

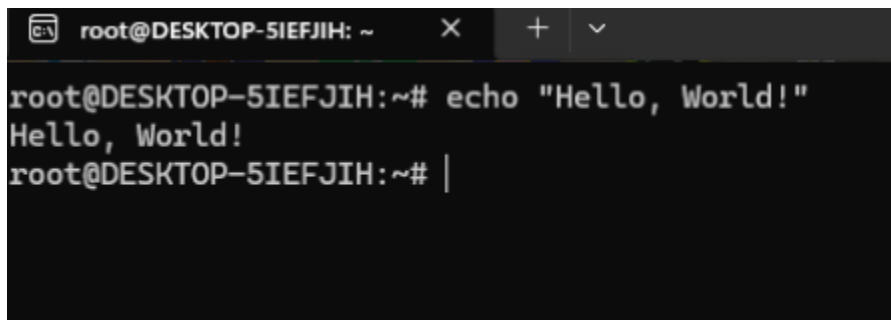
## ASSIGNMENT-2

### CONCEPTS OF OPERATING SYSTEM

#### PART A

- **Echo "Hello, World!"**

It will display the argument (Hello World) passed through the command echo.

A terminal window with a dark background. The title bar shows 'root@DESKTOP-5IEFJIH: ~' and window control buttons. The terminal text shows the command 'echo "Hello, World!"' being entered and executed, resulting in the output 'Hello, World!'. The prompt 'root@DESKTOP-5IEFJIH:~#' is visible on the next line.

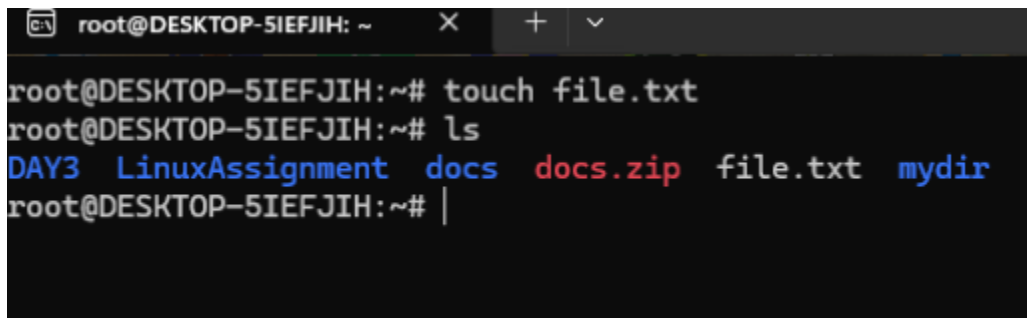
```
root@DESKTOP-5IEFJIH:~# echo "Hello, World!"
Hello, World!
root@DESKTOP-5IEFJIH:~# |
```

- **Name="Productive"**

This command assigning the label productive to that specific element.

- **Touch file.txt**

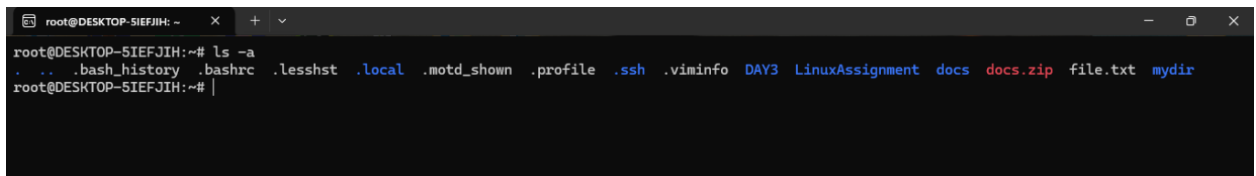
It will create the file named 'file.txt'.

A terminal window with a dark background. The title bar shows 'root@DESKTOP-5IEFJIH: ~' and window control buttons. The terminal text shows the command 'touch file.txt' being entered and executed. Then, the command 'ls' is entered, showing the output 'DAY3 LinuxAssignment docs docs.zip file.txt mydir'. The prompt 'root@DESKTOP-5IEFJIH:~#' is visible on the next line.

```
root@DESKTOP-5IEFJIH:~# touch file.txt
root@DESKTOP-5IEFJIH:~# ls
DAY3 LinuxAssignment docs docs.zip file.txt mydir
root@DESKTOP-5IEFJIH:~# |
```

- **ls -a**

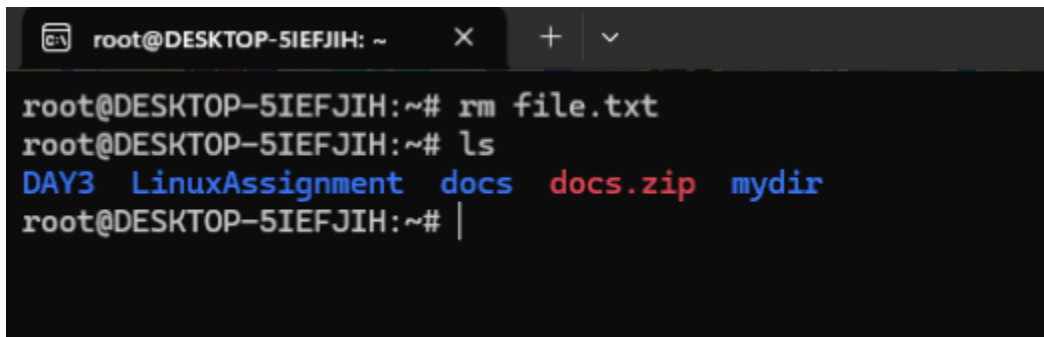
List information about the files (the current directory by default).



```
root@DESKTOP-5IEFJIH: ~  
root@DESKTOP-5IEFJIH:~# ls -a  
.  ..  .bash_history  .bashrc  .lesshst  .local  .motd_shown  .profile  .ssh  .viminfo  DAY3  LinuxAssignment  docs  docs.zip  file.txt  mydir  
root@DESKTOP-5IEFJIH:~#
```

- **rm file.txt**

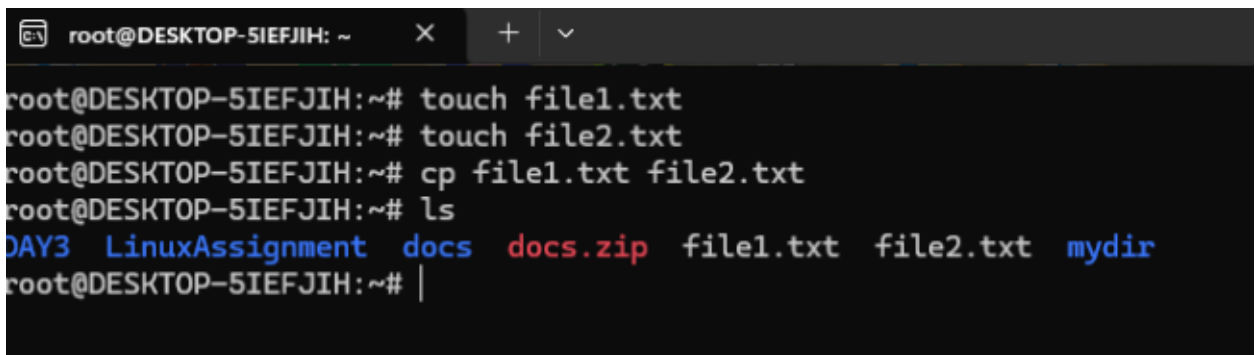
It will remove the file named 'file.txt'



```
root@DESKTOP-5IEFJIH: ~  
root@DESKTOP-5IEFJIH:~# rm file.txt  
root@DESKTOP-5IEFJIH:~# ls  
DAY3  LinuxAssignment  docs  docs.zip  mydir  
root@DESKTOP-5IEFJIH:~#
```

- **cp file1.txt file2.txt**

It will copy the contents of file1 to file2.



```
root@DESKTOP-5IEFJIH: ~  
root@DESKTOP-5IEFJIH:~# touch file1.txt  
root@DESKTOP-5IEFJIH:~# touch file2.txt  
root@DESKTOP-5IEFJIH:~# cp file1.txt file2.txt  
root@DESKTOP-5IEFJIH:~# ls  
DAY3  LinuxAssignment  docs  docs.zip  file1.txt  file2.txt  mydir  
root@DESKTOP-5IEFJIH:~#
```

- **mv file.txt/path/to/directory/**

It will move files and directories from one directory to another or to rename a file or directory

- **chmod 755 script.sh**

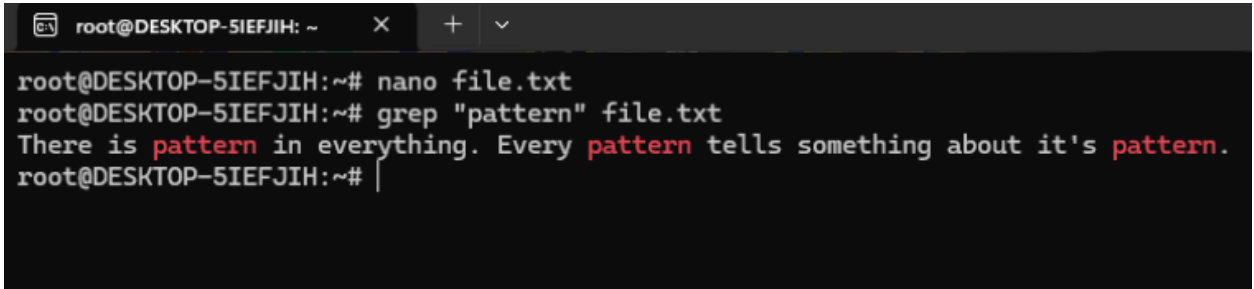
chmod is used to change the permission. It gives read,write and execute permission to owner.

Only Read and execute permission to both group and others of file script.sh

```
root@DESKTOP-5IEFJIH: ~  
root@DESKTOP-5IEFJIH:~# chmod 755 script.sh  
root@DESKTOP-5IEFJIH:~# ls -l  
total 36  
drwxr-xr-x 2 root root 4096 Feb 27 13:30 DAY3  
drwxr-xr-x 2 root root 4096 Feb 27 14:05 LinuxAssignment  
drwxr-xr-x 2 root root 4096 Feb 27 13:14 docs  
-rw-r--r-- 1 root root 586 Feb 28 11:27 docs.zip  
-rw-r--r-- 1 root root 82 Mar 2 10:09 file.txt  
-rw-r--r-- 1 root root 103 Mar 2 10:21 file1.txt  
-rw-r--r-- 1 root root 86 Mar 2 10:30 file2.txt  
-rw-r--r-- 1 root root 68 Mar 2 10:25 file2.txt.save  
drwxr-xr-x 2 root root 4096 Mar 2 10:17 mydir  
-rwxr-xr-x 1 root root 0 Mar 2 10:02 script.sh  
root@DESKTOP-5IEFJIH:~# |
```

- **grep “pattern” file.txt**

The grep command searches for a string “pattern” in groups of files. When it finds a pattern that matches in more than one file, it prints the name of the file, followed by a colon.

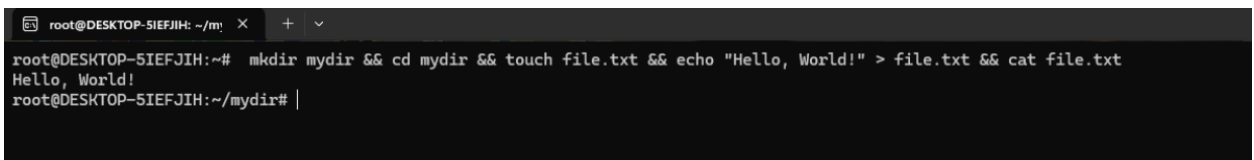


```
root@DESKTOP-5IEFJIH: ~  
root@DESKTOP-5IEFJIH:~# nano file.txt  
root@DESKTOP-5IEFJIH:~# grep "pattern" file.txt  
There is pattern in everything. Every pattern tells something about it's pattern.  
root@DESKTOP-5IEFJIH:~#
```

- **Kill PID**

It sends the TERM signal to the specified process, giving it a chance to shut down in an orderly manner.

- **Mkdir mydir && cd mydir && touch file.txt && echo “Hello, World!” > file.txt && cat file.txt**



```
root@DESKTOP-5IEFJIH: ~/my  
root@DESKTOP-5IEFJIH:~# mkdir mydir && cd mydir && touch file.txt && echo "Hello, World!" > file.txt && cat file.txt  
Hello, World!  
root@DESKTOP-5IEFJIH:~/mydir#
```

- `ls -l | grep ".txt"`

```
root@DESKTOP-5IEFJIH: ~  
root@DESKTOP-5IEFJIH:~# ls -l | grep ".txt"  
-rw-r--r-- 1 root root 82 Mar 2 10:09 file.txt  
-rw-r--r-- 1 root root 0 Mar 2 09:57 file1.txt  
-rw-r--r-- 1 root root 0 Mar 2 09:58 file2.txt  
root@DESKTOP-5IEFJIH:~# |
```

- **cat file1.txt file2.txt | sort | uniq**

first cat will copy the contents of file1 into file2. After that it sorts the contents in a unique way.

```
root@DESKTOP-5IEFJIH: ~  
root@DESKTOP-5IEFJIH:~# cat file1.txt  
cdac juhu  
cdac khargar  
cdac bangalore  
cdac pune  
cdac bangalore  
cdac delhi  
cdac noida  
cdac bhubneshwar  
root@DESKTOP-5IEFJIH:~#  
root@DESKTOP-5IEFJIH:~#  
root@DESKTOP-5IEFJIH:~# cat file2.txt  
IIT bombay  
IIT delhi  
IIT kharagpur  
cdac mumbai  
cdac delhi  
cdac bhubneshwar  
cdac pune  
  
root@DESKTOP-5IEFJIH:~# cat file1.txt file2.txt | sort | uniq  
  
IIT bombay  
IIT delhi  
IIT kharagpur  
cdac bangalore  
cdac bhubneshwar  
cdac delhi  
cdac juhu  
cdac khargar  
cdac mumbai  
cdac noida  
cdac pune  
root@DESKTOP-5IEFJIH:~# |
```

- **ls -l | grep "^d"**

It is used to current directory with the user.

```
root@DESKTOP-5IEFJIH: ~  
root@DESKTOP-5IEFJIH:~# ls -l | grep "^d"  
docs  
docs.zip  
root@DESKTOP-5IEFJIH:~# |
```

- **grep -r "pattern" /path/to/directory/**

It is used to search for a specific pattern inside all files within a given directory and its subdirectories.

- **cat file1.txt file2.txt | sort | uniq -d**

It is used to find duplicate lines from both the files i.e., file1.txt and file2.txt.

```
root@DESKTOP-5IEFJIH: ~  
root@DESKTOP-5IEFJIH:~# cat file1.txt file2.txt | sort | uniq -d  
cdac bangalore  
cdac bhubneshwar  
cdac delhi  
cdac pune  
root@DESKTOP-5IEFJIH:~# |
```

- **chmod 644 file.txt**

chmod is used to change the permission. It will set read and write permission for owner. Only read permission to both the group and other.

```
root@DESKTOP-SIEFJIH: ~  
root@DESKTOP-SIEFJIH:~# chmod 644 file.txt  
root@DESKTOP-SIEFJIH:~# ls -a  
. .bash_history .lessht .motd_shown .ssh DAY3 docs file.txt file2.txt mydir  
.. .bashrc .local .profile .viminfo LinuxAssignment docs.zip file1.txt file2.txt.save script.sh  
root@DESKTOP-SIEFJIH:~# ls -l  
total 36  
drwxr-xr-x 2 root root 4096 Feb 27 13:30 DAY3  
drwxr-xr-x 2 root root 4096 Feb 27 14:05 LinuxAssignment  
drwxr-xr-x 2 root root 4096 Feb 27 13:14 docs  
-rw-r--r-- 1 root root 586 Feb 28 11:27 docs.zip  
-rw-r--r-- 1 root root 82 Mar 2 10:09 file.txt  
-rw-r--r-- 1 root root 103 Mar 2 10:21 file1.txt  
-rw-r--r-- 1 root root 86 Mar 2 10:30 file2.txt  
-rw-r--r-- 1 root root 68 Mar 2 10:25 file2.txt.save  
drwxr-xr-x 2 root root 4096 Mar 2 10:17 mydir  
-rwxr-xr-x 1 root root 0 Mar 2 10:02 script.sh  
root@DESKTOP-SIEFJIH:~#
```

- **cp -r source\_directory destination\_directory**

It is used to copy a directory and its contents (including subdirectories and files) to another location in Linux.

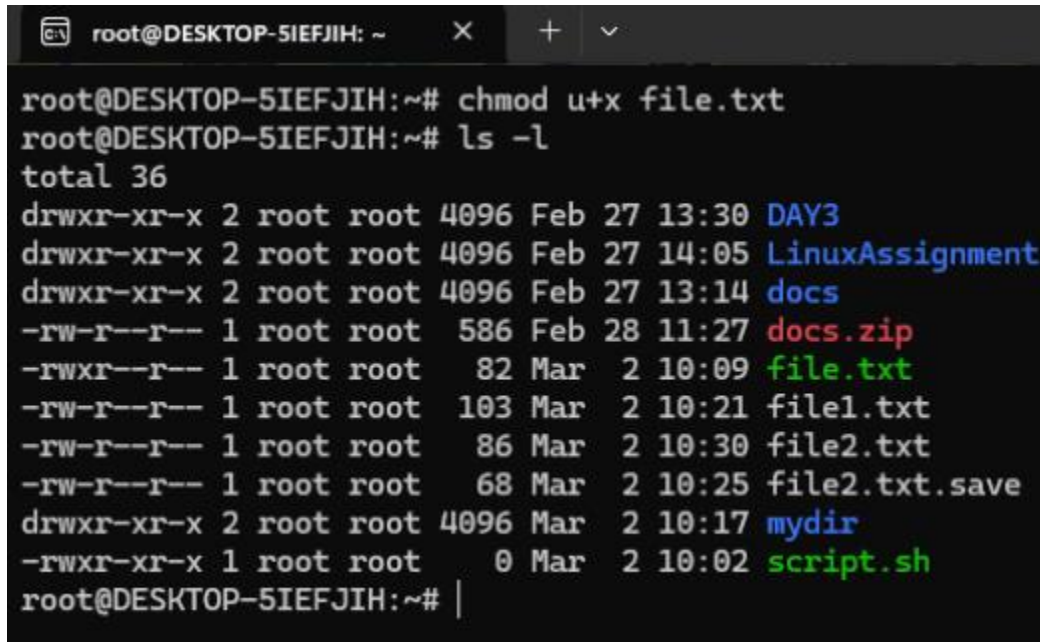
- **find /path/to/search -name "\*.txt"**

It will find .txt file in the given directory.



- **chmod u+x file.txt**

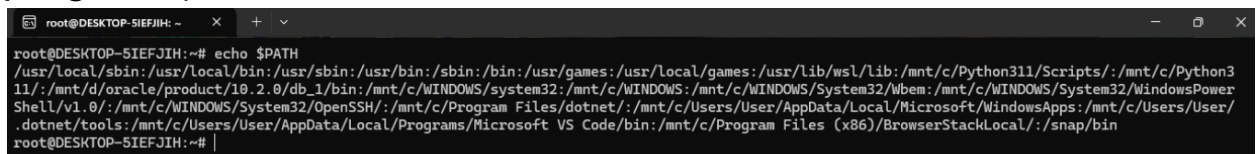
chmod is used to change the permission. In this command user is able to execute the file.

A terminal window titled 'root@DESKTOP-5IEFJIH: ~' with standard window controls. The terminal shows the execution of 'chmod u+x file.txt' followed by 'ls -l'. The output of 'ls -l' lists several files and directories with their permissions, owners, sizes, and timestamps. The file 'file.txt' is highlighted in green in the original image, showing permissions '-rwxr--r--'.

```
root@DESKTOP-5IEFJIH:~# chmod u+x file.txt
root@DESKTOP-5IEFJIH:~# ls -l
total 36
drwxr-xr-x 2 root root 4096 Feb 27 13:30 DAY3
drwxr-xr-x 2 root root 4096 Feb 27 14:05 LinuxAssignment
drwxr-xr-x 2 root root 4096 Feb 27 13:14 docs
-rw-r--r-- 1 root root 586 Feb 28 11:27 docs.zip
-rwxr--r-- 1 root root 82 Mar 2 10:09 file.txt
-rw-r--r-- 1 root root 103 Mar 2 10:21 file1.txt
-rw-r--r-- 1 root root 86 Mar 2 10:30 file2.txt
-rw-r--r-- 1 root root 68 Mar 2 10:25 file2.txt.save
drwxr-xr-x 2 root root 4096 Mar 2 10:17 mydir
-rwxr-xr-x 1 root root 0 Mar 2 10:02 script.sh
root@DESKTOP-5IEFJIH:~# |
```

- **echo \$PATH**

It will print the environment variable that stores a list of directories where executable files (commands, scripts, programs) are located.

A terminal window titled 'root@DESKTOP-5IEFJIH: ~' with standard window controls. The terminal shows the execution of 'echo \$PATH', which outputs a long string of directory paths separated by colons, representing the system's PATH environment variable.

```
root@DESKTOP-5IEFJIH:~# echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/usr/lib/wsl/lib:/mnt/c/Python311/Scripts:/mnt/c/Python311:/mnt/d/oracle/product/10.2.0/db_1/bin:/mnt/c/WINDOWS/system32:/mnt/c/WINDOWS:/mnt/c/WINDOWS/System32/Wbem:/mnt/c/WINDOWS/System32/WindowsPowerShell/v1.0:/mnt/c/WINDOWS/System32/OpenSSH:/mnt/c/Program Files/dotnet:/mnt/c/Users/User/AppData/Local/Microsoft/WindowsApps:/mnt/c/Users/User/.dotnet/tools:/mnt/c/Users/User/AppData/Local/Programs/Microsoft VS Code/bin:/mnt/c/Program Files (x86)/BrowserStackLocal/./snap/bin
root@DESKTOP-5IEFJIH:~# |
```

## **PART B**

### **❖ Identify true and false:**

1. ls is used to list files and directories in a directory. **TRUE**
2. mv is used to move files and directories. **TRUE**
3. cd is used to copy files and directories. **FALSE**
4. pwd stands for "print working directory" and displays the current directory. **TRUE**
5. grep is used to search for patterns in files. **TRUE**
6. chmod 755 file.txt gives read, write, and execute permissions to the owner, and read and execute permissions to group and others. **TRUE**
7. mkdir -p directory1/directory2 creates nested directories, creating directory2 inside directory1 if directory1 does not exist. **TRUE**
8. rm -rf file.txt deletes a file forcefully without confirmation. **TRUE**

## ❖ Identify the Incorrect Commands:

1. **chmodx** is used to change file permissions.  
**chmod** is used to change file permissions  
.
2. **cpy** is used to copy files and directories.  
**cp** is used to copy files and directories.
3. **mkfile** is used to create a new file.  
**touch file.txt** is used to create a new file.
4. **catx** is used to concatenate files.  
**cat** is used to concatenate files.
5. **rn** is used to rename files.  
**mv** is used to rename files.

## PART- E

1. Consider the following processes with arrival times and burst times:

PID	ARRIVAL TIME	BURST TIME
P1	0	5
P2	1	3
P3	2	6

Calculate the average waiting time using First-Come, First-Served (FCFS) scheduling.

Ans. Avg. waiting time= 3.33

2. Consider the following processes with arrival times and burst times:

PID	ARRIVAL TIME	BURST TIME
P1	0	3
P2	1	5
P3	2	1
P4	3	4

Calculate the average turnaround time using Shortest Job First (SJF) scheduling.

Ans. Avg Turnaround Time = 5.5.

3. Consider the following processes with arrival times, burst times, and priorities (lower number indicates higher priority):

PID	ARRIVAL TIME	BURST TIME	PRIORITY
P1	0	6	3
P2	1	4	1
P3	2	7	4
P4	3	2	2

Calculate the average waiting time using Priority Scheduling.

**Ans. Avg. Waiting Time = 7.75.**

4. Consider the following processes with arrival times and burst times, and the time quantum for Round Robin scheduling is 2 units:

PID	ARRIVAL TIME	BURST TIME
P1	0	4
P2	1	5
P3	2	2
P4	3	3

Calculate the average turnaround time using Round Robin scheduling.

**Ans. Avg. Turnaround time = 9.**

5. Consider a program that uses the `fork()` system call to create a child process. Initially, the parent process has a variable `x` with a value of 5. After forking, both the parent and child processes increment the value of `x` by 1. What will be the final values of `x` in the parent and child processes after the `fork()` call

Ans.

Let `x = 5`.

Call `fork()`.

Both parent and child increase `x` by 1.

child `x = 6`.

parent `x = 6`.

Final values of `x`:

child process, `x = 6`.

parent process, `x = 6`.