x. 2 smabrouk smabrouk  6 Nov 22 19:26 Music
drwxr-xr-x. 1 smabrouk smabrouk  64 Nov 28 02:52 mycv
dr----- 2 smabrouk smabrouk   6 Nov 23 15:12 myteam
-rw-r--r-- 1 smabrouk smabrouk 2617 Nov 28 00:51 oldpasswd
drwxr-xr-x. 2 smabrouk smabrouk 4096 Nov 28 03:18 Pictures
drwxr-xr-x. 2 smabrouk smabrouk  6 Nov 22 19:26 Public
drwxr-xr-x. 2 smabrouk smabrouk  6 Nov 22 19:26 Templates
----- 1 smabrouk smabrouk   0 Nov 24 14:52 test_perm
d----- 2 smabrouk smabrouk   6 Nov 24 14:52 testperm
drwxr-xr-x. 2 smabrouk smabrouk  6 Nov 22 19:26 Videos
Salma >>
```