

Mission Name:- Standard Streams and File Descriptors

* Every process - every running instance of a command - interacts with its environment by means of communication channels called streams.

* Standard streams:-

- > standard input → stdin
- > standard output → stdout
- > standard error → stderr

* To redirect stderr to a file, we need to precede the ">" operator by 2.

Eg:- `rm /dev/null 2> err.`

* We can make 2 redirections separately but it is also possible to do it together in one command.

Eg:- `ls /dev/null /home/inexisting >all_op 2>all_op`

The above method should be avoided as the ^{stating} `>` character for 2nd redirection gets eaten up.

* A parent process is a process that spawned another process.

* File Descriptors (fd):-

0 for stdin

1 for stdout

2 for stderr

* Normally:- all has same default output

standard input $\rightarrow 0 \rightarrow /dev/pts/0$

standard output $\rightarrow 1 \rightarrow /dev/pts/0$

standard error $\rightarrow 2 \rightarrow /dev/pts/0$

After command $> filename:-$

standard input $\rightarrow 0 \rightarrow /dev/pts/0$

standard output $\rightarrow 1 \rightarrow filename$

standard error $\rightarrow 2 \rightarrow /dev/pts/0$

After command $2> filename:-$

standard input $\rightarrow 0 \rightarrow /dev/pts/0$

standard output $\rightarrow 1 \rightarrow /dev/pts/0$

standard error $\rightarrow 2 \rightarrow filename$

* fd's can be duplicated.

* we can duplicate fd 2 to fd 1 by using $"2>&1"$.

This action will send both output and error to the same file.

Eg:- $ls /dev/null /home/inexistent >all-output_v2$
 $2>&1$

* $tr \rightarrow$ transliteration

* Input Redirection can be done using $"<"$.

Eg:- $tr [:lower:] [:upper:] <sorted_stdin >mad_vow$