

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ctr+alt+f1 وارد محیط اصلی لینوکس می‌شوید

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

passwd : پسورد هر کاربر را میتوانید عوض کنید اگر تنها passwd را وارد کنید پسورد کاربری که با آن لاگین هستید عوض می‌شود اما اگر همراه با username وارد شود (passwd user\_name) پسورد کاربر مورد نظرتان عوض می‌شود

=====================================================================================

دستورات مقدماتی :

shell : به معنی پوسته میباشد که از یه هسته (صدف) مراقبت میکند . من اگر بخواهم یک دستور را به هسته ی سیستم عامل ارسال کنم بایستی از طریق یک رابط کاربری این کار را انجام دهم در اینجا منظور از رابط کاربری shell میباشد در سیستم عامل های مختلف shell های مختلف وجود دارد bash معروفترین shell در خانواده ی linux میباشد

command : دستوراتی هست که از طریق shell به هسته ارسال میکنیم از دستورات میتوانند option یا arguments داشته باشند

نکته مهمی که در سیستم عامل لینوکس باید به آن توجه کرد این است که بین حروف بزرگ و کوچک تفاوت میگذارد

در ترمینال قبل از @ نام کاربری شما که با آن login کرده‌اید و بعد از @ اسم کامپیوتر شما میباشد

اگر به عنوان کاربر root لاگین کرده باشید در انتها علامت # را مشاهده میکنید و اگر به عنوان کاربر معمولی login کرده باشید در انتها علامت $ را مشاهده میکنیم

ls ====> list cd ====> change directory clear or ctrl+l ====> ترمینال را پاک میکند

ls -l ====> با جزییات بیشتر نمایش میدهد در قسمت اول سطوح دسترس ls \*.txt ls file??.txt ==>هر ؟ یک کاراکتر

cd ~ ====> در هر کجایی که باشید شماره به دایرکتوری هوم کاربر میبرد cat file00.txt file01.txt

با استفاده از دستور cat میتوانید محتویات فایل‌ها را مشاهده کنید همینطور مشاهده محتویات ۲ فایل به صورت همزمان

echo sajjad ==== > با استفاده از این دستور میتوانید یک خروجی دریافت کنید مثلاً سجاد خروجی این مثال است

توجه داشته باشید که فاصله (space) به معنی option بعدی میباشد و در دستور echo خروجی مناسب شما چاپ نشود

شما میتوانید ماننده زبان‌های برنامه نویسی به متغیر ها مقدار دهی کنید و متغیر ها را فرا خوانی کنید(بعد از علامت $)

a=sajjad

b=ahmad

echo $a ====> sajjad echo $a$b ====> sajjadahmad echo $a $b ===> sajjad ahmad

میتوانیم دستوراتی که وارد میکنید را در یک خط قرار دهیم اما خروجیشان در یک خط نباشد (با استفاده از ;)

echo $a; echo $b

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

توجه شود که متغیر هایی که تعریف می‌شوند با بسته شدن ترمینال از بین می‌روند

متغیر هایی وجود دارد که به صورت سراسری تعریف می‌شوند ماننده PATH که مسیری که دستورات در آن‌ها دخیره شده است را نشان میدهد

echo $PATH

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

دستور زیر نشان میدهد که هر دستور (مثلاً ls ) در کدام پوشه ذخیره شده است

which ls

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

دستور history تاریخچه دستوراتی که در ترمینال وارد شده است را نمایش میدهد

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_