Setpgid:

Format

```
#define _POSIX_SOURCE
#include <unistd.h>

int setpgid(pid_t pid, pid_t pgid);
```

pid_t *pid* is the process ID (PID) of the process whose PGID you want to change. This must either be the caller of setpgid() or one of its children, and it must be in the caller's session. It cannot be the PID of a session leader. If *pid* is zero, the system uses the PID of the process calling setpgid().
pid_t *pgid* is the new PGID you want to assign to the process identified by *pid*. If *pgid* indicates an existing process group, it must be in the caller's session. If *pgid* is zero, the system uses the PID of the process indicated by *pid* as the ID for the new process group. The new group is created in the caller's session.

setpgid() sets the process group ID of the process specified by *pid* to *pgid*. If *pid* is zero, the process ID of the current process is used. If *pgid* is zero, the process ID of the process specified by *pid*is used. If **setpgid**() is used to move a process from one process group to another (as is done by some shells when creating pipelines), both process groups must be part of the same session. In this case, the *pgid* specifies an existing process group to be joined and the session ID of that group must match the session ID of the joining process. Sets the process group ID (PGID) of a process within the session of the calling process, so you can reassign a process to a different process group, or start a new process group with the specified process as its group leader.

The rules for how setpgid() may be used are a bit complicated.

1. A process may set the process group of itself or one of its children. It may not change the process group for any other process on the system, even if the process calling setpgid()has root privileges.
2. A session leader may not change its process group.
3. A process may not be moved into a process group whose leader is in a different session from itself. In other words, all the processes in a process group must belong to the same session.

Getpgrp:

used for getting and setting the process group ID (PGID) of a process. The preferred, POSIX.1-specified ways of doing this are: **getpgrp**(void), for retrieving the calling process's PGID; and **setpgid**(), for setting a process's PGID.

Usage:

getpgid() returns the process group ID of the process specified by *pid*. If *pid* is zero, the process ID of the current process is used.

The call **setpgrp**() is equivalent to **setpgid(0,0)**.

Similarly, **getpgrp**() is equivalent to **getpgid**(0) . Each process group is a member of a session and each process is a member of the session of which its process group is a member.

Return value:

On success, **setpgid**() and **setpgrp**() return zero. On error, -1 is returned, and *errno* is set appropriately.

The POSIX.1 **getpgrp**() always returns the PGID of the caller.

getpgid(), and the BSD-specific **getpgrp**() return a process group on success. On error, -1 is returned, and *errno* is set appropriately.