ls -a
(all)
Lists all the files (including .* files)

ls -S
(size)
Lists the biggest files first

ls -l
(long)
Long listing (type, date, size,
owner, permissions)

ls -r
(reverse)
Reverses the sort order

ls -t
(time)
Lists the most recent files first

ls ltr
(options can be
combined)
Long listing, most recent files at
the end

ls *txt
The shell first replaces *txt by all the file and directory names
ending by txt (including .txt), except those starting with .,
and then executes the ls command line.
ls d
.*
Lists all the files and directories starting with .
d
tells ls not to display the contents of directories.
cat ?.log
Displays all the files which names start by 1 character and end by .log

./
The current directory. Useful for commands taking a directory
argument. Also sometimes useful to run commands in the current
directory (see later).
So ./readme.txt and readme.txt are equivalent.
../
The parent (enclosing) directory. Always belongs to the . directory

(see ls a).
Only reference to the parent directory.
Typical usage:
cd ..

cd <dir>
Changes the current directory to <dir>.
cd
Gets back to the previous current directory.
pwd
Displays the current directory ("working directory").

cp <source_file> <target_file>
Copies the source file to the target.
cp file1 file2 file3 ... dir
Copies the files to the target directory (last argument).
cp -i
(interactive)
Asks for user confirmation if the target file already exists
cp -r
<source_dir> <target_dir> (recursive)
Copies the whole directory.

Smart directory copy with rsync
rsync (remote sync) has been designed to keep in sync
directories on 2 machines with a low bandwidth connection.
Only copies files that have changed. Files with the same size are compared
by checksums.
Only transfers the blocks that differ within a file!
Can compress the transferred blocks
Preserves symbolic links and file permissions: also very useful for copies on
the same machine.
Can work through ssh (secure remote shell). Very useful to update the
contents of a website, for example.

rsync examples (1)
rsync -a /home/arvin/sd6_agents/ /home/sydney/misc/
-a:archive mode. Equivalent to rlptgoD...
easy way to tell you want
recursion and want to preserve almost everything.
rsync -Pav --delete/home/steve/ideas//home/bill/my_ideas/
-P: --partial (keep partially transferred files) and --progress
(show progress during transfer)
--delete: delete files in the target which don't exist in the source.

Caution: directory names should end with / . Otherwise, you get a my_ideas/ideas/ directory at the destination.

mv and rm commands
mv <old_name> <new_name> (move)
Renames the given file or directory.
mv -i (interactive) If the new file already exits, asks for user confirm
rm file1 file2 file3 ... (remove)
Removes the given files.
rm -i (interactive)
Always ask for user confirm.
rm -r dir1 dir2 dir3 (recursive)
Removes the given directories with all their contents.

Creating and removing directories
mkdir dir1 dir2 dir3 ... (make dir)
Creates directories with the given names.
rmdir dir1 dir2 dir3 ... (remove dir)
Removes the given directories
Safe: only works when directories and empty.
Alternative: rm -r (doesn't need empty directories).

Displaying file contents
Several ways of displaying the contents of files.
cat file1 file2 file3 ... (concatenate)
Concatenates and outputs the contents of the given files.
more file1 file2 file3 ...
After each page, asks the user to hit a key to continue.
Can also jump to the first occurrence of a keyword
(/ command).
less file1 file2 file3 ...
Does more than more with less.
Doesn't read the whole file before starting.
Supports backward movement in the file (? command).

The grep command
grep <pattern> <files>
Scans the given files and displays the lines which match the given pattern.
grep error *.log
Displays all the lines containing error in the *.log files
grep -i error *.log
Same, but case insensitive
grep -ri error .
Same, but recursively in all the files in . and its subdirectories
grep -v info *.log
Outputs all the lines in the files except those containing info.

The sort command
sort <file>
Sorts the lines in the given file in character order and outputs them.
sort -r <file>
Same, but in reverse order.
sort -ru <file>
u: unique. Same, but just outputs identical lines once.
More possibilities described later!

Fedora coomands:

gedit /etc/inittab

gedit /etc/rc.local

su -

passwd

- **Check if a service is running**:
- service *servicename* status
- **Starting a service**:
- service *servicename* start
- **Stopping a service**:
- service *servicename* stop

ifconfig

setup

cat /etc/sysconfig/network-scripts/ifcfg-eth0

THe OUTPUT will look like this (i have it set to DHCP)

```
DEVICE=eth0
BOOTPROTO=dhcp
HWADDR=00:D0:B7:08:09:BB
ONBOOT=yes
```

OUTPUT for static IP (example)

```
#
# File: ifcfg-eth0
#
DEVICE=eth0
IPADDR=192.168.1.100
NETMASK=255.255.255.0
BOOTPROTO=static
ONBOOT=yes
```