# \*\*\*Khái niệm cơ bản về tiến trình\*\*\*

Là một chương trình chạy trên hệ thống. Mỗi chương trình chạy hệ thống đều có tiến trình tương ứng

Được định danh bởi một PID phục vụ cho việc giám sát, điều khiển và quản lý của hệ thống

# \*\*\*Các loại tiến trình\*\*\*

Có 2 loại tiến trình gồm tiến trình cha và tiến trình con

Khi một tiến trình sinh ra 1 tiến trình khác thì:

* Tiến trình ban đầu đượ gọi là tiến trình cha và được định danh bởi PPID(Parent)
* Tiến trình mới sẽ được gọi là tiến trình con

Tương tác giữa tiến trình cha và tiến trình con

* Khi tiến trình con đang chạy thì tiến trình cha sẽ chờ
* Khi tiến trình con hoàn thành thì tiến trình cha sẽ tiếp tục thực thi và tiến trình con sẽ được kết thúc.

Các trạng thái của tiến trình

* Run: tiến trình đang chạy
* Sleep: Tiến trình đang ở trạng thái chờ hoạt động
* Zombie: Tiến trình mà tiền trình cha đã kết thúc
* Stop: Tiến trình đã dừng hoạt động

# \*\*\*Mục đích của quản lí tiến trình\*\*\*

* Xác định những tiến trình đang chạy trên hệ thống
* Trạng thái của các tiến trình
* Tài nguyên mà các tiến trình đang sử dụng
* Kiểm soát tiến trình
* Kết thúc các tiến tình không mong muốn
* Thực thi các nghiệp vụ theo lịch
* Kiểm soát tải của hệ thống, giúp kiểm soát và tối ưu lại hệ thống, tránh ảnh hượng đến nghiệp vụ khác.

# \*\*\*Các câu lệnh quản lý tiến trình\*\*\*

* `ps -ef`: liệt kê các tiến trình đang hoạt động trên hệ thống

\*Các thông số\*

- UID: User quản lý tiến trình

- PID: ID tiến trình

- PPID: ID tiến trình cha

- STIME: Start time

- c: CPU đang thực thi

- TIME: Thời gian thực thi của tiến trình

- CMD: Câu lệnh thực thi

- `ps -fu oracle`: Hiện thị các tiến trình của user oracle

- `pgrep`: Hiển thị thông tin các tiến trình của một chương trình

Ví dụ liệt kê các tiến trình chạy java: `pgrep -l java`

* `top`: Hiện thị trạng thái của các tiến trình đang hoạt động trên hệ thống

Đây là một công cụ đa năng cho phép hiển thị tất cả những tải của hệ điều hành

* -`Kill` : Để xử lí tiến trình nó không stop theo lệnh được hoặc bị treo chúng ta cần kill để tránh ảnh hưởng đến hệ thống.
* Có 2 loại gồm: `kill -9` và `kill -15`
* - kill -9: kết thúc tiến trình ngay lập tức.
* - kill -15: gửi tín hiệu kết thúc đến tiến trình, chờ tiến trình thực hiện cleanup và kết thúc(trong một số trường hợp kill -15 không thể tắt được tiến trình, buộc phải sử dụng kill -9 để dừng tiến trình.
* - `pkill`: kết thúc một tiến trình hoặc nhiều tiến trình theo tên hoặc thuộc tính của tiến trình
* Ví dụ: kill tất cả các tiến trình java trên máy chủ: pkill java
* Lưu ý: cần đặc biệt thẩn trọng trong khi sử dụng `pkill`. Trong hầu hết các trường hợp nên sử dụng `ps` hoặc `pgrep` để xác định tiến trình và sử dụng `kill` để kết thúc tiến trình.
* # \*\*\*Đặt lịch tiến trình chạy tự động\*\*\*
* Giảm thiểu thời gian tác động vào hệ thống. Hệ thống yêu cầu onl mà ta không thể onl ta cần có tiến trình để nó rà soát xem tiến trình có chết không. Nếu chết thì nó tự bật lên.
* ## Mục đích
* - Thiết lập lịch cho 1 tác vụ được thực hiện vào 1 thời điểm xác định
* - Giúp tự động hóa các công việc mang tính lặp lại
* - Hỗ trợ cho việc vận hành, giám sát hệ thống.
* ## Các kiểu đặt để nó chạy tự động
* - AT: đặt lịch cho 1 lệnh hoặc 1 tiến trình thực thi 1 lần duy nhất vào 1 thời điểm xác định.
* - Crontab: Cho phép chạy một lệnh hay một nhóm các lệnh. Đặt lịch cho 1 lệnh hoặc 1 tiến trình thực thi lặp lại nhiều lần
* \*Nguyên lý hoạt động\*
* - Sử dụng daemon cron
* - Lịch thực thi được lưu trong file crontab
* - Daemon cron đọc file crontab để thực thi theo lịch và câu lệnh cấu hình crontab