

Why are `exit` and `cd` handled as built-in commands and no child processes are forked? What would happen if you forked children for these commands?

---

If you `cd` / `exit` on forked children, the child processes will `cd` / `exit`. However, the parent process that runs the terminal will be unaffected. We want our terminal to `cd` / `exit`, when we call these cmds so we hardcode them to act on the parent process and spawn children for everything else.

Why should you use `killpg(getpid(), SIGTERM);` before `main()` exits?

---

When a child spawns in the parent process, it inherits the `group_id` of the parent process. This continues down to all children of the children, and so on, unless we manually change the `group_id` of a process.

`killpg(getpid(), SIGTERM)` will terminate all processes in the `group_id` of the parent. Since we don't change any `group_id`, all our processes will share a single `group_id`. Thus we terminate all children and the parent using `killpg`, preventing any zombies that might be left behind.