

جزوه درس

آزمایشگاه سیستم عامل

این جزوه حاصل جمع آوری و کنار هم گذاشتن مطالب مربوط به جزوات و سایت‌های مختلف می باشد که اسامی سایت‌های مربوطه، پایان هر بخش در قسمت منابع ذکر شده است.

فهرست مطالب

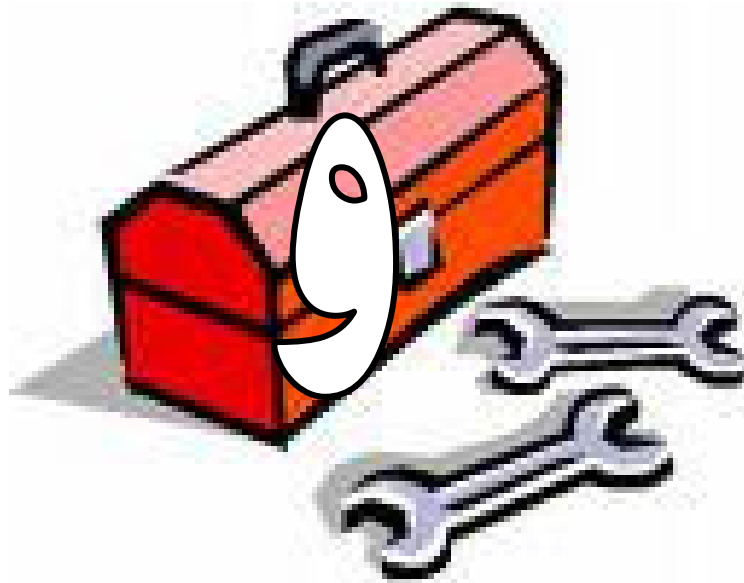
بخش ۱: Setup System و اعمال تغییرات در آن	۵
۱-۱ فرایند بوت و برنامه مدیر بوت	۶
۲-۱ BIOS	۶
۳-۱ CMOS	۷
۱-۳-۱ بروز آوری CMOS (برنامه نصب)	۱۰
۴-۱ اعمال تغییرات در Setup سیستم	۱۱
۵-۱ سوالات	۱۸
۶-۱ دستور کار	۱۸
۷-۱ پروژه برنامه نویسی	۱۸
۸-۱ منابع	۱۸
بخش ۲: کار با رجیستری در ویندوز	۱۹
۱-۲ رجیستری چیست؟	۲۰
۲-۲ برخی از مزایای پایگاه داده رجیستری	۲۱
۳-۲ محل فایل های رجیستری	۲۱
۴-۲ ویرایش اطلاعات موجود در رجیستری	۲۱
۵-۲ ایجاد Shortcut برای برنامه Regedit	۲۲
۶-۲ ساختار و اجزای رجیستری در برنامه Regedit	۲۲
۶-۲ اجزای تشکیل دهنده رجیستری	۲۲
۷-۲ انواع داده در بانک رجیستری	۲۵
۸-۲ انجام عملیات مختلف در برنامه Regedit	۲۵
۸-۲-۱ گرفتن نسخه پشتیبان از رجیستری و بازیابی آن	۲۵
۸-۲-۲ انتقال دادن تنظیمات رجیستری	۲۵
۸-۲-۳ تغییر دادن مقدار یک پارامتر	۲۶
۸-۲-۴ جستجوی یک پارامتر در Regedit	۲۶
۸-۲-۵ تغییر نام و یا حذف یک کلید یا یک پارامتر	۲۶
۸-۲-۶ تعریف یک کلید یا یک پارامتر	۲۶
۹-۲ مشاهده نتیجه تغییرات اعمال شده در رجیستری	۲۶
۱۰-۲ چگونه تغییرات در رجیستری را غیر فعال کنیم	۲۷
۱۱-۲ چند مثال از تغییرات در رجیستری	۲۸
۱-۱۱-۲ غیر فعال کردن Task Manager	۲۸
۲-۱۱-۲ غیر فعال کردن دستور Shut Down	۲۸
۳-۱۱-۲ از کار انداختن راست کلیک در Desktop	۲۹
۴-۱۱-۲ برداشتن منوی File از مرورگر ویندوز	۲۹
۵-۱۱-۲ از کار انداختن راست کلیک در Desktop	۲۹
۶-۱۱-۲ تنظیم پنجره خوش آمدگویی ویندوز	۲۹
۱۲-۲ برنامه نویسی رجیستری به وسیله C#	۳۰
۱-۱۲-۲ کلاس Registry	۳۰
۲-۱۲-۲ کلاس Registrkey	۳۰
۳-۱۲-۲ متدهای کلاس Registrykey	۳۱
متد OpenSubKey()	۳۱

	DeleteSubKey() / CreateSubKey() & متدهای
۲۹.....	DeleteSubKeyTree()
۳۱.....	GetSubKeyNames() متد
۳۲.....	GetValue() / GetValuesNames() متدهای
۳۳.....	SetValue() متد
۳۳.....	SubKeyCount و Valuecount های Property استفاده از
۳۴.....	۴-۱۲-۲ مثالی از یک برنامه Registry با C#
۳۶.....	۱۵-۲ دستور کار
۳۶.....	۱۶-۲ پروژه برنامه نویسی
۳۶.....	۱۷-۲ منابع
۳۷.....	بخش ۳: آشنایی با لینوکس و دستورات آن
۳۸.....	۱-۳ تاریخچه لینوکس
۳۶.....	۱-۱-۳ آغاز داستان
۳۹.....	۲-۱-۳ کودک جدید در افق
۴۰.....	۳-۱-۳ مقایسه و توسعه
۴۲.....	۴-۱-۳ طلوع لینوکس روی میزی (Desktop Linux)
۴۲.....	۵-۱-۳ لینوکس در جهان سوم
۴۲.....	۶-۱-۳ از میز کار تا ابر کامپیوترها
۴۲.....	۷-۱-۳ داستان ادامه دارد
۴۳.....	۸-۱-۳ نشان لینوکس
۴۳.....	۳-۳ کدام توزیع گنو/لینوکس را انتخاب کنیم؟
۴۴.....	۱-۳-۳ توزیع چیست؟
۴۴.....	۲-۳-۳ علت تنوع توزیع ها چیست؟
۴۴.....	۳-۳-۳ انتخاب توزیع؟
۴۵.....	۴-۳ ویژگیهای سیستم عامل یونیکس (UNIX)
۴۶.....	۵-۳ معماری سیستم عامل لینوکس (Architecture OF Linux System)
۴۶.....	۶-۳ ساختار سیستم عامل یونیکس (UNIX)
۴۷.....	۷-۳ ساختار سیستم فایل (File System) در یونیکس (UNIX)
۴۸.....	۸-۳ مفهوم فایل سیستم در یونیکس (UNIX) و لینوکس (Linux)
۴۸.....	۹-۳ انواع فایلها در لینوکس
۴۹.....	۱۰-۳ معرفی دایرکتوری های سطح بالای سیستم فایل های رایج در لینوکس
۵۰.....	۱۱-۳ LILO و GRUB و فرایند راه اندازی لینوکس
۵۱.....	۱۲-۳ محیط X Windows چیست؟
۵۱.....	۱۳-۳ پروژه KDE و GNOME
۵۲.....	۱۴-۳ چرا ابونتو (ubuntu)
۵۴.....	۱۵-۳ ورود به سیستم
۵۵.....	۱۶-۳ آشنایی با بخشهای مختلف محیط های گرافیکی
۵۵.....	۱-۱۶-۳ محیط گرافیکی GNOME
۵۶.....	۲-۱۶-۳ محیط گرافیکی KDE
۵۷.....	۳-۱۶-۳ چند نکته
۵۸.....	۱۷-۳ استفاده از دستورات در خط فرمان
۵۸.....	۱-۱۷-۳ دو نوع کاربر (دو سطح دسترسی)
۵۸.....	۲-۱۷-۳ بررسی نشست ورود به سیستم
۵۹.....	۳-۱۷-۳ ترفندهای ویرایش دستورات خط فرمان CommandEditingTricks
۵۹.....	۴-۱۷-۳ دستور hostname و uname
۶۰.....	۵-۱۷-۳ دستور تاریخ
۶۰.....	۶-۱۷-۳ دستور cal

۶۰	۷-۱۷-۳ پوستر چیست؟
۶۱	۸-۱۷-۳ استفاده از چند دستور هم زمان
۶۱	۹-۱۷-۳ دستور clear و فراخوانی مجدد فرمان (history)
۶۱	۱۰-۱۷-۳ دستورات کمک (man, what is, Info, Help)
۶۱	۱۱-۱۷-۳ خروج از یونیکس (UNIX) و لینوکس (Linux)
۶۲	۱۲-۱۷-۳ تغییر رمز عبور
۶۲	۱۳-۱۷-۳ هسته لینوکس و دستور cat
۶۳	۱۴-۱۷-۳ دستور df, du و فرمت کردن فلاپی
۶۳	۱۵-۱۷-۳ دستورات واریسی فایل
۶۵	۱۶-۱۷-۳ تغییر مسیر
۶۵	۱۷-۱۷-۳ به کار بستن پیکربندیهای یونیکس (UNIX)
۶۵	۱۸-۱۷-۳ دستور pwd
۶۵	۱۹-۱۷-۳ دستور cd
۶۶	۲۰-۱۷-۳ به فرم دلخواه درآوردن خط فرمان
۶۷	۲۱-۱۷-۳ دستور ls
۶۸	۱۶-۳ کار کردن با فایل ها و فهرست ها
۶۹	۱۷-۳ تغییر دادن حالت دسترسی به یک فایل
۷۰	۱۸-۳ تنظیم اسامی مستعار
۷۰	۱۹-۳ دستورات پردازش فایل
۷۰	تغییر مسیر پیغامهای خطا
۷۰	حذف فایل
۷۰	دستور کپی cp
۷۱	حذف فهرستها
۷۱	دستور mv
۷۱	دستور od
۷۱	یافتن فایلها
۷۲	دستور wc
۷۲	دستور grep
۷۲	pipes
۷۳	۲۰-۳ حساب کاربری
۷۵	۱-۲۰-۳ دستور users, who و w
۷۵	تفاوت JCPU و PCPU
۶۵	۲۱-۳ مدیریت پردازش
۶۵	دستور ps
۷۶	۲۲-۳ کار با DVD, CD, Flopy, ...
۷۷	۲۳-۳ تنظیمات IP در لینوکس
۷۹	۲۴-۳ پیکربندی پوستر فرمان
۷۹	۲۵-۳ نوشتن برنامه در محیط لینوکس (Linux)
۸۲	۲۶-۳ سرانجام پنگوئن محبوب توروالدز چه خواهد شد؟
۸۳	۲۷-۳ دستور کار
۸۳	۲۸-۳ سوالات
۸۳	۲۹-۳ منابع
۶۹	تحقیق

بخش:

Setup سیستم



اعمال تغییرات

در آن

پیش آگاهی

پس از مطالعه این بخش:

- با CMOSE، BIOS و Setup آشنا خواهید شد.
- می توانید تنظیمات Setup را تغییر دهید.

برنامه Setup و تغییر پیکربندی سیستم جزء DOS یا Windows و یا سیستمهای عامل دیگر نیست، ولی به دلیل تأثیر فراوانی که این برنامه بر روی کارکرد کل سیستم و بخصوص سیستم عامل نصب شده در کامپیوتر دارد، از اینرو خوب است قبل از شروع آزمایشها با Setup سیستم آشنا شویم.

اما قبل از پرداختن به Setup سیستم و اعمال تغییرات در آن بهتر است با فرایند بوت، برنامه مدیر بوت، BIOS و CMOS آشنا شویم.

۱-۱ فرایند بوت و برنامه مدیر بوت:

اصطلاح فنی راهاندازی کامپیوتر را فرایند بوت یا Bootstrapping می گویند و یا بطور خلاصه تر بوت (boot یا booting). بخش آغازین این فرایند توسط کدهای ذخیره شده در ROM کامپیوتر انجام می شود. این بخش در همه سیستم عامل ها یکسان می باشد. وظیفه این کد بارگذاری مدیر بوت (Boot Loader) سیستم عامل برای ادامه فرایند می باشد. برنامه های مدیر بوت برنامه هایی هستند که توسط ROM کامپیوتر (یا همان BIOS) بارگذاری می شوند. برنامه مدیر بوت ابتدا مقداری اطلاعات پیکربندی را دریافت کرده و سپس هسته سیستم عامل را بارگذاری می کند و ادامه فرایند بوت را به آن می سپارد.

۱-۲ BIOS (Basic Input-Output System)

پردازنده مرکزی نمی تواند بطور اتوماتیک نحوه ارتباط با دستگاه های جانبی را تشخیص دهد، بلکه نیاز به نوعی برنامه نویسی پشتیبانی دارد که در حافظه بارگذاری میشود و اطلاعات لازم را در مورد یک وسیله به CPU منتقل می کند. این برنامه پشتیبان سرویس های ورودی/خروجی پایه (BIOS) نام دارد. برنامه هایی که وظیفه آنها اطلاع دادن نحوه ارتباط CPU با یک دستگاه جانبی میباشد سرویس یا راه انداز نامیده میشوند. در واقع، هر دستگاهی که روی کامپیوتر است به BIOS نیاز دارد.

یک برنامه نویس با استعداد میتواند در صورتی که کتاب کد صفحه کلید را داشته باشد، برای آن یک BIOS بنویسد. اکنون این سوال پیش می آید که این برنامه پشتیبان در کجا ذخیره شود؟ این برنامه باید با سیستم عامل تعامل داشته باشد. پس ذخیره سازی کد این برنامه درون سیستم عامل بسیار عالی خواهد بود. البته بسیاری از سیستم عامل ها از قبل دارای کد هایی هستند که نحوه ارتباط به صفحه کلید و ماوس و سایر دستگاه های جانبی را در بر دارند.

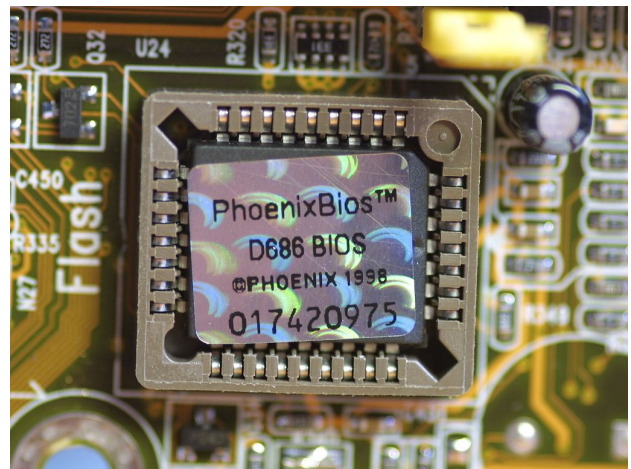
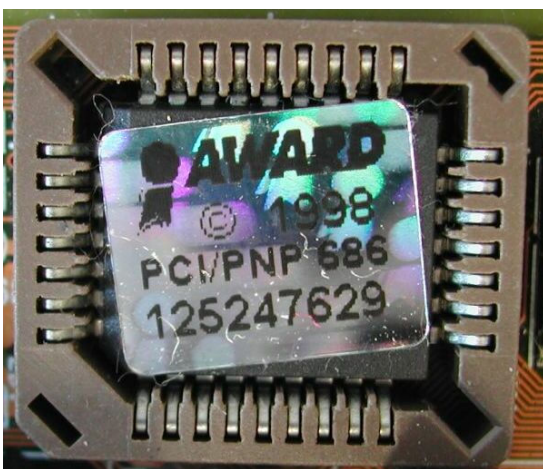
پس این حالت روی سیستم عامل در حال کار بسیار خوب جواب خواهد داد. ولی آیا به این موضوع اندیشیده اید که هنگام جمع کردن یک سیستم نو و وقتی هنوز سیستم عاملی وجود ندارد چه باید کرد؟

CPU باید برای راه اندازی سخت افزارهای مهم به BIOS سیستم دسترسی داشته باشد. نه تنها صفحه کلید، بلکه مانیتور، هارد درایو، درایوهای نوری، پورت های USB و رم. پس این کد نمیتواند درون هارد درایو یا دیسک های سی دی ذخیره شود. بسیاری از دستگاه ها را باید قبل از اینکه حتی هارد نصب شده باشد راه اندازی کرد.

بهترین وسیله برای ذخیره برنامه های پشتیبانی روی مادربرد است. بدین ترتیب یک مشکل حل میشود، ولی مشکل دیگری بوجود می آید: مادربرد از چه دستگاه ذخیره سازی استفاده میکند؟

DRAM که قابل استفاده نیست چون با قطع برق تمام داده ها پاک میشوند. باید نوعی محل ذخیره دائمی برای برنامه خود داشته باشید که نیازی به دیگر دستگاه های جانبی برای شروع کارش نداشته باشد. و این محل ذخیره سازی باید روی مادربرد سوار شود.

ROM: مادربردها برنامه پشتیبانی کنترلر صفحه کلید را به همراه سایر برنامه ها روی دستگاه خاصی بنام چیپ حافظه فقط خواندنی (ROM) ذخیره میکنند. یک چیپ ROM برنامه ها را دقیقاً مانند RAM ذخیره میکند که مانند یک صفحه گسترده ۸ بیتی است. ولی ROM دو تفاوت عمده با RAM دارد. اولاً، چیپ های ROM غیر قابل تغییر هستند، یعنی اطلاعات روی آنها با خاموش شدن کامپیوتر و قطع برق از بین نمیرود. دوماً، چیپ های ROM تجاری فقط خواندنی هستند، یعنی اینکه وقتی برنامه یک بار روی آنها ذخیره شد، دیگر قابل تعویض نیست. مادربردهای امروزی از نوع پیشرفته تری از ROM بنام Flash ROM استفاده میکنند که با ROM معمولی متفاوت است و میتوان محتویات آنرا توسط فرآیند بسیار خاصی بنام تخلیه بروز یا عوض کرد. وقتی CPU میخواهد با کنترلر صفحه کلید ارتباط برقرار کند، به چیپ ROM مراجعه میکند تا به برنامه پشتیبان مناسب دسترسی پیدا کند.



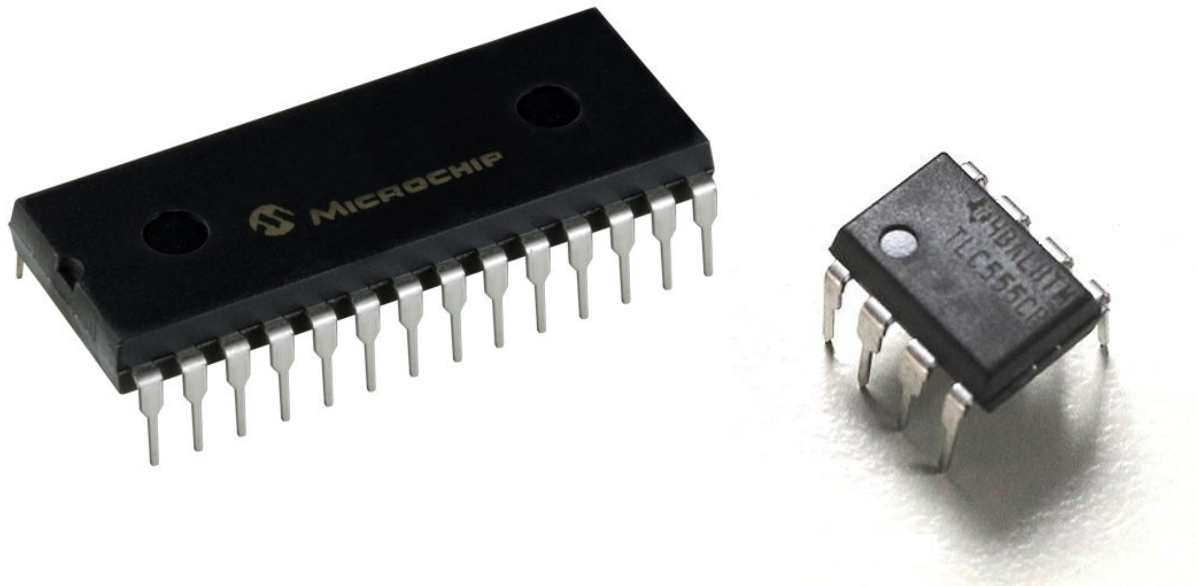
شکل ۱-۱: تصویر دو نمونه حافظه BIOS

همانطور که گفته شد برنامه بایوس بروی یک حافظه از نوع ROM نصب می شود (BIOS ROM) که به راحتی می توانید این حافظه را که ممکن است عبارت BIOS نیز بروی آن درج شده باشد، روی مادربرد کامپیوترتان پیدا کنید: با توجه به اینکه نرم افزار بایوس بر روی حافظه ROM قرار می گیرد امکان تغییر در این برنامه به وسیله کاربر وجود ندارد و کاربران کامپیوتر فقط قادر هستند برخی از تنظیمات قسمت های خاص این نرم افزار را که توسط شرکت سازنده از قبل پیش بینی شده، توسط منوی setup تغییر دهند. حال این سوال پیش می آید که این تغییراتی که ما در منوی Setup ایجاد می کنیم در کجا ثبت می شود با توجه به اینکه می دانیم نرم افزار BIOS بروی حافظه ROM قرار دارد و امکان تغییر آن وجود ندارد؟

۱-۳ CMOS (Complementary metal-Oxide Semiconductor)

در کامپیوترهای قدیمی XT، برای تعیین پیکربندی سیستم (نوع گردونه های دیسک نرم، دیسک سخت و کارت گرافیکی و ...) از یکسری اتصال دهنده (Jumper) استفاده می شد، ولی به دلیل مشکلات کار، در کامپیوترهای جدیدتر AT این تغییرات بروی یک حافظه مجزای دیگر که از نوع RAM است و به نام CMOS نامیده می شود، قرار می گیرد. یک چیپ حافظه جداگانه بنام نیمه رسانای اکسید فلزی مکمل (CMOS) اطلاعات پارامترهای خاص دستگاه های مختلف را نگهداری میکند که با قطع جریان برق، اطلاعات آن پاک می شود. CMOS برنامه ها را ذخیره نمیکند. فقط داده هایی را که توسط BIOS

خوانده میشود برای تکمیل برنامه ارتباط با سخت افزار قابل تعویض نگهداری میکند. CMOS بعنوان ساعت نیز کار میکند و تاریخ و زمان را نگه داری میکند.



شکل ۱-۲: دو نمونه CMOS

بسیاری از چیپ های CMOS حدود ۶۴ مگابایت داده را ذخیره سازی میکنند، ولی کامپیوتر فقط مقدار بسیار کمی از آن را نیاز دارد (حدود ۱۲۸ بیت) تا اطلاعات لازم را در مورد سخت افزار قابل تعویض نگهداری کند. گول اندازه کوچک داده ها را نخورید. اطلاعات درون CMOS برای کارکرد کامپیوتر حیاتی هستند.

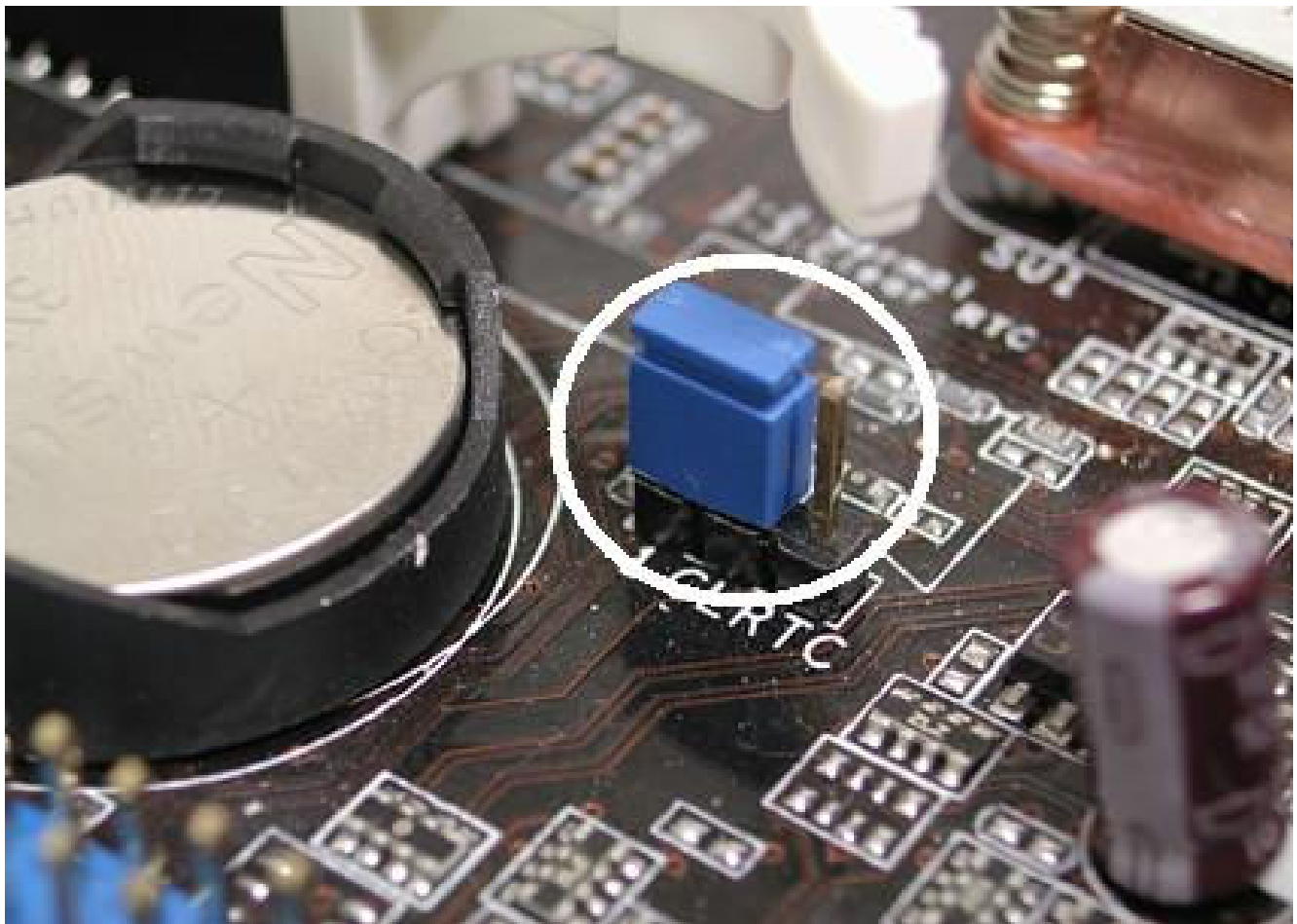
اگر داده های ذخیره شده روی CMOS در مورد قطعه خاصی از سخت افزار با مشخصات واقعی سخت افزار مطابقت نداشته باشد، کامپیوتر قادر به دسترسی به سخت افزار مورد نظر نخواهد بود. پس صحت این اطلاعات بسیار مهم است. حال فرض کنید به اشتباه تغییرات نادرستی را در بایوس ایجاد کردیم مثل تغییر در کلاک (CLOCK) و یا فرکانس عملکرد سی پی یو (CPU) و RAM که باعث شد کامپیوتر در هنگام بوت شدن هنگ کند و به هیچ عنوان سیستم بالا نیاید یا پسوردی بر روی کامپیوتر گذاشته اید که آن را فراموش نموده اید...

چون می دانیم همه این اطلاعات بروی حافظه CMOS قرار می گیرد برای پاک کردن این اطلاعات و بازگشت به تنظیمات اولیه بایوس، باید CMOS را ریست کرد که برای این کار سه راه وجود دارد:

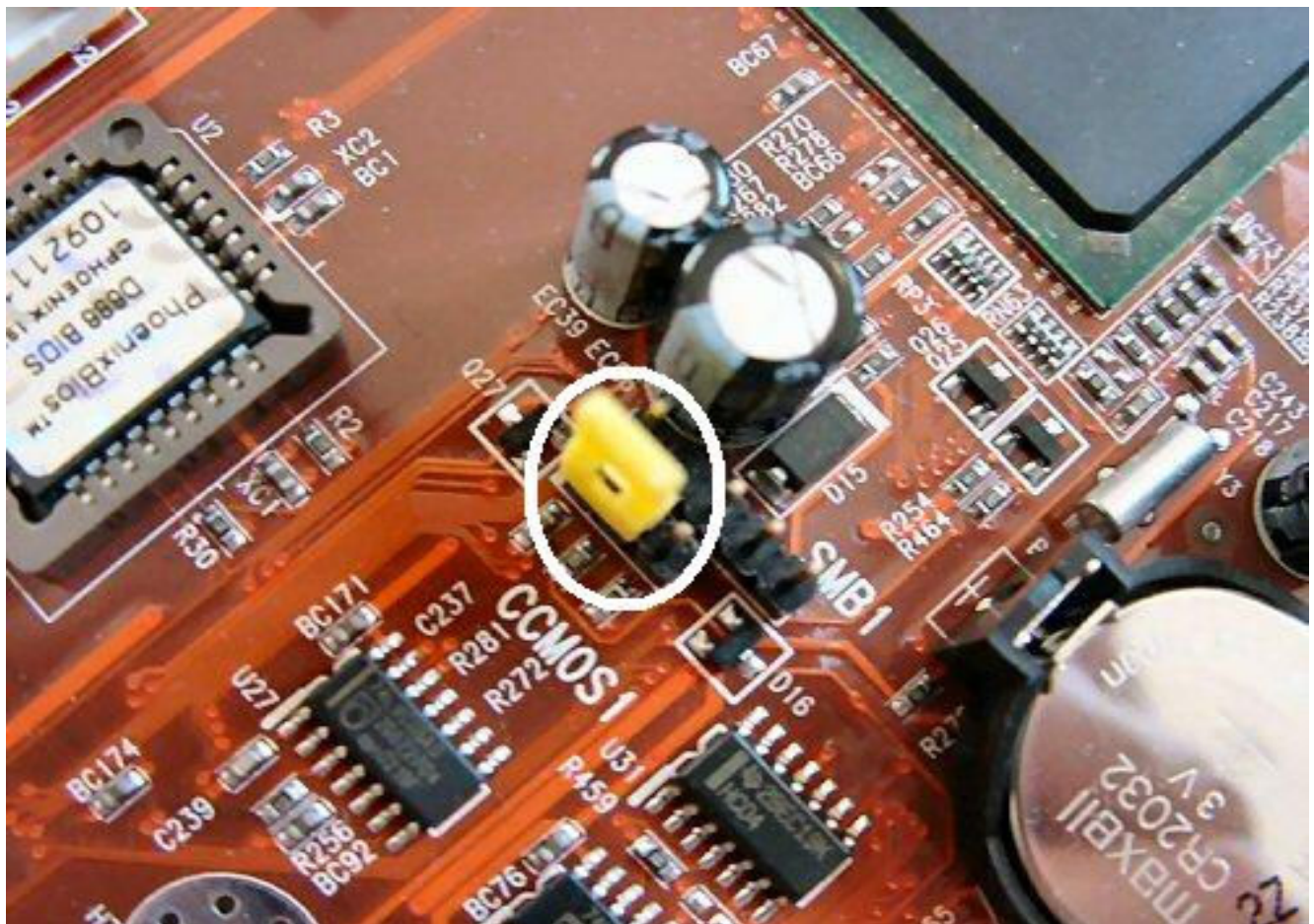
۱- در آوردن باتری CMOS

۲- تغییر جامپر CMOS که عموماً این جامپر در نزدیکی باتری CMOS قرار دارد و با نام CLRCMOS می باشد (در شکل با دایره مشخص شده)

۳- فشار دادن دکمه ریست CMOS که این دکمه بروی برخی از مادربردها وجود دارد در تصاویر زیر باتری CMOS را مشاهده کنید:



شکل ۱-۳: دو نمونه Motherboard و جامپر CMOS



اگر هر یک از سخت افزار های گفته شده را تعویض کنید، پس CMOS چگونه باید عوض شود؟

۱-۳-۱ بروز آوری CMOS (برنامه نصب)

هر کامپیوتر دارای برنامه ای درون رام سیستم، بنام برنامه نصب یا ابزار نصب سیستم است که به شما این امکان را میدهد تا به CMOS دسترسی پیدا کرده و آنرا بروز کنید. وقتی کامپیوتر خود را روشن میکنید، اولین چیزی که خواهید دید اطلاعات BIOS است. (پس از اینکه برنامه موجود در بایوس اجرا شد نخستین صفحه حاوی متنی ساده برروی مانیتور شما به نمایش در آمده و اطلاعات کمی در مورد نوع پردازنده و کارت گرافیک یا نوع هارد دیسک و سی دی رام متصل شده به سیستم به نمایش در می آید)

این اطلاعات ممکن است به یکی از شکل های زیر باشند:

 **American
Megatrends**
6UX7-4X P15

Released: 07/12/2000
AMIBIOS (C)1999 American Megatrends Inc.,



Check System Health OK,
CPU ID:0683 Patch ID:0010
Pentium III - 667 MHz
Checking NVRAM..
393216KB OK

WAIT...
Auto-Detecting Pri Master...IDE Hard Disk
Auto-Detecting Pri Slave...IDE Hard Disk
Auto-Detecting Sec Master...ATAPI CDROM
Auto-Detecting Sec Slave...Not Detected
Pri Master: 3.02 ST310212A
Ultra DMA Mode-4, S.M.A.R.T. Capable and Status OK
Pri Slave : 3.39 ST310211A
Ultra DMA Mode-4, S.M.A.R.T. Capable and Status OK
Sec Master: VYS7 CD05211

و یا

● Award Modular BIOS v6.00PG. An Energy Star Ally
Copyright (C) 1984-2003 Phoenix Technologies, LTD

Main Processor : AMD Athlon(tn) 64 Processor 3200+
Memory Testing : 1048576K OK
CPU0 Memory Information: DDR 400 CL:3 ,1T Dual Channel, 128-bit

IDE Channel 1 Master : WDC WD1200JB-75CRA0 16.06U16
IDE Channel 1 Slave : None
IDE Channel 2 Master : SONY CD-RW CRX175E2 S002
IDE Channel 2 Slave : TOSHIBA CD=DVDW SDR5372U TU11

IDE Channel 3 Master : None
IDE Channel 4 Master : None

Detecting IDE drives ...

Press DEL to enter SETUP, ESC to Enter Boot Menu
07/01/2005-MF-CK804-6A61FA1DC-10

در این هنگام برنامه بایوس در آخرین خط اجرای فرامین خود کنترل عملکرد را برای اجرای دستورات بعدی به نخستین رسانه دارای اطلاعات سیستمی (First Boot مثلاً فلاپی دیسک) میدهد. حتماً شما هم تاکنون به پیغام زیر هنگامی که یک فلاپی غیر سیستمی در فلاپی درایو قرار داده و کامپیوتر را بوت نموده اید برخورد کرده اید.

Non-system disk or disk error

Remove the disk and press any key to continue

معنی این پیغام این است که دیسکت شما فاقد اطلاعات لازم برای بوت کردن سیستم است. اگر شما این دیسکت را از دستگاه خارج کرده و یک کلید را بدخواه فشار دهید بایوس سیستم به دومین رسانه دارای فایل‌های سیستمی و بصورت پیش فرض به قسمت master boot record – MBR هارد دیسک مراجعه خواهد کرد.

در این مرحله کدهای اجرائی و سیستمی موجود یکی پس از دیگری اجرا میگردند و مثلاً در مورد سیستم عامل داس در نهایت به خط فرمان خواهیم رسید.

اما قبل از اجرای آخرین خط فرمان برنامه بایوس (Setup) یعنی هنگام بوت سیستم (راه اندازی) همیشه میتوان به برنامه نصب سیستم دسترسی داشت. سوال اساسی این است که چگونه میتوان هنگام بوت به برنامه نصب دسترسی پیدا کرد؟ Phoenix و Award ، AMI از کلیدهای مختلفی برای ورود به برنامه نصب سیستم استفاده میکنند. برای مثال، در پایین صفحه شکل قبل به شما گفته شده بود که Press DEL to Enter Setup. یعنی برای ورود به برنامه نصب سیستم باید کلید DEL را فشار دهید. بیاد داشته باشید که این فقط یکی از روشهای ممکن است. سازندگان مادربرد میتوانند از هر ترکیب کلیدی برای ورود به برنامه نصب استفاده کنند. میتوان کامپیوتر را طوری تنظیم کرد که پیغام نحوه ورود به برنامه نصب را نیز نمایش ندهد.

این پیغام و همچنین نحوه مشاهده و تغییر اطلاعات Setup و امکاناتی که Setup در اختیار کاربر میگذارد، به نوع Setup بستگی دارد.

اکثراً یکی از ترکیب کلیدهای زیر برنامه نصب سیستم را باز خواهد کرد

CTRL-ALT-Enter, ALT-Esc, F1, F2, DEL, Esc یا CTRL-S. با چندبار تلاش میتوانید ترکیب کلید صحیح را پیدا کنید. اگر موفق به انجام این کار نشدید، به دفترچه راهنمای مادربرد مراجعه کنید.

۱-۴ اعمال تغییرات در Setup سیستم

در اینجا قسمتهای مهمی که معمولاً در هر Setup وجود دارند، مورد بحث قرار میگیرند. این قسمتها شاید در سیستم شما موجود نبوده یا عنوان دیگری داشته باشند و یا اینکه عملکردی متفاوت با آنچه در اینجا شرح داده می شود، از خود نشان دهند.

در این بخش که فقط با کیبورد قابل تنظیم است میتوانید تنظیمات بسیار زیادی انجام دهید که به مهمترین آنها اشاره می شود. توجه کنید که منو بندی و گزینه ها ممکن است در مادربردهای مختلف متفاوت باشد اما اصول کلی و مفهوم تمام آنها یکی است که ما به گزینه های اصلی مربوط به دو سازنده معروف آیسى بایوس (AMI و Award) پرداخته ایم:

(1) Standard CMOS Features (قابلیتهای استاندارد)

این بخش شامل تنظیمات اصلی و مشترک بین تمام سیستمها میباشد و شامل موارد زیر است:

Date & Time:

تنظیم و نمایش تاریخ - تنظیم و نمایش ساعت

: IDE 0 Master

تنظیم و نمایش دستگاهی که به پورت IDE0 روی مادر برد نصب شده و در حالت Master قرار دارد

: IDE 0 Slave

تنظیم و نمایش دستگاهی که به پورت IDE0 روی مادر برد نصب شده و در حالت Slave قرار دارد

: IDE 1 Master

تنظیم و نمایش دستگاهی که به پورت IDE1 روی مادر برد نصب شده و در حالت Master قرار دارد

: IDE 1 Slave

تنظیم و نمایش دستگاهی که به پورت IDE1 روی مادر برد نصب شده و در حالت Slave قرار دارد

: Serial ATA 0

تنظیم و نمایش دستگاهی که به پورت SATA0 روی مادر برد نصب شده .

: Serial ATA 1

تنظیم و نمایش دستگاهی که به پورت SATA1 روی مادر برد نصب شده .

Floppy Device [Drive A / Drive B]

تنظیمات و فعال/غیرفعال کردن درایو فلاپی دیسک

: System Information

نمایش کلیاتی راجع به سیستم (در برخی سیستمها وجود دارد)

- نکته: دستگاههایی مانند سی دی رام ، رایتر و هارد اغلب **ATA(Advanced Technology Attachment)** هستند و با کابلی ۸۰ رشته ای موسوم به **IDE(Electronics Integrated Drive)** که به مادربرد وصل می شود . اغلب مادربردها دارای یک یا دو پورت IDE هستند(یعنی ۲ یا ۴ دستگاه ATA را میتوان به آن وصل کرد. به ATA در مواقعی **PATA(Parallel ATA)** نیز گویند).

ابتدا و انتهای این کابل ها از یک کانکتور خاص استفاده می گردد. در قسمت میانی کابل فوق از یک کانکتور دیگر نیز استفاده می گردد. مجموع طول کابل فوق نمی تواند از ۱۸ اینچ (۴۶ سانتیمتر) بیشتر باشد.

- کانکتور آبی برای اتصال به برد اصلی
- کانکتور مشکی برای اتصال به درایو اولیه (Master)
- کانکتور خاکستری برای اتصال به درایو ثانویه (Slave)

از آنجا که کابل IDE دارای سوکت(کانکتور) میباشد دو سوکت آن به دو دستگاه قابل نصب است که یکی Slave و دیگری Master خواهد بود . برای تنظیم اینکه کدام دستگاه Master و کدامیک Slave باشند در پشت این دستگاهها جامپرهایی تعبیه شده که با توجه به توضیحات روی دستگاه قابل تنظیم هستند .

هاردها و رایترها با سوکت **SATA(Serial Advanced Technology Attachment)** نیز موجودند که سرعت بسیار بیشتری دارند و کابل آنها کوچکتر میباشد . این دستگاهها با کابل موسوم به ساتا به مادربرد وصل می شوند . به هر پورت ساتا یک وسیله نصب میشود . در حال حاضر اغلب مادربردها دارای ۲ ، ۴ و یا ۶ پورت ساتا هستند .

۲) Advanced CMOS Features (امکانات ویژه و خاص)

شامل تنظیمات امکانات ویژه و خاص مربوط به BIOS سیستم می باشد .

مثلا در اغلب بایوس ها شامل موارد زیر است :

(توضیح: گزینه های قابل انتخاب برای هر تنظیم در داخل گروه [] قرار داده شده)

Boot Sector Protection : [Enable/ Disable]

فعال (Enable) یا غیرفعال (Disable) بودن سیستم حفاظت از خرابی و دستکاری غیرمجاز بایوس توسط ویروس و یا اتفاقات ناخواسته

Full Screen Logo Screen : [Enable/ Disable]

نمایش یا عدم نمایش لوگوی شرکت سازنده مادربرد در ابتدای روشن شدن سیستم

Quick Booting : [Enable/ Disable]

با قرار دادن این گزینه در حالت فعال (Enable) ، سیستم پس از گذشت ۱۰ ثانیه از چک کردن تنظیمات شروع به بالا آمدن میکند . (اگر چک قعات بیش از ۱۰ ثانیه طول بکشد از چک کردن مابقی صرفنظر می کند)

Boot Up Num-Lock LED : [ON / OFF]

مشخص کردن وضعیت پیش فرض چراغ دکمه های شماره (Numbers Keys)

Boot Sequence [C, CD-Rom,A / A,CD-Rom,C / CD-Rom.C,A / CD- Rom,A,C]

با ورود به این بخش می توانید اولویت مراجعه سیستم برای بالا آمدن را به دستگاه دلخواه بدهید .

Hard Disk Boot Priority : [HDD 0 / HDD 1 / HDD 2/ ...]

تنظیم اولویت بین هارددیسکها برای بالا آمدن (اینکه سیستم عامل با کمک کدام هارد بالا بیاید)

First(1st) Boot Device : [Hard Disk / CD-ROM / Floppy / USB HDD/...]

تنظیم اولین دستگاهی که توسط سیستم برای بوت شدن مورد استفاده قرار میگیرد

Second(2nd) Boot Device : [Hard Disk / CD-ROM / Floppy / USB HDD/...]

تنظیم دومین دستگاهی که توسط سیستم برای بوت شدن مورد استفاده قرار میگیرد . (به دستگاه دوم تنها زمانی مراجعه می شود که دستگاه اول نتوانسته سیستم را بوت کند)

Third(3rd) Boot Device : [Hard Disk / CD-ROM / Floppy / USB HDD/...]

تنظیم سومین دستگاهی که توسط سیستم برای بوت شدن مورد استفاده قرار میگیرد . (به دستگاه سوم تنها زمانی مراجعه می شود که دستگاه دوم نتوانسته سیستم را بوت کند)

Other Boot Device

تنظیم دیگر دستگاهها که توسط سیستم برای بوت شدن مورد استفاده قرار میگیرند.

– نکته : هنگامی که میخواهیم از روی سی دی ویندوز نصب کنیم بایستی First Boot Device را CD-ROM قرار دهیم تا سیستم به محض روشن شدن به درایو سی دی مراجعه کرده و از روی سی دی بالا بیاید .

برای نصب سیستم عامل داس بایستی First Boot Device را Floppy Disc قرار دهیم تا سیستم به محض روشن شدن به درایو فلاپی مراجعه کرده و از روی فلاپی بالا بیاید .

۳) Integrated Peripherals (دستگاههای مجتمع روی مادربرد)

تنظیمات مربوط به تمامی دستگاهها و قطعات روی مادربرد (مجتمع شده بر روی مادربرد) یا اصطلاحاً تنظیمات دستگاههای آنبرد (On Board) دستگاههای آنبرد مانند سیستم کارت صوت مادربرد و یا پورتهای USB (یو اس بی) و همچنین در بعضی از مادربردها تنظیمات کارت گرافیکهای تعبیه شده روی مادربرد و نیز تنظیمات مربوط به کارت شبکه تعبیه شده بر روی مادربرد .

به توضیح بخشهای مهم این قسمت میپردازیم :

USB Controller : [Disable / 2 Ports / 4 Ports / 6 Ports / 8 Ports/...]

تنظیم فعال یا غیر فعال بودن پورتهای USB مادربرد

USB Device Legacy Support : [Enable / Disable]

تنظیم امکان/عدم امکان استفاده از دستگاههای USB در سیستم عامل

Onboard Lan Controller : [Enable / Disable]

تنظیم فعال/غیرفعال بودن کنترلر شبکه

Audio Controller : [Enable / Disable]

تنظیم فعال/غیرفعال بودن کارت صوت مجتمع شده

On-Chip ATA Controller : [Enable / Disable]

تنظیم کنترلر دستگاههای اِتا (ATA)

On-Chip SATA Controller : [Enable / Disable]

تنظیم فعال/غیرفعال بودن کنترلر دستگاههای SATA

Raid Sata Mode : [IDE / Raid0 / Raid 1 / Raid 2]

تنظیم سیستم استفاده از Raid (استفاده بهینه و بدون خطا از چند هارددیسک)

: I/O Device

تنظیمات مربوط به پورتهای خروجی و ورودی مادربرد مانند LPT, Com, Mid

USB Keyboard Support : [Enable / Disable]

فعال یا غیرفعال بودن کیبورد با پورت USB (برای کیبورد های یو اس بی حتما باید این گزینه فعال باشد)

USB Mouse Support : [Enable / Disable]

فعال یا غیرفعال بودن ماوس با پورت USB (برای موس های یو اس بی حتما باید این پورت فعال باشد ، در غیر اینصورت هنگام نصب ویندوز در زمانی که سیستم یو اس بی را شناسایی نمی کند موس غیر فعال می شود)

Onboard Serial Port 1&2

تنظیمات مربوط به پورتهای سریال

۴) Power Management Setup (تنظیمات مدیریت تغذیه سیستم)

در این قسمت اغلب چند بخش اساسی وجود دارد :

ACPI(Advanced Configuration and Power Interface) Function(Or Suspend Mode) : [Enable / Disable]

تنظیم عملکرد حالت ACPI که در ویندوز ۲۰۰۰ و اکس پی قابل استفاده است)

USB Device Wake UP : [Enable / Disable]

اینکه سیستم با دستگاههای متصل به USB از حالت استندبای خارج گردد یا خیر

Resume From S3 By PS2 Keyboard: [Enable / Disable]

اینکه سیستم با کیبورد متصل به PS2 از حالت استندبای خارج گردد یا خیر

Resume From S3 By PS2 Mouse : [Enable / Disable]

اینکه سیستم با ماوس متصل به PS2 از حالت استندبای خارج گردد یا خیر

Power Button Function : [Power Off / Suspend]

عملکرد دکمه پاور روی کیس (خاموش کند و یا به حالت استندبای برود)

: Keyboard Power ON

این بخش دارای ۳ گزینه است :

- Password : با زدن پسووردی دلخواه سیستم روشن شود

- Disable : حالت روشن شدن سیستم با صفحه کلید غیرفعال باشد

- Keyboard 98 : با زدن دکمه پاور روی کیبوردهای قدیمی سیستم روشن شود .

Mouse Power ON : [Enable / Disable]

اینکه وقتی کلیک دکمه چپ ماوس کلیک گردد سیستم روشن شود .

Modem Ring resume : [Enable / Disable]

این گزینه اگر فعال باشد ، اگر مودم شما به خط تلفن متصل باشد و کسی با شما تماس بگیرد سیستم روشن می شود

۵) **Pnp/PCI Configuration** (پیکربندی و تنظیمات مربوط به دستگاههای**PNP(Plug & Play) و PCI (Peripheral Component Interconnect)**

دستگاههای PCI دستگاههایی هستند که از طریق اسلات (شکاف) های PCI به مادربورد متصل می شوند . مانند مودم ، تی وی کیچر و یا کارت صوت

در این بخش هرکدام از شکافهای (Slot) مادربورد را میتوان تنظیم کرد که از کدام شماره وقفه (IRQ) استفاده کند . بهتر است گزینه Auto را انتخاب کنیم تا سیستم به طور اتوماتیک شماره وقفه به دستگاهها نسبت دهد .

۶) **PC Health Status یا H/W Monitor** (وضعیت سلامتی سیستم)

این بخش نمایانگر وضعیت کلی سیستم شامل سرعت فنها ، دمای جاری سیستم و پردازنده ، ولتاژ پردازنده و ... میباشد . در بعضی از سیستمها در این بخش گزینه های زیر نیز وجود دارند :

Chassis Instruction : [Enable / Disable]

عملکردهای جانبی کیس مانند باز بودن درب

CPU Smart Fan Target : [Disable / Level 1/Level2/...]

تنظیم اتوماتیک سرعت فن پردازنده در دمای زیاد

CPU Fan Fail Warning : [Enable / Disable]

هشدار در هنگام خرابی یا عملکرد بد فن پردازنده

CPU Temp Warning : [Disable / 50 / 60 / 70 / 80]

هشدار رسیدن دمای پردازنده به دمای خاص

CPU Temp Warning : [Enable / Disable]

هشدار در هنگام خرابی یا عملکرد بد فن سیستم (کیس)

۷) **Frequency/Voltage Control** (کنترل ولتاژ و فرکانس)

در این بخش می توان فرکانس کاری پردازنده و ولتاژ کاری رم و پردازنده و همچنین ولتاژ کارت گرافیکی متصل به شکاف AGP را تنظیم کرد .

این بخش بیشتر برای OverClock کردن بکار میرود .

دستکاری این بخش بدون داشتن اطلاعات دقیق و کافی ممکن است باعث سوختن مادربرد ، پردازنده و یا کارت گرافیک گردد .

۸) **Default Top Performance / Loaf Fail-Safe / Optimized** (تنظیمات اولیه و کارا)

این بخش که با سه نام ذکر شده بالا در دسترس است شامل تنظیمات پیش فرض و اولیه کارخانه سازنده و همچنین پیش تنظیمات پیشنهادی شرکت سازنده میباشد .

به برخی از گزینه های رایج آن میپردازیم :

: Load Fail-Safe Default

بارگزاری تنظیمات پیش فرضی که از برخی از قابلیت های مادربرد چشمپوشی می کند .

: Optimized Default & Top Performance

بارگزاری تنظیمات پیش فرضی که بیشترین کارایی مادربرد سیستم را به همراه دارد.

۹) **User Password & BIOS Setting Password - Set Supervisor** (کلمه عبور تنظیمات

بایوس / تنظیم کلمه عبور کاربر)

در این بخش میتوان برای ورود به سیستم و یا ورود به تنظیمات بایوس کلمه عبور گذاشت .

با تنظیم System Password یا User Password می توان برای بالا آمدن سیستم یک کلمه عبور گذاشت تا کسی نتواند بدون مجوز از کامپیوتر استفاده کند .

با تنظیم Password Supervisor می توان برای ورود به بخش تنظیمات بایوس کلمه عبور قرار داد .

– نکته : با برداشتن باتری روی مادربرد این دو کلمه عبور پاک خواهند شد و کلیه تنظیمات به حالت اولیه (تنظیمات کارخانه) برمیگردد .

۱۰) **Save And Exit** (ذخیره و خروج از بایوس)

در بیشتر موارد با زدن کلید F10 تغییرات اعمال شده در بایوس ذخیره می شود و سیستم ری استارت و آماده کار خواهد شد . در بعضی از مادربردها هم بایستی گزینه Save And Exit را برای ذخیره و خروج انتخاب کرد .

گزینه Saving Exit Without نیز موجب خروج از بخش تنظیمات البته بدون ذخیره تغییرات اعمال شده میگردد .

۱-۵ سوالات:

۱. تفاوت CMOS و BIOS را بیان کنید.
۲. Setup سیستم چیست؟
۳. چگونه می توان پسورد BIOS را حذف کرد؟ (۳ روش سخت افزاری و ۲ روش نرم افزاری)
۴. آیا بدون استفاده از برنامه Setup، امکان تغییر محتوای حافظه CMOS وجود دارد؟
۵. به چه روشی می توان دیسک سخت را به سیستم معرفی کرد؟
۶. برنامه زیر را در نظر بگیرید:

```
MOV AL,10H
OUT 70H,AL
IN AL,71H
INT 20H
```

عملکرد هر سطر را بیان کنید.

۱-۶ دستورکار

تذکر: اگر گزینه ای در Setup سیستم شما وجود ندارد از آزمایش آن مورد صرف نظر کنید.

۱. برای ورود به برنامه Setup در سیستم شما چه کار باید کرد؟
 ۲. برنامه Setup سیستم شما توسط چه شرکتی نوشته شده است؟
 ۳. مشخصات دیسک سخت را یادداشت کنید. آیا می توانید این مشخصات را تغییر دهید؟ اگر می توانید، چه گزینه هایی را خود برنامه Setup پیشنهاد می کند؟
 ۴. مشخصات حافظه اصلی (RAM) سیستم تان را یادداشت کنید (نوع و حجم)؟
 ۵. با تغییراتی که در Setup می دهید کاری کنید که موقع راه اندازی سیستم، اول دیسک سخت برای وجود یا عدم وجود سیستم عامل بررسی شود. تغییری را که در Setup بوجود آورده اید، بنویسید.
 ۶. برای سیستم خود و برنامه Setup رمز عبور بگذارید تا افراد دیگر نتوانند وارد برنامه Setup شوند، ولی با رمز عبور بتوانند وارد سیستم شوند. اگر چنین امکانی در Setup سیستم شما وجود ندارد، این گام را انجام ندهید. توجه: پس از انجام این گام، حتماً رمز عبور را از روی سیستم بردارید.
- تذکر: پس از انجام آزمایش، Setup را به حالت اولیه آن برگردانید.

۱-۷ پروژه برنامه نویسی

برنامه ای بنویسید که پسورد Setup را حذف کند.

۱-۸ منابع

<http://tafazoli.iut.ac.ir>
<http://www.forum.microrayaneh.com>
<http://dialup.persianblog.ir>
<http://certification.mihanblog.ir>
<http://arteah.parsblog.com>

بخش ۲:

کار با



در ویندوز

پیش آگاهی

پس از مطالعه این بخش:

- با فایل ثبت داده ای در ویندوز (Registry) آشنا خواهید شد.
- قادر خواهید بود در بانک رجیستری تغییرات اعمال کنید.
- در محیط C# جهت تغییرات در رجیستری، برنامه نویسی کنید.

در ابتدا کامپیوترها و کاربران بر اساس سیستم عامل Ms-Dos فعالیت می کردند و برنامه هایی که ساخته می شد بر اساس این سیستم عامل بودند . برنامه تحت Dos برنامه هایی بودند که کاربر از راهی مشخص وارد برنامه می شد .مراحل را طی می کرد و راه خروج نیز مشخص بود و کاربر همیشه مجبور به طی کردن مسیرهای مشخص بود و برای رسیدن به منو و پنجره ای ، حتماً باید از منو و پنجره قبلی آن عبور می کرد و هیچگونه اختیاری در تنظیم منو ، پنجره ها و یا تغییر آنها نداشت (که احتیاج باشد در جایی ثبت شد). این خاصیت سیستم عامل Dos ، باعث می شد که اکثر تنظیمات این برنامه های کاربردی در مرحله برنامه نویسی و Compile کردن انجام شود و احتیاجی جهت ثبت تنظیمات در فایل نباشد.

۲-۱ رجیستری چیست؟

Registry در فرهنگ کامپیوتر به فایل یا فایللهایی اطلاق میشود که عهده دار تنظیمات ویندوز می باشند . در سیستم عامل Dos سه فایل به نامهای Config.sys ، Msdos.sys و Autoexec.bat وجود داشتند که سیستم عامل Dos و جمعاً برنامه های کاربردی تنظیماتی مانند اختصاص دادن Buffer و یا تعریف Cdrom و غیره را در این فایل ها ثبت می کردند و در واقع این سه فایل نقش بانک داده ای ثبت و یا همان Registry را ایفا می کردند.

بعد از سیستم Ms-dos ، سیستم عامل Windows 3.x به بازار عرضه شد که تحولی در سیستم عامل و نحوه ساخت برنامه های کاربردی ایجاد کرد . سیستم عامل Windows 3.x دارای پنجره هایی بود که کاربر می توانست در نحوه ظاهر شدن ، رنگ ، اندازه پنجره ها و نحوه اجرا شدن برنامه ها دخالت کرده (امکانی که در Dos وجود نداشت) و آنها را به دلخواه تنظیم کند . چیزی که در این جا مسلم است ، این است که تنظیمات سیستم عامل و یا تنظیماتی که توسط کاربر اعمال می شد باید در فایلی ذخیره می شد تا هر بار که سیستم عامل راه اندازی می شود ، احتیاجی به تنظیمات مجدد نباشد ، از طرفی به علت آنکه تنوع تنظیماتی که توسط کاربر و یا برنامه های کاربردی ایجاد می شد ، بسیار متعدد بود و ممکن بود که کاربران بخواهند دائماً این تنظیمات را تغییر دهند و از طرفی دیگر اگر سازندگان این سیستم عامل می خواستند که از همان سه فایل Dos برای ثبت این تنظیمات استفاده کنند ، ممکن بود این سه فایل که فایل های حیاتی و Boot کننده سیستم نیز بودند ، دچار اشکال شده و آسیب ببینند . به همین منظور این بار و در این سیستم عامل فایل هایی با پسوند ini. که مهمترین آنها Win.ini ، System.ini و Control.ini بودند ، وظیفه ثبت تنظیمات و یا Registry را برعهده گرفتند.

و در نهایت نسل سوم رجیستری از زمانی آغاز شد که سیستم عامل ویندوز سری ۹X که معروفترین و پراکارتین آنها ۹۵ و ۹۸ بودند به بازار عرضه شد . این ویندوز به واسطه ساختار آن می تواند سخت افزارهای مختلفی را پشتیبانی کند و برنامه های کاربردی زیادی تحت این ویندوز ساخته شد و به واسطه این رابط گرافیکی کاربران می خواهند تنظیماتی را بر روی پنجره ها ، فونتها ، سخت افزارها و برنامه ها به وجود آورند . در این میان آنچه که مسلم است این است که این تنظیمات احتیاج به ثبت در بانک داده ای ثبت (Registry) دارند و از طرفی فایل های Control.ini ، Win.ini و System.ini از نوع فایل های

متنی بودند و به واسطه این تنظیمات حجم این فایل ها بسیار زیاد می شد. و Load شدن این فایل های متنی در هنگام شروع ویندوز و اعمال تنظیمات ذخیره شده در آنها، زمان زیادی را به خود اختصاص می دادند و از طرفی آسیب پذیر می شدند، به همین دلیل سازندگان ویندوز این تنظیمات را بر دو فایل System.dat و User.dat بنا نهادند که این فایل ها بر اساس اعداد باینری و Hex ساخته می شدند، هم امنیت بیشتری داشتند و هم حجم کمتری اشغال می کردند.

در واقع رجیستری را می توان به عنوان بانک اطلاعاتی معرفی کرد که برای نگهداری تنظیمات و گزینه های نسخه های ۳۲ بیتی ویندوز همچون ویندوز ME، ۹۵، ۹۸ و NT/۲۰۰۰ مورد استفاده قرار می گیرد. و شامل اطلاعات و تنظیماتی است که برای تمام قطعات سخت افزاری، نرم افزاری، کاربران و تنظیمات مربوط به خود سیستم مورد استفاده قرار می گیرد.

هر زمان که کاربری تغییری در تنظیمات مربوط به Control Panel یا System Policies، File Associations ایجاد کند و یا برنامه نرم افزاری جدیدی را نصب کند، این تغییرات در رجیستری ویندوز ثبت شده و باقی می ماند.

۲-۲ برخی از مزایای پایگاه داده رجیستری :

فایلهای رجیستری دارای خاصیتهای فقط خواندنی و پنهان و سیستمی هستند، بنابراین احتمال آنکه بطور تصادفی توسط یک کاربر غیر حرفه ای حذف گردند از بین رفته است. (احتمالی که در مورد فایلهای *.INI وجود داشت) رجیستری نه تنها مکانی برای ذخیره پارامترهای سخت افزاری و سیستم عامل است بلکه برنامه های کاربردی نیز میتوانند به جای استفاده از فایلهای *.INI جداگانه برای ذخیره کردن پارامترهای خود، آزادانه از رجیستری استفاده کنند. ابزارهای متنوعی جهت مدیریت و کار با رجیستری در ویندوز وجود دارد. در رجیستری ابزارهایی وجود دارند که شما را قادر میسازند هر پارامتری را یافته و اطلاعات آنرا درخواست کرده و یا تغییر دهید. سیستم عامل ویندوز به هنگام نصب (Setup) خود پایگاه داده رجیستری را ایجاد میکند که شامل دو فایل User.dat و System.dat می باشد.

۲-۳ محل فایل های رجیستری

در ویندوز 98.95 رجیستری در داخل دو فایل مخفی واقع در دایرکتوری ویندوز، بنام های User.dat و System.dat گنجانده شده اند

در ویندوز Me، علاوه بر آن دو فایل بالا، یک فایل دیگر نیز بنام Classes.dat وجود دارد که برای ذخیره اطلاعات رجیستری استفاده می شود.

در ویندوز NT، 2000 و Xp، فایلها بطور جداگانه در داخل دایرکتوری با مسیر زیر قرار دارد:

%SystemRoot%\windows\System32\Config

۲-۴ ویرایش اطلاعات موجود در رجیستری

همانطور که گفته شد رجیستری ویندوز پایگاه داده ای است که در آن مجموعه اطلاعات پیکره بندی سیستم و احیاناً اطلاعات شخصی کاربر را نگهداری میکند و گفته شد که این پایگاه داده بصورت فایلهای متنی ساده نیست و دارای ساختاری پیچیده است. اما بطور مستقیم نمی توان به فایل های رجیستری دسترسی پیدا کرد. بلکه باید از یک ابزار شناخته شده بنام «Registry Editor» جهت ویرایش و ایجاد تغییرات در رجیستری استفاده کنید.

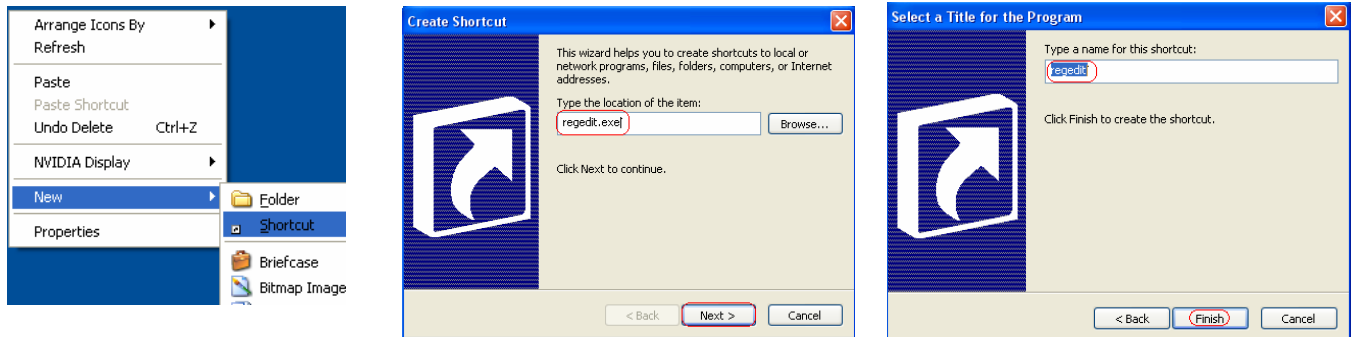
نحوه اجرای این برنامه که در خود سیستم عامل ویندوز قرار دارد به یکی از دو صورت زیر است :

از منوی Start گزینه Run را اجرا میکنیم و سپس نام Regedit را تایپ کرده و Enter میکنیم.

از پنجره Windows Explorer به پوشه Windows رفته و فایل Regedit.exe را اجرا میکنیم

۲-۵ ایجاد Shortcut برای برنامه Regedit

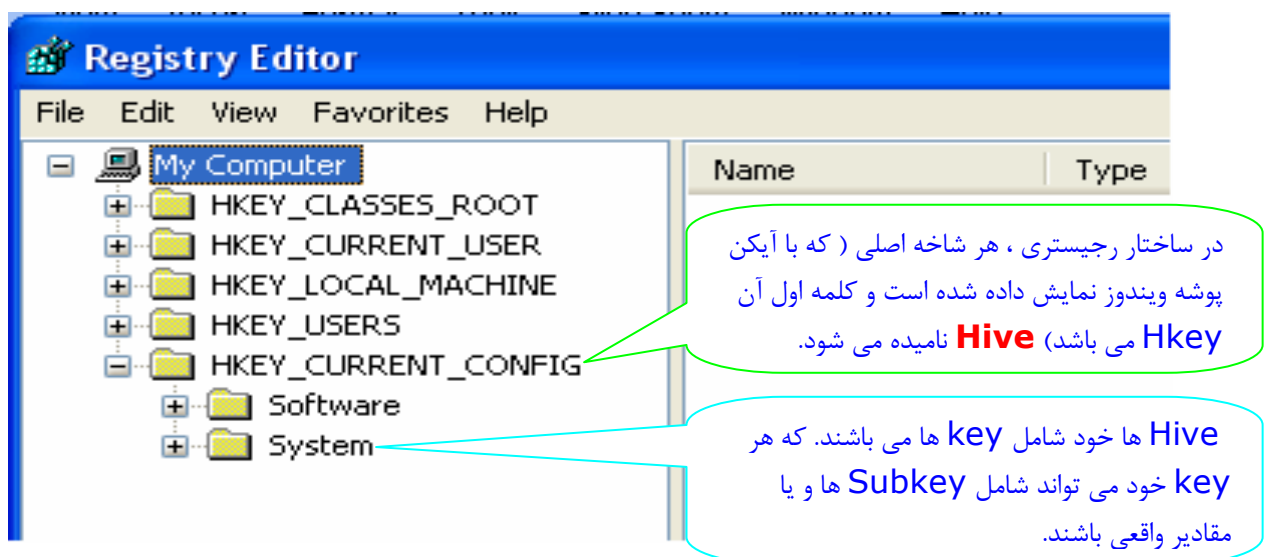
برای ایجاد ایکن میانبر برای Regedit.exe ، در قسمت خالی دسک تاپ راست کلیک کرده و گزینه New و Shortcut را انتخاب می کنیم. در خط فرمان عبارت «Regedit.exe» را تایپ نموده و بعد از زدن Next نام دلخواهی را تایپ می کنید با زدن دکمه Finish یک ایکن میانبر روی دیسک تاپ ایجاد خواهد شد.



شکل ۲-۱: ایجاد Shortcut برای برنامه Regedit

۲-۶ ساختار و اجزای رجیستری در برنامه Regedit

رجیستری دارای یک ساختار سلسله مراتبی می باشد، که شکل ظاهری آن مشابه Windows Explorer بوده و در آن ، ساختمان رجیستری بصورت یک ساختار درختی (سلسله مراتبی) نمایش داده می شود.



شکل ۲-۲: ساختار درختی و اجزاء پنجره Registry Editor

۲-۶-۱ اجزای تشکیل دهنده رجیستری:

الف) Subtree (Hive یا HotKey)

رجیستری از ۵ فیلد اصلی (Hive) تشکیل شده است که به این ۵ فیلد کلید (Key) میگویند. از آنجا که این کلیدهای ویژه نوعی کلید فوری (HotKey) جهت دسترسی به مقادیر ذخیره شده در رجیستری میباشند در رجیستری آنها را با واژه HKEY در ابتدای نام هر کلید نام گذاری کرده اند .

این ۵ کلید در اصطلاح کلیدهای ریشه یا اصلی (Root Key) در رجیستری میگویند. لازم بذکر است تمام مقادیر پیکره بندی و تعاریف سیستم و همچنین تعاریف کاربر بصورت ساختمان داده ای سلسله مراتبی در این ۵ کلید ریشه قرار دارند که جهت تغییر و یا تصحیح هر تعریف باید به کلید مورد نظر و همچنین زیر کلیدهای آن مراجعه نمود.

در ادامه به ذکر نام و ویژگیهای هر کدام از کلیدهای ریشه می پردازیم :

۱- HKEY_CLASSES_ROOT:

این کلید حاوی همان اطلاعات موجود در رجیستری ویندوز ۳/۱ میباشد. که این اطلاعات عبارتند از: پسوندهای فایل ها و اینکه هر پسوندی چه معنی دارد و نیز اطلاعات OLE و DDE و علاوه بر اینها در این کلید چندین کلید فرعی برای میانبرها (Shortcut) و دیگر امکانات مربوط به صفحه نمایش وجود دارد.

۲- HKEY_CURRENT_USER:

در این کلید ریشه اطلاعات کاربری که هم اکنون به ویندوز لاگین کرده است وجود خواهد داشت. در اصل حاوی اطلاعات مربوط به پروفایل شخصی کاربر فعال می باشد و یا به عبارتی در این کلید پارامترهای تعیین شده در فایل User.dat مربوط به پروفایل شخص فعال وجود دارد. در این کلید پارامترهایی که شما در پنجره Control Panel تعیین کرده اید پارامترهای مربوط به کار با برنامه های کاربردی و ... وجود دارد. چند مورد از زیر کلیدهای موجود در این کلید عبارتند از :

AppEvent: این کلید مشخص کننده فایلهای صوتی که در صورت وقوع یک واقعه خاص مثل باز شدن یک پنجره باید اجرا شوند را تعیین میکند.

Control Panel: این کلید حاوی پارامترهایی است که در برخی قسمتهای کنترل پانل تعیین گردیده است.

Keyboard Layout: در این کلید آنچه که شما در تنظیمات صفحه کلید منظور نموده اید قرار دارد.

Software: این کلید حاوی پارامترهایی است که شما در حین نصب انواع نرم افزارها تنظیم نموده اید.

۳- HKEY_LOCAL_MACHINE:

در این کلید محتویات فایل SYSTEM.DAT ذخیره میشود که عبارتند از پارامترهای مربوط به سخت افزارها و نرم افزارهای موجود در کامپیوتر شما. کلیدهای فرعی این کلید عبارتند از :

HARDWARE: در این کلید چندین کلید فرعی مربوط به پورتهای و انواع مودمهای مورد استفاده در سیستم شما میباشد.

SOFTWARE: در این کلید پارامترهای مربوط به نرم افزارهای نصب شده در کامپیوتر شما که سیستم از آنها استفاده میکنند ذخیره میشود.

SYSTEM: در این کلید کلیدهای فرعی و پارامترهای مربوط به شروع به کار ویندوز ذخیره میشود.

۴- HKEY_USERS:

در این کلید کلیه اطلاعات و یا ارجاعات و تنظیمات پروفایلهای کاربران مختلف که در این سیستم عامل یک پروفایل (حساب کاربری) و تنظیمات مربوط به آن دارند موجود میباشد.

۵- HKEY_CURRENT_CONFIG:

این کلید نیز حاوی کلیه اطلاعات و تنظیمات سخت افزارهای نصب شده کاربر فعال میباشد. بدیهی است هر کاربر دارای یک حساب کاربری معتبر (پروفایل) و اطلاعات مربوط به خود میباشد.

ب) Key

مانند Folder ها و Subfolder های روی هارد هستند. هر Key متناظر با Object های سخت افزاری یا نرم افزاری می باشد. Subkey ها، KEY هایی هستند که درون یکسرس Key قرار دارند.

ج) Entity

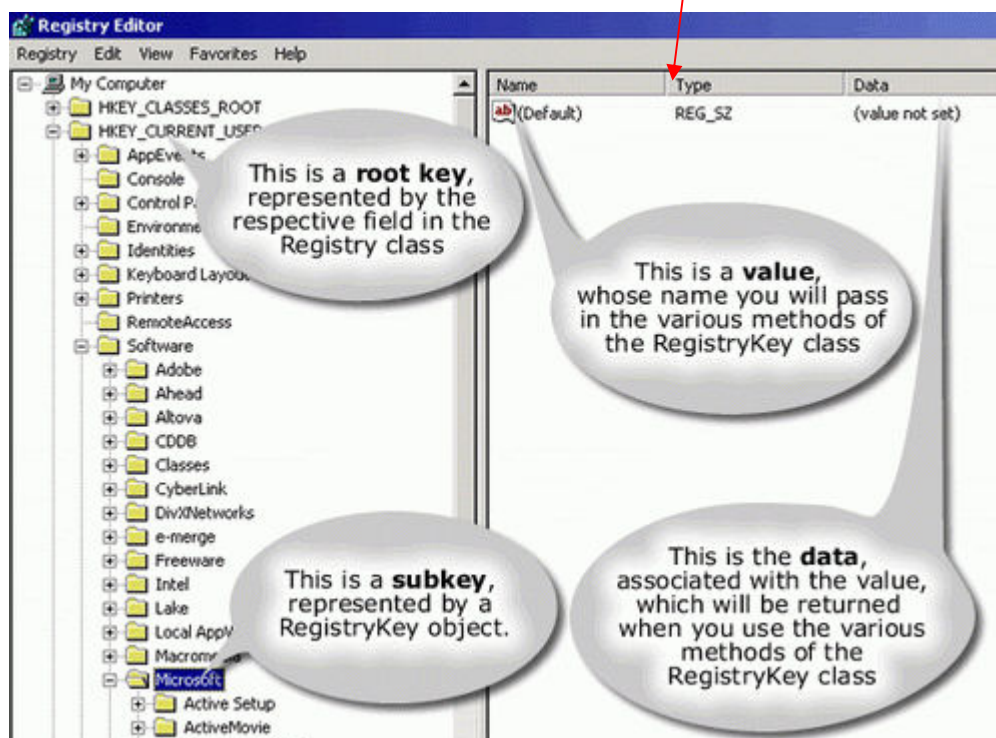
هر Key دارای یک یا چند Entity است.

هر Entity دارای ۳ بخش می باشد:

۱- Name (نام)

۲- Data Type (نوع داده)

۳- Value (مقدار)



شکل ۲-۳: پنجره Registry Editor و اجزاء و داده های آن

همانطور که در شکل ۱-۲ و ۲-۲ مشاهده میکنید این برنامه به دو قسمت در چپ و راست تقسیم میشود. در سمت چپ نام کلید ها و زیر کلیدها بصورت سلسله مراتبی (مانند Windows Explorer) قرار دارد و در سمت راست نیز پارامترها جهت تنظیمات هر کلید قرار دارد. بدیهی است جهت رفتن به داخل کلید ها و زیر کلیدها باید از دستگیره (+) آن کلید استفاده کرد و قاعدتاً با کلیک کردن بر روی هر کلید در سمت راست پارامترها آن کلید لیست میشوند.

۲-۷ انواع داده در بانک رجیستری

تا اینجا ما به شرح کلیدهای ریشه رجیستری پرداختیم که خود آنها نیز به نوبه خود دارای زیر کلید یا کلیدهای فرعی بودند. اما باید بدانیم که تنظیمات یا همان پارامترها به چه صورت در رجیستری ذخیره میشوند. پارامترها و مقادیرشان در غالب مقادیر (VALUE) ذخیره میشود که ما به آنها پارامتر میگوئیم

پارامتر: هر پارامتر یک تنظیم یا تعریف داخل یک کلید است. هر پارامتر دست کم شامل ۲ بخش است. اسم (NAME) که شرحی از مقدار تعریفی است و دیگر داده ها (DATA) که خود تعریف یا تنظیم است. در ادامه باید بانیم که این پارامترها دارای چه نوع های هستند. این انواع عبارتند از :

1-STRING 2-BINARY 3-DWORD 4-MULTI STRING 5-EXPANDABLE STRING

۱- **STRING**: شامل مجموعه توضیحات حرفی - عددی , که این نوع داده های داخل گیومه نگهداری میشوند. مانند شماره سریال یا نام یک برنامه و یا نام یک فونت , مانند: "TXTfile"

۲- **BINARY**: بوسیله جفت اعداد مبنای ۱۶ نمایش داده میشود. یعنی یک رشته کارکترهای مبنای ۱۶ شامل اعداد ۰ تا ۹ و اولین ۶ حرف الفبای انگلیسی مانند: 0100AC92

۳- **DWORD**: یک مقدار حداکثر ۴ بیتی که هم بصورت کاراکترهای مبنای ۱۶ و هم کاراکترهای دهدهی نمایش داده میشود. مقادیر DWORD را تعریف میکند مانند: 0X00000112

۴- **MULTI STRING**: همان مقدار STRING است که تنظیمات چندگانه را پشتیبانی میکند

۵- **EXPANDABLE STRING**: همانطوریکه میدانیم در هر سیستم عامل متغیرهای محیطی وجود دارند که به منظور یاری رساندن به برنامه نویسان و الاخصوص استفاده های سیستم مورد بهره برداری قرار میگیرند بمنظور استفاده نمودن از متغیرهای محیطی در رجیستری ویندوز XP (مثلا مسیرهی) باید از این نوع استفاده نمود مانند:
systemroot%\system32\ctfmon.exe
لازم بذکر ست که %systemroot% متغیر محلی است که مسیر دایرکتوری ویندوز را بر میگردداند .

۲-۸ انجام عملیات مختلف در برنامه Regedit

۲-۸-۱ گرفتن نسخه پشتیبان از رجیستری و بازیابی آن

همانطور که میدانیم رجیستری پایگاه داده بسیار مهمی است که اگر دچار اشکال شود ممکن است تداخلی در کار ویندوز پیش بیاید و یا دیگر ویندوز کار نکند. برای جلوگیری از چنین حالتی امکان ایجاد کپی و بازیابی آن در Regedit موجود می باشد. این دو فعل با عنوان Import (بازیابی) و Export (ایجاد نسخه پشتیبان) در اولین و دومین گزینه های منوی File موجود میباشد. همانطور که در شکل مشاهده میکنید میتوان با تغییر گزینه Save as در کادر محاوره ای Exporting Registry File میتوان نحوه گرفتن نسخه پشتیبان را به فایل های رجیستری ویندوز 9x و NT4 و یا به یک فایل Text تغییر داد.

۲-۸-۲ انتقال دادن تنظیمات رجیستری

توجه شود که قبل از انجام هر گونه تغییرات روی رجیستری ویندوز حتما باید یک پشتیبان از فایل ها و داده های موجود در رجیستری تهیه کرد. تا از نصب دوباره سیستم عامل در مواردی که تنظیمات اشتباهی انجام گرفته باشد جلوگیری شود و با اعمال اطلاعات فایل پشتیبان این مشکل حل گردد.

برای ایجاد فایل پشتیبان از منوی فایل گزینه Export را انتخاب می کنیم.
برای اعمال کردن تنظیمات موجود در فایل پشتیبان به روی رجیستری از منوی فایل گزینه Import را انتخاب می کنیم.
فایل پشتیبان با پسوند REG. می باشد که می توان توسط Notepad محتویات آن را مشاهده کرد.

۲-۸-۳ تغییر دادن مقدار یک پارامتر

تغییر دادن مقدار یک پارامتر از یک کلید کار بسیار ساده است. برای این منظور ابتدا کلید مورد نظر را یافته پس از انتخاب پارامتر مورد نظر در سمت راست نمایش داده میشوند. روی پارامتر مورد نظر خود دبل کلیک میکنیم. طبق نوع آن پارامتر پنجره ای باز میشود که میتوان در آن مقدار پارامتر مذکور را تغییر داد .

۲-۸-۴ جستجوی یک پارامتر در Regedit

در قسمت قبل توضیح داده شد که چگونه مقدار یک پارامتر را عوض نمائیم. اما همیشه یافتن پارامتر مورد نظر کار ساده ای نیست. برای رفع این نقص در برنامه Regedit در منوی Edit گزینه ای بنام Find قرار دارد که وظیفه جستجوی پارامتر مورد نظر را عهده دار است. شکل زیر نمایی از این پنجره است .

۲-۸-۵ تغییر نام و یا حذف یک کلید یا یک پارامتر

بمنظور تغییر نام یک پارامتر و یا یک کلید از کلید F2 و یا از منوی Edit گزینه Rename استفاده میشود و همچنین بمنظور حذف نمودن یک کلید یا یک پارامتر از کلید Del و یا از منوی Edit گزینه Delete استفاده میشود.

۲-۸-۶ تعریف یک کلید یا یک پارامتر

اگر در هر طرف از دو طرف نرم افزار Regedit کلیک راست کنید منوی کرکره ای باز خواهد شد که در آن گزینه ای با عنوان New موجود است. حال با انتخاب گزینه مد نظر قادر خواهید بود یک کلید یا یک پارامتر ایجاد کنید .

۲-۹ مشاهده نتیجه تغییرات اعمال شده در رجیستری

برای مشاهده تغییرات اعمال شده روی رجیستری سیستم عامل باید سیستم را یک بار Restart یا Log off کرد . برای مشاهده سریع تغییرات می توان راه کار زیر را بر اساس نوع سیستم عامل ، اعمال کرد:

در ویندوز Xp,2000,NT

۱- با انتخاب برگه Process از صفحه Task manager ، Explorer.exe را انتخاب کرده و از اجرا خارج می کنیم.

۲- با استفاده از New Task و نوشتن explorer.exe دوباره آن را اجرا می کنیم.

۲-۱۰ چگونه تغییرات در رجیستری را غیر فعال کنیم؟

برای اینکه از تغییرات رجیستری ویندوز توسط کاربران جلوگیری کنیم می توانیم رجیستری را ببندیم.

راه اول (با استفاده از پنجره regedit):

از منوی Start وارد Run شده و عبارت regedit را وارد نمایید ، سپس Enter را بزنید تا وارد ویرایشگر رجیستری شوید و به مسیر زیر بروید:

HKEY_Current_User\Software\Microsoft\Windows\CurrentVersion\Policies\System

در قسمت دیگر صفحه ، بر روی فضای خالی از صفحه راست کلیک کرده و از New گزینه DWORD Value را انتخاب کنید و نام آن را DisableTaskMgr قرار دهید.

بر روی DisableRegistryTools دوبار کلیک کرده و مقدار Value را بر روی ۱ تنظیم سپس OK کنید.

جهت بازگرداندن به حالت قبلی و فعال کردن Regedit کافی است عدد ۱ را به ۰ تغییر دهید.

راه دوم (با استفاده از پنجره Group Policy):

۱- از منوی Start/RUN برنامه GPedit.msc را اجرا کنید.

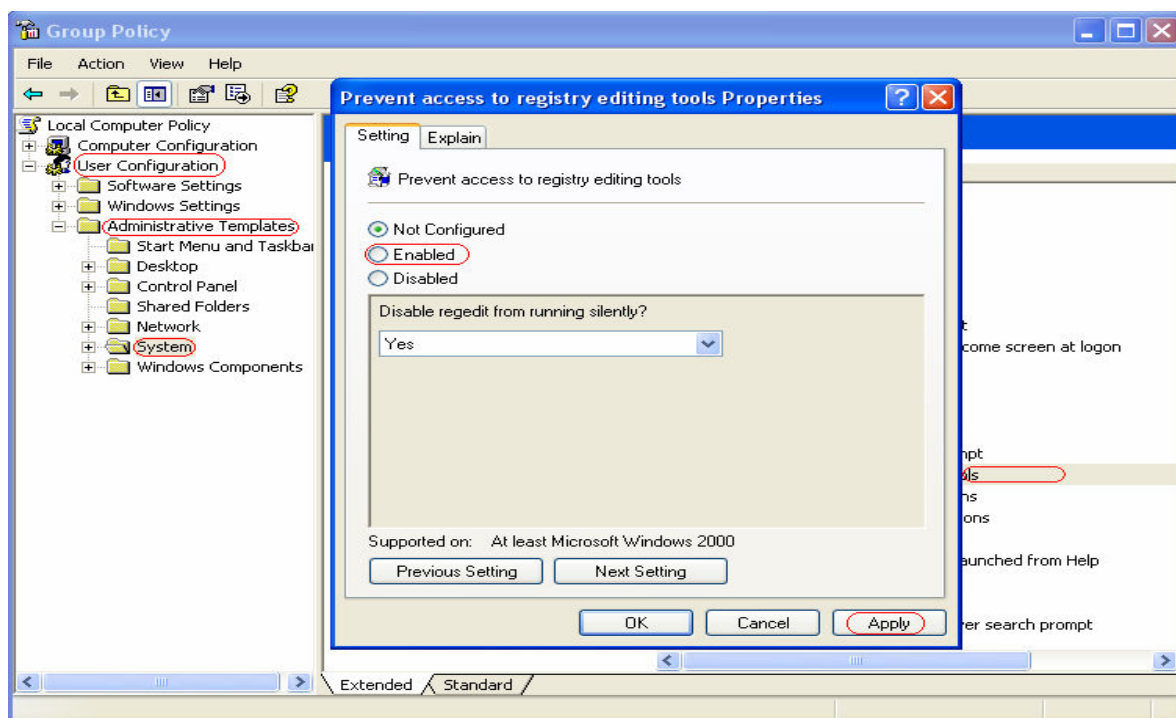
۲- از پنجره ظاهر شده مسیر زیر را پیدا کنید.

user configuration \ administrative templates \ system

۳- حال از پنجره ظاهر شده گزینه prevent access to registry editing tools را دابل کلیک کنید و از

پنجره ظاهر شده گزینه Enable را انتخاب کرده و پنجره را با زدن Ok ببندید. حال دیگر رجیستری باز نخواهد شد.

برای فعال کردن عکس این کار را انجام دهید.



شکل ۲-۴: غیر فعال کردن تغییرات در رجیستری

۲-۱۱ چند مثال از تغییرات در رجیستری

۲-۱۱-۱ غیر فعال کردن Task Manager

حتما شما نیز با پنجره Task Manager آشنا هستید. پنجره ای که در مواقع بحرانی با فشردن سه کلید **Ctrl + Alt + Delete** به کمک شما می آید. بنا به هر دلیلی ممکن است دوست داشته باشید Task Manager را در ویندوز غیر فعال کنید .

راه اول (با استفاده از پنجره **regedit**):

از منوی **Start** وارد **Run** شده و عبارت **regedit** را وارد نمایید ، سپس **Enter** را بزنید و به مسیر زیر بروید:

HKEY_Current_User\Software\Microsoft\Windows\CurrentVersion\Policies\System

در قسمت دیگر صفحه ، بر روی فضای خالی از صفحه راست کلیک کرده و از **New** گزینه **DWORD Value** را انتخاب کنید.

سپس نام آن را **DisableTaskMgr** قرار دهید.

بر روی **DisableTaskMgr** دابل کلیک کرده و مقدار **Value** را بر روی ۱ تنظیم کنید . با **OK** کردن تنظیمات ذخیره می شود.

جهت بازگرداندن به حالت قبلی و فعال کردن Task Manager کافی است عدد ۱ را به ۰ تغییر دهید.

راه دوم (با استفاده از پنجره **Group Policy**):

در این تکنیک دسترسی به Task Manager را بدون استفاده از رجیستری غیر فعال می کنیم.

از منوی **Start** وارد **Run** شده و عبارت **gpedit.msc** را وارد نمایید ، سپس **Enter** را بزنید و به مسیر زیر بروید:

User Configuration\Administrative Templates\System\Ctrl+Alt+Delete options

حال در قسمت سمت راست پنجره روی **Remove Task Manager** دابل کلیک کرده و از پنجره باز شده چک باکس **Enabled** را انتخاب کنید.

۲-۱۱-۲ غیر فعال کردن دستور Shut Down

برای غیر فعال کردن فرمان Shut Down کلید زیر را در رجیستری پیدا کنید:

[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer]

یک مقدار جدید از نوع **DWORD** با نام **noclose** ایجاد کرده و آنرا برابر با 1 قرار دهید تا فرمان Shut Down غیر فعال گردد. برای فعال سازی مجدد آن ، مقدار فوق را برابر با 0 قرار دهید

۲-۱۱-۳ از کار انداختن راست کلیک در Desktop

وارد رجیستری شوید و کلید زیر را پیدا کنید :

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\Currentversion\Policies\Explorer

یک متغیر جدید از نوع DWORD بسازید و اسم آن را NoViewContextMenu را به آن بدهید با دادن مقدار ۱ به این متغیر منوی مزبور را از کار میندازید و با دادن مقدار ۰ به آن دوباره آن را به کار می اندازید .

۲-۱۱-۴ برداشتن منوی file از مرورگر ویندوز

رجیستری را باز کرده و کلید زیر را پیدا کنید :

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\Currentversion\Policies\Explorer

یک متغیر جدید از نوع DWORD بسازید و به آن نام NofileMenu بدهید اگر به این متغیر مقدار ۱ بدهید حذف و اگر مقدار ۰ بدهید به جای خود باز میگردد .

۲-۱۱-۵ نمایش یک عنوان دلخواه برای قسمت Logon در ویندوز

به کلید زیر بروید :

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\Current Version\Winlogon

داده ای از نوع String تعریف و نام آنرا LegaNoticeCaption قرار دهید .
حال مقدار آنرا متن دلخواه خود قرار دهید . این متن درون پنجره نمایش داده میشود .
داده ای دیگر از نوع String ایجاد و نام آنرا LegaNoticeText قرار دهید . این متن در بخش عنوان پنجره قرار میگیرد .

۲-۱۱-۶ تنظیم پنجره خوش آمدگویی ویندوز

به کلید زیر بروید :

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Explorer\Tips

خوب شما میتوانید در این قسمت ۴۸ نکته در ویندوز ۹۸ و ۹۵ و ۵۰ نکته در XP و NT قرار دهید که در هنگام بالا آمدن ویندوز برای شما نمایش دهد شما میتوانید متغیرهایی از نوع STRING VALUE بسازید و به آن نامی از ۰ تا ۵۰ بدهید و نکات خود را در آن قرار دهید تا با هر بار بالا آمدن ویندوز به نمایش درآید این نکته بسیار بسیار جالبی است که خیلی کارا میشه باهاش کرد مانند نکاتی که میخواستیم کاربران ویندوز آنها را رعایت کنند و

۲-۱۲ برنامه نویسی رجیستری به وسیله C#

در ابتدا با این دید که registry اطلاعاتی بیش از آنچه در فایل است را نگهداری نمی کند و شیوه کاری .NET. ذخیره سازی اطلاعات به صورت فایل های XML است موضوع را پی می گیریم.

پیش از هر کاری یک workspace در .NET Visual Studio ایجاد کنید.

File → New → Project

و در قسمت نام ، یک نام برای پروژه انتخاب کنید.

برای شروع باید به صورت صریح Microsoft.win32 را در کد خود به عنوان Refrence قرار دهید تا به ابزار دستکاری Registry دست یابید.

```
using Microsoft.Win32;
```

این refrence در بین دیگر refrence های برنامه قرار می گیرد، که دستیابی به اجزای مرتبط با Registry برای کنترل آن را فراهم می کند.

دو کلاس در این namespace قرار دارد:

Registry-۱

RegistryKey-۲

این دو کلاس با هم استفاده می شوند تا بهتر بتوان عملیات های مختلف در Registry را انجام داد.

۲-۱۲-۱ کلاس Registry

کلاس Registry به تنهایی نمی تواند ابزار های زیادی در اختیار کاربر قرار دهد زیرا این کلاس به ندرت هفت نود سطح بالا در Registry را برای دستیابی و دستکاری در اختیار کاربر قرار می دهد وقتی این کلاس، در داخل کلاس RegistryKey استفاده می شود قدرتمند است. با این وجود با ورود به محیط .NET Visual Studio و افزودن namespace فوق و تایپ کردن " Registry. " ، لیست هفت Subtree زیر مشاهده خواهد شد.

Registry Class Name	Registry Value
Registry.ClassesRoot	HKEY_CLASSES_ROOT
Registry.CurrentConfig	HKEY_CURRENT_CONFIG
Registry.CurrentUser	HKEY_CURRENT_USER
Registry.DynData	HKEY_DYN_DATA
Registry.LocalMachine	HKEY_LOCAL_MACHINE
Registry.PerformanceData	HKEY_PERFORMANCE_DATA
Registry.Users	HKEY_USERS

بعد از شناسایی کلاس Registry به بررسی کلاس RegistryKey می پردازیم.

۲-۱۲-۲ کلاس RegistryKey

این کلاس در بخش دستکاری Registry شامل بخش های بیشتری است که استفاده از آنها می تواند منجر به ایجاد یا حذف داده شود.

برای استفاده از این کلاس باید یک نمونه از کلاس را ایجاد کنیم.

```
RegistryKey OurKey = Registry.Users;
```

در کد بالا یک نمونه از این کلاس با نام `OurKey` ایجاد کرده ایم که در داخل `Registry` به صورت یک زیرکلید از `HKEY_USERS` مقداردهی شده است. اساساً این مقداردهی به این معناست که هر متد استفاده شده برای این کلید تاثیر مستقیم در زیر کلید (`subkey`)، `HKEY_USERS` و آیتمهای داده ای آن دارد.

۲-۱۲-۳ متدهای کلاس `RegistryKey` متد `OpenSubKey()`:

این متد یکی از مهمترین متدهای کلاس `RegistryKey` است زیرا اجازه دسترسی و دستکاری `subkey` مربوط به کلید سطح بالا را به کاربر می دهد. ممکن است عجیب به نظر برسد اگر همانند بالا از نوع کلید `HKEY_USERS` قرار داده باشیم. با استفاده از این متد می توان `OurKey` را به یک زیر کلید از نوع `HKEY_USERS` تبدیل کرد.

```
RegistryKey OurKey = Registry.Users;
OurKey = OurKey.OpenSubKey(".DEFAULT", true);
```

در مثال بالا یک نمونه از کلاس `RegistryKey` ایجاد شده است و `value` آن توسط کلید `HKEY_USERS` مقدار دهی شده است. سپس یک `subkey` در داخل `HKEY_USERS` که `".DEFAULT"` نام دارد باز (`Open`) شده است. پارامتر دوم متد تعیین کننده متد کلید شامل خواندن یا خواندن/نوشتن است. مقدار `true` به معنای قابلیت ویرایش کلید است.

متدهای `DeleteSubKey()` / `CreateSubKey()` & `DeleteSubKeyTree()`

هر سه متد فوق به مدیریت `subkey` های کلید جاری می پردازند و نام آنها به خوبی بیانگر عملیاتی است که انجام می دهند. کد زیر نحوه پیاده سازی آنها را نشان می دهد.

```
RegistryKey OurKey = Registry.Users;
// Create OurKey set to HKEY_USERS
OurKey.OpenSubKey(".DEFAULT", true);
// Set it to HKEY_USERS\DEFAULT
OurKey.CreateSubKey(@"DEFAULT\OurSubKey");
// Create the key HKEY_USERS\DEFAULT\OurSubKey
OurKey.CreateSubKey(@"DEFAULT\OurSubKey\Subkey");
// Create a sub key HKEY_USERS\DEFAULT\OurSubKey\Subkey
OurKey.DeleteSubKey(@"DEFAULT\OurSubKey\Subkey");
// Delete the subkey name "subkey"
OurKey.DeleteSubKeyTree(@"DEFAULT\OurSubKey");
// Delete the whole subkey and any subkeys below it
```

متد :GetSubKeyNames()

این متد اجازه یافتن نام subkey های ثانویه را به کاربر می دهد. به عنوان نمونه می توان همه نام های subkey های HKEY_USERS را به دست آورد. به کمک یک الگوریتم recursive می توان همه نام ها را به صورت یکجا مشاهده کرد. قطعه کد اول یک مثال ساده از متد فوق است.

```
foreach (string Keyname in OurKey.GetSubKeyNames())
{
    MessageBox.Show(Keyname);
}
//The second example shows how to tranfer the names into a string array
string[] Keynames = OurKey.GetSubKeyNames();
```

```
private void GetSubKeys(RegistryKey SubKey)
{
    foreach (string sub in SubKey.GetSubKeyNames())
    {
        MessageBox.Show(sub);
        RegistryKey local = Registry.Users;
        local = SubKey.OpenSubKey(sub,true);
        GetSubKeys(local);
        // By recalling itself it makes sure it get all the subkey names
    }
}
```

```
//This is how we call the recursive function GetSubKeys
RegistryKey OurKey = Registry.Users;
OurKey.OpenSubKey(".DEFAULT", true);
GetSubKeys(OurKey);
```

در قطعه کد بعدی مشاهده می شود که می توان از توابع بازگشتی برای بدست آوردن همه نام های subkey ها استفاده کرد.

متدهای :GetValue() / GetValuesNames()

هنگام استخراج اطلاعات از registry، از متد های فوق استفاده می شود. پس از استفاده از متد های دیگر برای دسترسی به subkey ها، به کمک این دو متد می توان اطلاعات subkey ها را از آن ها استخراج کرد. متد اول GetValue() اجازه دسترسی به مقدار نگهداری شده در بخش registry value را با استفاده از نام یک subkey به کاربر می دهد. به عنوان مثال در زیر کلید "test"، یک داده از نوع string (REG_SZ)، که نام آن "Testvalue" و مقدار آن "This is a test" است، قرار دارد. برای دسترسی به مقدار داده از قطعه کد زیر استفاده می شود.

```
RegistryKey OurKey = Registry.Users;
// Create OurKey set to HKEY_USERS
OurKey.OpenSubKey(".DEFAULT", true);
MessageBox.Show(OurKey.GetValue("Testvalue").ToString());
```


متد SetValue()

این متد یک Value را در داخل registry value مقداردهی می کند. برای استفاده از این متد شما باید نام و مقدار value را به متد ارسال کنید. در صورت عدم وجود value متد فوق آن را ایجاد خواهد کرد.

```
RegistryKey OurKey = Registry.Users;
OurKey.OpenSubKey(".DEFAULT", true);
OurKey.SetValue("Testvalue", 0);
```

استفاده از Property های Valuecount و SubKeyCount :

Property اول تعداد subkey هایی که مستقیماً در زیر یک کلید قرار دارند را مشخص می کند و Property دوم تعیین کننده تعداد value ها در یک subkey معین است.

```
RegistryKey OurKey = Registry.Users;
OurKey.OpenSubKey(".DEFAULT", true);
MessageBox.Show(OurKey.SubKeyCount.ToString());
MessageBox.Show(OurKey.ValueCount.ToString());
```

باید به این نکته اشاره کرد که وقتی کار با Registry به پایان رسید، باید به کمک متد Close() به عملیات خود خاتمه دهید مانند کد زیر:

```
RegistryKey OurKey = Registry.Users;
OurKey.Close();
```

۲-۱۲-۴ مثالی از یک برنامه Registry با C#

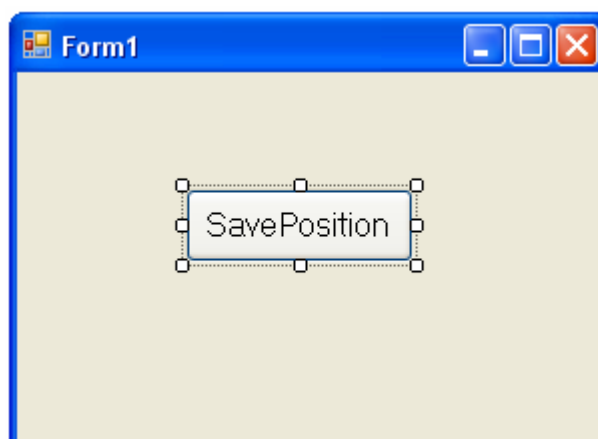
حال به ارائه مثالی از Registry می پردازیم:

فرض کنید که می خواهیم Location یک فرم را توسط Registry کنترل کنیم. برای این کار از یک کلید استفاده می کنیم که از نوع HKEY_CURRENT_USER تعریف شده است. برای ذخیره Location که از Point است نیاز به دو Value با نامهای X و Y می باشد.

```
// Attempt to open the key
RegistryKey key =
    Registry.CurrentUser.OpenSubKey("Software\\Play\\WindowPos");
// If the return value is null, the key doesn't exist
if (key == null)
{
    // The key doesn't exist; create it / open it
    key = Registry.CurrentUser.CreateSubKey(
        "Software\\Play\\WindowPos");
}
// Attempt to retrieve the value X; if null is returned, the Value
// doesn't exist in the registry.
if (key.GetValue("X") != null)
{
    // The value exists; move the form to the coordinates stored in the
    // registry
    Location = new Point((int)key.GetValue("X"), (int)key.GetValue("Y"));
}
```

در این کد ما سعی کرده ایم مقادیر X و Y را از کلید فوق بازیابی کنیم اگر این دو Value وجود داشته باشند فرم را به مکان مناسب انتقال می دهیم که توسط تنظیم Location فرم این کار انجام می شود.

Location از نوع Point است پس ابتدا یک Object از نوع Point تعریف می کنیم و آن را با Value های مناسب از registry-lookup مقداردهی می کنیم و آن را به عنوان مقدار جدید Location، قرار می دهیم. مکان فرم هنوز Save نشده است پس Value های X و Y هنوز در کلید Registry وجود ندارند. برای اینکه اطلاعات ذخیره شود یک Button به فرم اضافه می کنیم. برای این کار یک Button در بخش Designer به فرم اضافه کرده و قطعه کد زیر را در کد Button می نویسیم.



```
// Open the key
RegistryKey key =
    Registry.CurrentUser.OpenSubKey("Software\\Play\\WindowPos", true);
// Set the registry values to correspond to the form's coordinates on the
screen.
key.SetValue("X", Location.X);
key.SetValue("Y", Location.Y);
```

نکته ای که در این کد باید مورد توجه قرار گیرد اهمیت پارامتر دوم متد `openSubKey` است که از نوع `Boolean` می باشد. اگر این پارامتر `true` باشد، `Key` در مد نوشتاری باز می شود و به کاربر اجازت می دهد از متد `SetValue` استفاده کند در غیر این صورت (در صورت `false` بودن این پارامتر) فراخوانی این متد سبب بروز خطای زیر می شود:

error: UnauthorizedAccessException

حال با زدن کلید `F5` می توان برنامه را اجرا کرده و با تغییر مکان فرم و زدن `Button` طراحی شده `Location` فرم را ذخیره می کنیم.

۲-۱۳ دستور کار

با استفاده از تغییرات در رجیستری کاری کنید که:

۱. تصویر desktop عوض نشود
 ۲. control panel از لیست برنامه ها حذف شود
 ۳. قسمت run غیر فعال شود و یا حذف شود (سپس دوباره فعال شود)
 ۴. از تغییر پسورد توسط کاربر جلوگیری شود.
- مراحل کار را بنویسید

۲-۱۴ پروژه برنامه نویسی

برنامه ای بنویسید که

- الف- بدون مراجعه به محیط ویرایشگر رجیستری ، نام کلید، مسیر ایجاد کلید ، نام مقادیر موجود در کلید ، نوع داده مقادیر و مقدار آن را گرفته و در رجیستری ایجاد کند.
- ب- با استفاده از دستورات برنامه بالا نرم افزاری طراحی کنید که بطور اتوماتیک تنظیمات یکی از تکنیک های رجیستری را انجام دهد.

۲-۱۵ منابع

www.computing.net
www.IRITN.com
www.hamkelasy.com
www.iau-neyshabur.ac.ir

بخش ۳:

آشنایی با



و دستورات آن

پیش آگاهی

هدف از گردآوری این بخش، آشنا ساختن شما با چگونگی کار با ایت سیستم عامل است. البته در حدی که شما را قادر سازد، آزمایشهای مربوط به سیستم عامل لینوکس را انجام دهید.

پس از مطالعه این بخش، خواهید توانست:

- با سیستم ارتباط برقرار کنید.
- با پرونده ها کار کنید و برنامه های قابل اجرا را اجرا کنید.
- از راهنمای سیستم استفاده کرده، اطلاعات مورد نیاز خود را از آن بیابید.
- پرونده های خود را با ویرایشگرها، ویرایش کنید.
- برنامه های خود را با ابزارهای برنامه نویسی، ترجمه (Compile)، اشکال زدایی (Debug) و اجرا کنید.

۳-۱ تاریخچه لینوکس

۳-۱-۱ آغاز داستان

در سال 1991 در حالی که جنگ سرد رو به پایان میرفت و صلح در افقها هویدا میشد، در دنیای کامپیوتر، آینده بسیار روشنی دیده میشد. با وجود قدرت سخت افزارهای جدید، محدودیت های کامپیوترها رو به پایان میرفت. ولی هنوز چیزی کم بود... و این چیزی نبود جز فقدان عمیق در حیطه سیستم های عامل.

داس، امپراطوری کامپیوترهای شخصی را در دست داشت. سیستم عامل بی استخوانی که با قیمت ۵۰۰۰۰ دلار از یک هکر سیاتلی توسط بیل گیتز (Bill Gates) خریداری شده بود و با یک استراتژی تجاری هوشمند، به تمام گوشه های جهان رخنه کرده بود. کاربران PC انتخاب دیگری نداشتند. کامپیوترهای اپل مکینتاش بهتر بودند. ولی قیمتهای نجومی، آنها را از دسترس اکثر افراد خارج می ساخت.

خیمه گاه دیگر دنیای کامپیوترها، دنیای یونیکس بود. ولی یونیکس به خودی خود بسیار گرانبه بود. آنقدر گرانبه که کاربران کامپیوترهای شخصی جرات نزدیک شدن به آنرا نداشتند. کد منبع یونیکس که توسط آزمایشگاههای بل بین دانشگاهها توزیع شده بود، محتاطانه محافظت میشد تا برای عموم فاش نشود. برای حل شدن این مسئله، هیچیک از تولید کنندگان نرم افزار راه حلی ارائه ندادند.

بنظر میرسید این راه حل به صورت سیستم عامل MINIX ارائه شد. این سیستم عامل، که از ابتدا توسط اندرو اس. تاننباوم (Andrew S. Tanenbaum) پروفیسور هلندی، نوشته شده بود به منظور تدریس عملیات داخلی یک سیستم عامل واقعی بود. این سیستم عامل برای اجرا روی پردازنده های ۸۰۸۶ اینتل طراحی شده بود و بزودی بازار را اشباع کرد.

بعنوان یک سیستم عامل، MINIX خیلی خوب نبود. ولی مزیت اصلی آن، در دسترس بودن کد منبع آن بود. هرکس که کتاب سیستم عامل تاننباوم را تهیه میکرد، به ۱۲۰۰۰ خط کد نوشته شده به زبان C و اسمبلی نیز دسترسی پیدا میکرد. برای نخستین بار، یک برنامه نویس یا هکر مشتاق میتواند کد منبع سیستم عامل را مطالعه کند. چیزی که سازندگان نرم افزارها آنرا محدود کرده بودند. یک نویسنده بسیار خوب، یعنی تاننباوم، باعث فعالیت مغزهای متفکر علوم کامپیوتری در زمینه بحث و گفتگو برای ایجاد سیستم عامل شد. دانشجویان کامپیوتر در سرتاسر دنیا با خواندن کتاب و کدهای منبع، سیستمی را که در کامپیوترشان در حال اجرا بود، درک کردند. و یکی از آنها لینوس توروالدز (Linus Torvalds) نام داشت.



۳-۱-۲ کودک جدید در افق

در سال ۱۹۹۱، لینوس بندیکت توروالدز (Linus Benedict Torvalds) دانشجوی سال دوم علوم کامپیوتر دانشگاه هلسینکی فنلاند و یک هکر خود آموخته بود. این فنلاندی ۲۱ ساله، عاشق وصله پینه کردن محدودیت هایی بود که سیستم را تحت فشار قرار میدادند. ولی مهمترین چیزی که وجود نداشت یک سیستم عامل بود که بتواند نیازهای حرفه ای ها را برآورده نماید. MINIX خوب بود ولی فقط یک سیستم عامل مخصوص دانش آموزان بود و بیشتر به عنوان یک ابزار آموزشی بود تا ابزاری قدرتمند برای بکار گیری در امور جدی.

در این زمان برنامه نویسان سرتاسر دنیا توسط پروژه گنو (GNU) که توسط ریچارد استالمن (Richard Stallman) آغاز شده بود، تحریک شده بودند. هدف این پروژه ایجاد حرکتی برای فراهم نمودن نرم افزارهای رایگان و در عین حال با کیفیت بود. استالمن خط مشی خود را از آزمایشگاه معروف هوش مصنوعی دانشگاه MIT با ایجاد برنامه ویرایشگر emacs در اواسط و اواخر دهه ۷۰ آغاز نمود. تا اوایل دهه ۸۰، بیشتر برنامه نویسان نخبه آزمایشگاههای هوش مصنوعی MIT جذب شرکتهای نرم افزاری تجاری شده بودند و با آنها قرارداد های حفظ اسرار امضا شده بود. ولی استالمن دیدگاه متفاوتی داشت. وی عقیده داشت برخلاف سایر تولیدات، نرم افزار باید از محدودیت های کپی و ایجاد تغییرات در آن آزاد باشد تا بتوان روز به روز نرم افزارهای بهتر و کارآمد تری تولید نمود.

با اعلامیه معروف خود در سال ۱۹۸۳، پروژه GNU را آغاز کرد. وی حرکتی را آغاز کرد تا با فلسفه خودش به تولید و ارائه نرم افزار بپردازد. نام GNU مخفف GNU is Not Unix است. ولی برای رسیدن به رویای خود برای ایجاد یک سیستم عامل رایگان، وی ابتدا نیاز داشت تا ابزارهای لازم برای این کار را ایجاد نماید. بنابراین در سال ۱۹۸۴ وی شروع به نوشتن و ایجاد کامپایلر زبان C گنو موسوم به GCC نمود. ابزاری مبهوت کننده برای برنامه نویسان مستقل. وی با جادوگری افسانه ای خود به تنهایی ابزاری را ایجاد نمود که برتر از تمام ابزارهایی که تمام گروههای برنامه نویسان تجاری ایجاد کرده بودند قرار گرفت. GCC یکی از کارآمد ترین و قویترین کامپایلرهای است که تا کنون ایجاد شده اند.

تا سال ۱۹۹۱ تعداد زیادی ابزار ایجاد کرده بود ولی هنوز سیستم عامل رایگانی وجود نداشت. حتی MINIX هم لایسنس شده بود. کار بر روی هسته سیستم عامل گنو موسوم به HURD ادامه داشت ولی به نظر نمی رسید که تا چند سال آینده قابل استفاده باشد.



torvalds@osdl.org

این زمان برای توروالدز بیش از حد طولانی بود ...

در ۲۵ آگوست ۱۹۹۱، این نامه تاریخی به گروه خبری MINIX از طرف توروالدز ارسال شد:

از : لینوس بندیکت توروالدز

به: گروه خبری MINIX

موضوع: بیشتر چه چیزی را میخواهید در MINIX ببینید؟

خلاصه: نظرخواهی کوچک در مورد سیستم عامل جدید من

با سلام به تمام استفاده کنندگان از MINIX

من در حال تهیه یک سیستم عامل رایگان فقط به عنوان سرگرمی و نه به بزرگی و حرفه ای GNU برای دستگاههای ۳۸۶ و ۴۸۶ هستم. این کار از آوریل شروع شده و در حال آماده شدن است. من مایلم تا نظرات کاربران را در مورد چیزهایی که در MINIX دوست دارند یا ندارند، جمع آوری کنم.

زیرا سیستم عامل من حدودا شبیه آن است. مانند ساختار سیستم فایل مشابه و چیزهای دیگر... من اکنون bash نسخه ۱.۰۸ و GCC نسخه ۱.۴۰ را به آن منتقل کرده ام و به نظر میرسد که کار میکند. من در عرض چند ماه چیزی آزمایشی درست کرده ام

و مایلم بدانم که کاربران بیشتر به چه قابلیت‌هایی نیاز دارند؟ من از هر پیشنهادی استقبال می‌کنم. ولی قول نمی‌دهم همه آنها را اجرا کنم. لینوس

همانطور که در این نامه پیداست، خود توروالدز هم باور نمی‌کرد که مخلوقش آنقدر بزرگ شود که چنین تحولی در دنیا ایجاد کند. لینوکس نسخه ۰٫۰۱ در اواسط سپتامبر ۱۹۹۱ منتشر شد و روی اینترنت قرار گرفت. شور و اشتیاقی فراوان حول مخلوق توروالدز شکل گرفت. کدها دانلود شده، آزمایش شدند و پس از بهینه‌سازی به توروالدز بازگردانده شدند. لینوکس نسخه ۰٫۰۲ در پنجم اکتبر به همراه اعلامیه معروف توروالدز آماده شد:

از : لینوس بندیکت توروالدز

به: گروه خبری MINIX

موضوع: کدهای منبع رایگان هسته مشابه MINIX

آیا شما از روزهای زیبای MINIX 1.1 محروم شده‌اید؟ هنگامی که مردها مرد بودند و راه اندازهای دستگاه خود را خودشان مینوشتند؟ آیا شما فاقد یک پروژه زیبا هستید و می‌توانید تا سیستم عاملی داشته باشید تا بتوانید آنها مطابق با نیازهای خود در آورید؟ اگر اینگونه است، این نامه برای شما نوشته شده است.

همانطور که ماه پیش گفتم من در حال کار بر بروی یک سیستم عامل رایگان مشابه MINIX برای کامپیوترهای ۳۸۶ هستم. این سیستم عامل اکنون بجایی رسیده است که قابل استفاده است و مایل هستم که کدهای منبع را در سطح گسترده تر پخش نمایم. این نسخه ۰٫۰۲ است ولی من موفق شده‌ام که نرم افزارهای GNU-sed، GNU-Make، GCC، Bash، Compress و غیره را تحت آن اجرا کنم. کدهای منبع این پروژه را میتوانید از آدرس nic.funet.fi با آدرس ۱۲۸،۲۱۴،۶،۱۰۰ در دایرکتوری pub/OS/Linux پیدا کنید. این دایرکتوری همچنین دارای چند فایل README و تعدادی باینری قابل اجرا تحت لینوکس است. تمام کدهای منبع ارائه شده است زیرا هیچ یک از کدهای MINIX در آن استفاده نشده است. سیستم را میتوانید همانطور که هست کامپایل و استفاده کنید. کدهای منبع باینری ها را هم میتوانید در مسیر pub/GNU پیدا کنید.

لینوکس نسخه ۰٫۰۳ پس از چند هفته آماده شد و تا دسامبر، لینوکس به نسخه ۰٫۱۰ رسید. هنوز لینوکس فقط چیز کمی بیشتر از یک فرم اسکلت بود. این سیستم عامل فقط دیسکهای سخت AT را پشتیبانی میکرد و ورود به سیستم نداشت و مستقیماً به خط فرمان بوت میشد. نسخه ۰٫۱۱ خیلی بهتر شد. این نسخه از صفحه کلیدهای چند زبانه پشتیبانی میکرد، دیسکهای فلاپی و کارتهای گرافیکی VGA، EGA، هرکولس و... نیز پشتیبانی میشدند. شماره نسخه ها از ۰٫۱۲ به ۰٫۹۵ و 0.96 افزایش پیدا کرد و ادامه یافت. بزودی کد آن بوسیله سرویس دهنده های FTP در فنلاند و مناطق دیگر، در سرتاسر جهان منتشر شد.

۳-۱-۳ مقایسه و توسعه

بزودی توروالدز با مقایسه هایی از طرف اندرو تانباوم، معلم بزرگی که MINIX را نوشته بود، مواجه شد. تانباوم برای توروالدز مینویسد:

"من بر این نکته تاکید دارم که ایجاد یک هسته یکپارچه در سال ۱۹۹۱ یک اشتباه پایه ای بود. خدا را شکر که شما شاگرد من نیستید، و اگر نه برای چنین طرحی نمره بالایی نمی‌گرفتید".

توروالدز بعداً پذیرفت که این بدترین نکته در توسعه لینوکس بوده است. تانباوم یک استاد مشهور بود و هرچه که می‌گفت واقعیت داشت. ولی وی در مورد لینوکس اشتباه میکرد. توروالدز کسی نبود که به این سادگی ها پذیرای شکست باشد.

تانباوم همچنین گفته بود: "لینوکس منسوخ شده است."

اکنون نوبت حرکت نسل جدید لینوکس بود. با پشتیبانی قوی از طرف اجتماع لینوکس، توروالدز یک پاسخ مناسب برای تانباوم فرستاد:

"شغل شما استاد دانشگاه و محقق بودن است و این بهانه خوبی برای برخی مغز خرابکنی های MINIX است."

و کار ادامه یافت. بزودی صدها نفر به اردوگاه لینوکس پیوستند. سپس هزاران نفر و سپس صدها هزار نفر. لینوکس دیگر اسباب بازی هکرها نبود. با پشتیبانی نرم افزارهای پروژه GNU، لینوکس آماده یک نمایش واقعی بود. لینوکس تحت مجوز GPL قرار داده شد. با این مجوز همه می‌توانستند کدهای منبع لینوکس را به رایگان داشته باشند، بر روی آنها مطالعه کرده و آنها را تغییر دهند. دانشجویان و برنامه نویسان آنرا قاپیدند.

و خیلی زود تولید کنندگان تجاری وارد شدند. لینوکس به خودی خود رایگان بود و هست. کاری که این تولیدکنندگان انجام دادند، کامپایل کردن بخش ها و نرم افزارهای مختلف و ارائه آن بصورت یک فرمت قابل توزیع همانند سایر سیستم عاملها بود، تا مردم عادی نیز بتوانند از آن استفاده کنند. اکنون توزیع هایی مانند ردهت، دیبان و سوزی دارای بیشترین سهم کاربران در سرتاسر جهان هستند. با رابطهای گرافیکی کاربر جدید مانند KDE و GNOME، توزیع های لینوکس در بین مردم بسیار گسترش یافتند.

همچنین اتفاقات جالبی با لینوکس رخ میدهد. در کنار PC، لینوکس به روی اکثر پلتفرمها منتقل شده است. لینوکس تغییر داده شد تا کامپیوتر دستی شرکت 3Com یعنی PalmPilot را اجرا نماید. تکنولوژی کلاستر کردن این امکان را بوجود آورد تا بتوان تعداد زیادی از ماشینهای لینوکس را به یک مجموعه واحد پردازشی تبدیل نمود. یک کامپیوتر موازی. در آوریل ۱۹۹۶ محققین آزمایشگاههای ملی لوس آلاموس از ۶۸ کامپیوتر مبتنی بر لینوکس برای پردازش موازی و شبیه سازی موج انفجار اتمی استفاده کردند. ولی بر خلاف ابر کامپیوترهای دیگر، هزینه آنها بسیار ارزان تمام شد. ابر کامپیوتر خود ساخته آنها با تمام تجهیزات و سخت افزارها ۱۵۲۰۰۰ دلار هزینه در بر داشت و این یک دهم هزینه یک ابر کامپیوتر تجاری است. این ابر کامپیوتر به سرعت ۱۶ بیلیون محاسبه در ثانیه دست یافت و به رتبه ۳۱۵ام این ابر کامپیوتر جهان دست پیدا کرد و صد البته یکی از پایدارترین آنها بود. پس از سه ماه از آغاز فعالیت، هنوز بوت نشده بود.

بهترین موردی که امروزه برای لینوکس وجود دارد، طرفداران متعصب آن هستند. هنگامی که یک قطعه سخت افزاری جدید ارائه میشود، هسته لینوکس برای استفاده از آن تغییر داده میشود. برای مثال هنگام ارائه پردازنده ۶۴ بیتی شرکت AMD هسته به سرعت چند هفته برای کار با آن آماده شد. اکنون لینوکس بر روی تمام انواع خانواده های سخت افزاری موجود اعم از PC، Alpha، MAC و انواع سخت افزارهای درونه ای قابل اجراست که آنرا برای استفاده در ماشین آلات صنعتی و آلات و ادواتی که نیاز به پردازش کامپیوتری دارند، بسیار مناسب نموده است. لینوکس با همان فلسفه و هدفی که در سال ۱۹۹۱ ایجاد شد، وارد هزاره جدید شده است.

توروالدز، هنوز یک انسان ساده است. بر خلاف بیل گیتز او یک میلیاردر نیست. پس از اتمام مطالعاتش وی به آمریکا رفت تا با شرکت Transmeta همکاری نماید. پس از انجام یک پروژه فوق سری که توروالدز یکی از اعضای فعال آن بود، ترانسمتا پردازنده Cruose را با بازار ارائه کرد. توروالدز هنوز پرترفدار ترین و مشهورترین برنامه نویس جهان است. در حال حاضر توروالدز ترانسمتا را ترک نموده و با حمایت شرکتهای بزرگ به طور تمام وقت بر روی لینوکس کار میکند.

پس از یک دهه : لینوکس امروز

امروزه لینوکس بیش از یک دهه توسعه را پشت سر گذاشته است و یکی از سریع التوسعه ترین سیستم های عامل به شما میرود. از چند کاربر انگشت شمار در سالهای ۱۹۹۱ و ۱۹۹۲، امروزه میلیونها کاربر از لینوکس استفاده میکنند IBM. که زمانی بزرگترین دشمن جماعت Open Source به شمار می رفت، اکنون سرمایه گذاری عظیمی در زمینه توسعه راه حل های Open Source تحت لینوکس نموده است. در حال حاضر تعداد توسعه دهندگانی که برای افزایش قابلیتهای لینوکس تلاش میکنند، روز به روز افزایش می یابد.

امروزه تعداد زیادی از شرکتهای و موسسات حرفه ای تجاری، پشتیبانی از محصولات مبتنی بر لینوکس را بر عهده گرفته اند. اکنون دیگر استفاده از لینوکس در محیطهای اداری، پذیرفتن ریسک نیست. از نظر قابلیت اطمینان و پایداری و همچنین حفاظت در برابر انواع ویروسها چیزی بهتر از لینوکس وجود ندارد. با تلاش شرکتهای بزرگی مانند ردهت استفاده از لینوکس در محیطهای تجاری توسعه فراوان یافته و اکنون تعداد زیادی از شرکتهای کوچک و بزرگ در حال استفاده از سرویس دهنده ها و ایستگاههای کاری مبتنی بر لینوکس هستند.



۳-۱-۴ طلوع لینوکس روی میزی (Desktop Linux)

بزرگترین ایرادی که از لینوکس گرفته میشد چه بود؟ قبلاً محیط تمام متنی لینوکس، بسیاری از کاربران را از استفاده کردن از آن بر حذر میداشت. با اینکه در استفاده از محیط متنی کنترل کامل سیستم در اختیار شماست، ولی این محیط اصلاً برای کاربران عادی سیستمهای کامپیوتری مناسب نیست. محیطهای گرافیکی که بر پایه X-Window وجود داشتند نیز پاسخگوی امکاناتی که سیستمعاملهای گرافیکی مانند ویندوز برای کاربران خود ارائه میکردند، نبودند. ولی از چند سال گذشته این وضعیت در حال تغییر بوده است. اکنون محیطهای گرافیکی حرفه‌ای مانند KDE (K Desktop Environment) و GNOME (GNU Network Object Model Environment) تصویر لینوکس را کامل کرده‌اند. این محیطهای گرافیکی اکنون بسیار کاربر پسند و قدرتمند شده‌اند و وجود این سیستمهاست که امروزه کاربران عادی نیز میتوانند از لینوکس استفاده کنند.

۳-۱-۵ لینوکس در جهان سوم

ورود لینوکس به کشورهای جهان سوم تحولی ایجاد نموده است. قبل از وجود لینوکس کشورهای جهان سومی در زمینه کامپیوتر در سطح بسیار پایین تری قرار داشتند. هزینه سخت افزارها بسیار پایین آمده بود ولی هزینه نرم افزار برای این گونه کشورها همچنان کمر شکن بود. این امر باعث شد تا در بسیاری از این کشورها کپی غیر مجاز نرم افزارها گسترش پیدا کند که باعث میلیاردها دلار خسارت سالیانه میشود. یکی از عمده ترین دلایل این کار پایین بودن درآمد سرانه در این کشورهاست. هنگامی که مجموع درآمد سرانه سالیانه بیش از ۲۰۰ تا ۳۰۰ دلار نیست، هیچگاه امکان خرید یک سیستم عامل ۱۰۰ دلاری وجود نخواهد داشت.

طلوع لینوکس و سایر تولیدات باز متن، این وضعیت را تغییر داده است. این امکان وجود دارد تا بتوان لینوکس را در کامپیوترهای قدیمی ۴۸۶ و پنتیوم که اکنون در کشورهای توسعه یافته به تاریخ پیوسته‌اند ولی هنوز در کشورهای در حال توسعه از آنها استفاده میشود، اجرا نمود. همچنین استفاده از نرم افزارهای رایگان باز متن گسترش یافته تا جلوی هزینه های سرسام آور نرم افزاری این کشورها را بگیرد. امروزه در کشورهای آسیایی، آفریقایی و آمریکای لاتین استفاده از لینوکس و نرم افزارهای باز متن گسترش فراوانی یافته و با استفاده از خصلت ذاتی تغییر پذیری لینوکس، برای استفاده از زبانهای ملی این کشورها سفارشی شده است. امروزه مستندات لینوکس به اکثر زبانهای زنده جهان ترجمه شده‌اند.

۳-۱-۶ از میز کار تا ابر کامپیوترها

هنگامی که توروالدز لینوکس را ایجاد نمود، این مخلوق جدید، فقط یک اسباب بازی تازه برای هکرها بود. ولی از زمان دستگاههای ۳۸۶ که نخستین هسته لینوکس بر روی آنها اجرا میشد، لینوکس راه درازی را طی نموده است. یکی از مهمترین استفاده های امروزی لینوکس استفاده از آن در پردازشهای سنگین موازی در ابر کامپیوترهاست. امروزه اکثر ابر کامپیوترهایی که در جهان ساخته میشوند، از لینوکس به عنوان سیستم عامل خود استفاده میکنند.

۳-۱-۷ داستان ادامه دارد

حرکت لینوکس از یک پروژه هکری تا جهانی شدن یک انقلاب شگفت انگیز است. پروژه GNU که در اوایل دهه ۱۹۸۰ توسط ریچارد استالمن شروع شد، توسعه نرم افزارهای باز متن را رهبری نمود. پرفسور اندرو تاننباوم و سیستم عامل MINIX او مطالعه سیستم عامل ها را از حالت تئوری به عملی تبدیل نمود و در نهایت همت و تلاش توروالدز منجر به تولد لینوکس شد. امروزه لینوکس دیگر یک پروژه هکری به شما نمی رود بلکه یک حرکت جهانی است که توسط میلیونها نفر برنامه نویس باز متن و شرکتهای بزرگی مانند IBM حمایت میشود. لینوکس در تاریخ کامپیوتر به عنوان یکی از شگفت انگیز ترین محصولات تلاش بشری باقی خواهد ماند.



۳-۱-۸ نشان لینوکس

نشان لینوکس یک پنگوئن است. برخلاف سایر سیستم عاملهای تجاری، این نشان زیاد جدی نیست! توکس نشانگر وضعیت بدون نگرانی حرکت لینوکس است. این نشان تاریخچه بسیار جالبی دارد. لینوکس در ابتدا فاقد هر گونه نشانی بود. هنگامی که توروالدز برای تعطیلات به استرالیا رفته بود، در دیداری که از یک باغ وحش داشت، هنگامی که می خواست با یک پنگوئن بازی کند، پنگوئن دست وی را گاز گرفت و همین ایده ای شد تا از پنگوئن به عنوان نشان لینوکس استفاده شود.

۳-۲ Free Ware یا برنامه آزاد یعنی چه؟

نرم افزاری را آزاد می گویند که دارای شرایط زیر باشد:

۱. کاربر این نوع برنامه می تواند برنامه را برای هر هدفی اجرا کند.
 ۲. کاربر این نوع برنامه می تواند برنامه را بنا به نیازهای خود تغییر دهد برای این کار کاربر باید کد برنامه را در اختیار داشته باشد.
 ۳. کاربر این نوع برنامه می تواند برنامه را تکثیر کند و آن را رایگان و یا در مقابل پول در اختیار دیگران قرار دهد اما در هر صورت باید کد برنامه نیز منتشر شود.
 ۴. کاربر آزاد است که نسخه های تصحیح شده برنامه را منتشر کند.
- برای تضمین کردن شرایطی که یک Free Ware باید داشته باشد قانونی به نام Copy Left تهیه شده است که دقیقا در مقابل Copr Right قرار دارد و اجازه ی تملیکی شدن نرم افزارها را نمی دهد.
- در سال ۱۹۹۸، عده ای که عقیده داشتند واژه نرم افزار آزاد مناسب نمی باشد، شروع به استفاده از واژه "باز متن" (Open Source) کردند. تفاوت این گروه با طرفداران پروژه گنو و نرم افزار آزاد در این است که به عقیده این گروه نرم افزار انحصاری اشکالی ندارد، اما نرم افزار متن باز صرفا بهتر است، حال آنکه به عقیده طرفداران نرم افزار آزاد، نرم افزار انحصاری غلط است و باعث عقب ماندن جامعه می گردد. گروهی نیز به تازگی برای آنکه خود را به هیچ یک از این کمپها نسبت ندهند، از عبارت Free/Libre Open Source Software یا FLOSS استفاده می کنند که مجموعه ای از همه واژه های توصیف کننده نرم افزار آزاد می باشد.

۳-۳ کدام توزیع گنو/لینوکس را انتخاب کنیم؟



یکی از سوالاتی که توسط کاربرانی که مایل به انتقال به سیستم عامل گنو/لینوکس هستند مطرح می شود، انتخاب توزیع است. چرا انواع مختلفی از گنو/لینوکس وجود دارد؟ کدامیک مناسب تر است؟ در ادامه به معرفی اصول و مفاهیم توزیع های گنو/لینوکس و معرفی برخی از آنها می پردازیم و به مقایسه توزیع های مختلف خواهیم پرداخت.

۳-۳-۱ توزیع چیست؟

سیستم عامل گنو/لینوکس از بخش‌های بسیار زیادی تشکیل شده که هر بخش آن توسط عده‌ای خاص توسعه می‌یابد که هر یک در یک سمت جهان قرار دارند. معروف است که می‌گویند گنو/لینوکس مانند هواپیمایی است که هر قسمت آن را در یک کشور ساخته‌اند. البته این نکته نقطه قوت آن به شمار می‌رود. در صورتی که شما به عنوان یک کاربر بخواهید یک گنو/لینوکس داشته باشید، باید تمام این قطعات را جداگانه جمع‌آوری کرده و پس از کامپایل استفاده نمایید. درصد کمی از مردم این امکان و توانایی را دارند. بنابراین افراد و شرکت‌هایی اقدام به جمع‌آوری این قطعات مجزا و قرار دادن آنها کنار هم کرده‌اند و علاوه بر آن برای این مجموعه برنامه‌های نصب و مدیریت نوشته‌اند تا کار نصب و مدیریت سیستم را برای کاربران آسان کنند. به این مجموعه‌ها که توسط افراد و شرکت‌ها گردآوری شده است، توزیع یا Distribution گنو/لینوکس می‌گویند.

۳-۳-۲ علت تنوع توزیع‌ها چیست؟

هر یک از توزیع‌های گنو/لینوکس دارای ویژگی‌های خاصی است که آنرا از توزیع دیگر متمایز می‌کند. مثلاً ممکن است برنامه‌های نصب آنها با هم تفاوت داشته باشند (البته اصول نصب همه گنو/لینوکس‌ها یکسان است) و یا ابزارهای مدیریتی گرافیکی تهیه شده با هم متفاوت باشند و یا نسخه برنامه‌هایی که با یک توزیع خاص ارائه می‌شوند جدیدتر یا قدیمی‌تر باشند، محل فایل‌های پیکربندی آنها متفاوت باشد، و یا ممکن است توزیع‌هایی مخصوص امور خاص طراحی شده باشند. مثلاً مخصوص سرویس‌دهنده‌ها، مخصوص ایستگاه‌های کاری، مخصوص کامپیوترهای قدیمی، مخصوص مدیریت شبکه، مخصوص چند رسانه‌ای، مخصوص بازی و به همین ترتیب. توزیع‌ها به دو صورت تجاری و رایگان ارائه می‌شوند.

۳-۳-۳ انتخاب توزیع؟

همانطوری که در بالا اشاره شد، هر یک از توزیع‌ها دارای ویژگی‌های خاص خود هستند. مثلاً برنامه نصب یک توزیع بسیار راحت است و یک توزیع دیگر ممکن است از نظر پایداری و امنیت مطرح باشد. انتخاب توزیع بستگی به شرایط زیر دارد:

۱. سطح علمی کاربر
۲. مورد استفاده از گنو/لینوکس
۳. ویژگی‌های توزیع
۴. بازار

اکنون به بررسی یکایک این شرایط می‌پردازیم.

الف: سطح علمی کاربر: کاربرانی که دارای آشنایی کمتری با گنو/لینوکس هستند، جذب توزیع‌هایی می‌شوند که دارای ابزارهای پیکربندی گرافیکی است که آنها را قادر می‌سازد راحت‌تر سیستم‌شان را اداره و نصب نمایند. همچنین دارای نرم‌افزارهای جدیدی باشد که به آنها حداکثر قابلیت‌ها را ارائه نماید. از توزیع‌هایی که برای کاربران تازه کار مناسب‌تر هستند، می‌توان اوبونتو (Ubuntu)، سوزی (SuSE)، فدورا (Fedora)، مندریو (Mandriva)، لیندوز (Lindows)، لیکوریس (Lycoris)، میپس (Mepis) و XandarOS را نام برد. کاربرانی که پیشرفته‌تر هستند و ابزارهای پیکربندی گرافیکی برایشان مهم نبوده، کیفیت و سرعت سیستم برایشان مهم‌تر است جذب توزیع‌های حرفه‌ای‌تر مانند دبیان (Debian)، جنّتو (Gentoo) و اسلکور (Slackware) می‌شوند. ممکن است نصب و راه‌اندازی این توزیع‌ها برای کاربران تازه‌کار دشوار باشد، ولی در عوض هر سه آنها بسیار باکیفیت و پایدار هستند.

ب: مورد استفاده از گنو/لینوکس: برخی از توزیع‌ها مخصوص نیازهای خاصی طراحی شده‌اند. مثلاً امروزه از اصلی‌ترین نیازها می‌توان به سرویس‌دهنده‌ها و ایستگاه‌های کاری اشاره نمود. البته برخی از توزیع‌ها این امکان را به شما می‌دهند که هنگام نصب، نوع مصرف آنها را

تعیین کنید و با توجه به انتخاب شما، نرم افزارهای مربوط به آن مصرف خاص نصب خواهند شد. مثلاً اوبونتو، فدورا و دبیان این امکان را دارا هستند. برخی از توزیع ها تنها مخصوص یک نیاز طراحی شده اند و دارای ابزارهای مربوط به آن نیاز می باشند. مثلاً گنو/لینوکس ناپیکس (Knoppix) که یک توزیع روی میزی است، تنها دارای ابزارهایی است که برای کاربران روی میزی کاربرد دارد.

ج: ویژگی های توزیع: برخی اوقات یک توزیع دارای ویژگی های خاصی است که آنرا برای استفاده قابل انتخاب می سازد. مثلاً گنو/لینوکس اورالوکس (Oralux) دارای امکانات مخصوص نابینایان می باشد. مانند شناسایی صفحه نمایش های بریل و یا مرور صوتی وب و پست الکترونیک. و یا یک گنو/لینوکس ممکن است سخت افزارهای خاصی را به خوبی پشتیبانی نماید. و یا ممکن است سرعت و کیفیت یک توزیع یا آسانی استفاده از آن ملاک انتخاب قرار گیرد.

د: بازار: ممکن است موجود بودن یک توزیع در بازار و یا نبود آن ملاک انتخاب باشد. مثلاً در ایران فراوان ترین توزیع گنو/لینوکس، توزیع فدورا و ردهت است. اصلاً برخی افراد و موسسات گنو/لینوکس را به نام ردهت می شناسند.

به دلیل اینکه Linux بر گرفته و تجربه یک ربع قرن کار روی UNIX است، پس بهتر است ابتدا با ویژگیها و ساختار یونیکس آشنا شویم:

۳-۴ ویژگیهای سیستم عامل یونیکس (UNIX)

۱- **Multi programming:** چند برنامه در آن واحد روی حافظه اصلی resident شده اند و پردازنده ها بین آنها سوئیچ می کنند که باعث افزایش بازدهی سیستم می شود.

۲- **Time Sharing:** در این روش پردازنده مرکزی جسمی است که توسط تمام کاربران و پردازنده ها استفاده می شود. به این ترتیب که زمان CPU بین پردازنده های مختلف سیستم و کاربران تقسیم می شود و به هر یک کسری از زمان CPU تعلق می گیرد.

۳- **Multi User:** چند کاربره بودن.

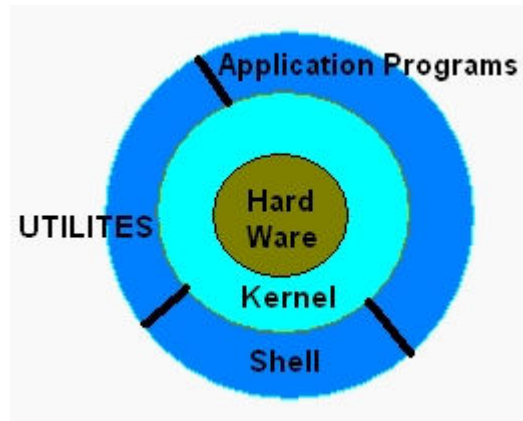
۴- **Multi Tasking:** امکان اجرای برنامه بصورت Foreground و Background

۵- **File System:** دارا بودن File system بصورت سلسله مراتبی و تأمین امنیت برای داده ای سیستم.

۶- **Generality:** بدین معنی که یک روش واحد بتواند اهداف چندگانه ای را برآورده سازد.

۷- **Transportability (قابلیت حمل):** سیستم عامل UNIX به آسانی قابل حمل است. برای یک سیستم جدید، با دستکاری اندک بر روی کدهای سیستم عامل UNIX می توان آنرا جهت نصب آماده نمود. توانایی حمل سیستم عامل UNIX از یک نوع کامپیوتر به نوع دیگر دلیل اصلی موفقیت آن می باشد.

۳-۵ معماری سیستم عامل لینوکس (ARCHITECTURE OF LINUX SYSTEM)



KERNEL

هسته اصلی سیستم عامل لینوکس kernel می باشد. kernel قسمتی از سیستم که حافظه، فایل ها و وسایل جانبی را مدیریت میکند. زمان و تاریخ را حفظ میکند برنامه های کاربردی را آغاز میکند و منابع سیستمی را تخصیص می دهد kernel مستقیماً با سخت افزار در ارتباط است

SHELL

shell نقش رابط بین کاربر و kernel را بر عهده دارد. shell یک برنامه مفید می باشد که دستورات و فرمانها را از کاربر دریافت میکند و بعد از ترجمه آن را جهت اجرا به kernel تحویل میدهد

UTILITES LINUX

لینوکس utilites یا فرمانها یک مجموعه در حدود ۲۰۰ برنامه که فرآیند لازم را برعهده دارند. این برنامه ها از طریق shell درخواست میشوند

APPLICATION SOFTWARE

برنامه هایی نظیر نرم افزار حسابداری و سیستمهای مدیریتی و data base می باشند

۳-۶ ساختار سیستم عامل یونیکس (UNIX)

سیستم عامل UNIX شامل چهار بخش اصلی زیر است:

Commands-۴

Shell-۳

File System-۲

Kernel-۱

۳-۷ ساختار سیستم فایل (File System) در یونیکس (UNIX)

فایل سیستم ساختمانی برای ذخیره و بازیابی اطلاعات ارائه می دهد. هر فایل سیستم دیسک را به چهار منطقه تقسیم می کند که عبارتند از:

- ۱- **Boot Block**: اولین بلاک حافظه جانبی که برای سیستم رزرو می شود و حاوی اطلاعات لازم جهت راه اندازی سیستم می باشد.
- ۲- **Super Block**: بلاک شماره ۱ را سوپر بلاک می نامند. این بلاک شامل مجموعه اطلاعاتی است که وضعیت فایل سیستم را مشخص می کند. این اطلاعات عبارتند از:

- 👉👉 سایز فایل سیستم
- 👉👉 تعداد **Inode** های موجود
- 👉👉 آدرس اولین بلاک حاوی اطلاعات
- 👉👉 تعداد کل بلاکهای موجود
- 👉👉 تعداد بلاکهای آزاد
- 👉👉 تعداد **Inode** های آزاد
- 👉👉 زمان آخرین بروزرسانی فایل سیستم
- 👉👉 اینکه آیا یک فایل سیستم درست **Close** شده است یا نه
- 👉👉 ویرایش و نوع فایل سیستم
- 👉👉 سایز هر بلاک

- ۳- **list**: یک ناحیه شامل یک لیست پیوندی از **inode** ها می باشد. **Inode** ساختاری بطول ۶۴ بایت می باشد و شماره آن از ۱ آغاز می شود و حاوی اطلاعات زیر می باشد:

- 👉👉 شماره کاربر فایل (**UID**)
- 👉👉 شماره گروه فایل (**GID**)
- 👉👉 آدرس فیزیکی محتوای فایل روی دیسک
- 👉👉 سایز فایل
- 👉👉 زمان ایجاد فایل
- 👉👉 نوع فایل
- 👉👉 زمان آخرین دستیابی به فایل
- 👉👉 زمان آخرین تغییر روی فایل
- 👉👉 مجوزهای فایل
- 👉👉 تعداد لینک های فایل

نکته: **inode** شماره ۱ برای سیستم رزرو شده است و **Inode** شماره ۲ مربوط به فهرست ریشه می باشد.

- ۴- **Data**: فضای آزاد باقی مانده روی دیسک بصورت یک لیست پیوندی از بلاکهای در دسترس دیسک نگهداری می شود که برای ذخیره داده ها مورد استفاده قرار می گیرد.

نکته: سیستم عامل **UNIX** حداقل یک فایل سیستم روی دیسک سخت اولیه اش دارد. این فایل سیستم **root** نام دارد و با علامت **/** مشخص می شود. فایل سیستم **root** شامل برنامه ها و دایرکتوری هایی است که توسط سیستم عامل ایجاد می شود.

نکته: نگهداری اطلاعات در فایل سیستم از وظایف سیستم عامل است. از دست دادن اطلاعات اتفاق نادری است زیرا فایل سیستم در برابر تخریب اطلاعات بسیار مقاوم است. سیستم عامل **UNIX** از برنامه **FSCK** برای تعمیر فایل سیستم خراب شده استفاده می کند. **FSCK** در هنگام بوت، بطور خودکار فایل سیستم **root** را چک می کند. هنگامی که سیستم بطور غیرمعمول **terminate** شده باشد، **FSCK** برای تمیز کردن فایل سیستم از مدیر سؤال می کند و در صورت تأیید او، فایل سیستم تمیز می گردد.

۳-۸ مفهوم سیستم فایل در یونیکس (UNIX) و لینوکس (Linux)

اگر بخواهیم یک توصیف ساده در مورد سیستمهای یونیکس و مشابه آن یعنی لینوکس ارائه کنیم باید بگوئیم : در یک سیستم مبتنی بر یونیکس هر چیزی یک فایل است یا یک فایل نماینده خود را دارد و اگر چیزی یک فایل نباشد حتما یک روند یا process است .

این توصیف برآستی حقیقت دارد چرا که در یونیکس فایلها ی مخصوصی وجود دارند که وظایفی بیشتر از یک فایل بر عهده دارند (برای مثال pipe ها و socket ها).

در یک سیستم گنو/لینوکس درست مثل یونیکس هیچ تفاوتی بین فایل و دایرکتوری وجود ندارد ، به این معنی که یک دایرکتوری هم خود یک فایل است که حاوی اسامی تعدادی فایل یا دایرکتوری دیگر است .

برنامه ها ، سرویسها ، متنها و تصاویر و سایر موارد مشابه و همچنین دستگاههای ورودی و خروجی input and output devices و عموما همه ابزارهای سیستمی همگی با یک فایل به سیستم معرفی میشوند .

حال با این تفکر اگر بخواهیم همگی این فایلها ی متنوع را در یک ساختار منطقی مرتب کرده و نگهداری کنیم باید به آنها بصورت یک ساختار درختی (شکل صفحه بعد) نگاه کنیم که روی هارد دیسک استقرار یافته است (مثل سیستم عامل DOS). این ساختار درختی از یک ریشه اصلی یا ROOT تشکیل شده و شاخه های بزرگتر منشعب شده از آن دارای زیر شاخه و شاخه های انتهائی دارای برگهائی هستند که همان فایلها ی ما میباشند.

هارد دیسک به پارتیشن تقسیم میشود و پارتیشنها برای ذخیره اطلاعات توسط کاربران با سیستم فایلها فرمت میشوند

سیستم فایل پیشفرض برای لینوکس -ext3 third extended linux file system می باشد. دیگر سیستم فایلها ext2 و msdos که برای فلاپیها مورد استفاده قرار میگیرند هستند. همچنین iso 9660 برای cd ها مورد استفاده قرار میگیرد.

۳-۹ انواع فایلها در لینوکس

۱ - فایلها ی معمولی regular files : بیشتر فایلها فقط فایل معمولی هستند که حاوی اطلاعات معمولی مثل متن ، کدهای اجرایی برنامه ها ، خروجی و ورودی برنامه های کاربردی دیگر میباشند .

۲ - فایلها ی اختصاصی :

دایرکتوری : فایلی حاوی اسامی دیگر فایلها

فایلها ی ویژه : حاوی مکانیزمهای چگونگی ورود و خروج اطلاعات به کامپیوتر (همه فایلها ی موجود در مسیر /dev از این دسته

هستند .

لینکها : سیستمی را فراهم میکنند که یک فایل یا دایرکتوری در نقاط مختلف ساختار درختی بدون نیاز به تکرار محتوی قابل مشاهده

و در دسترس باشند .

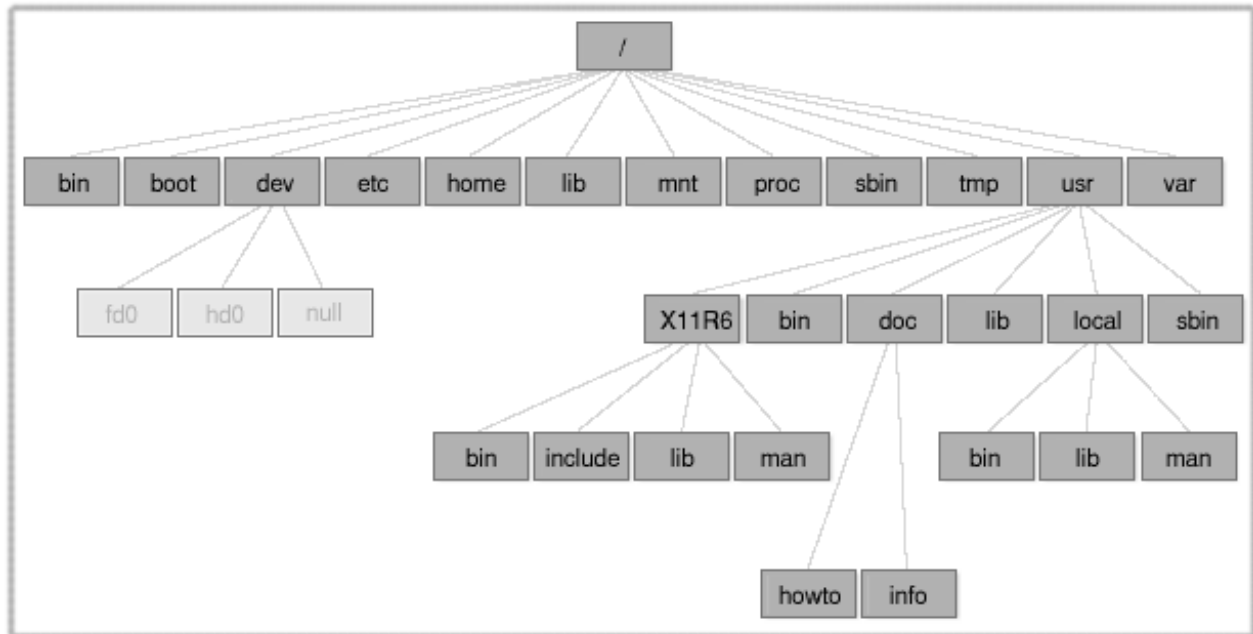
Domain یا sockets : نوع بخصوصی از فایلها که همانند سوکتها ی tcp/ip روندهائی را که برای کنترل دسترسی به فایل

سیستم در شبکه بصورت کنترل شده فراهم میکنند.

Named piped : کمابیش عملی همانند سوکتها انجام میدهند ولی متفاوت با روالی که سوکتها در پیش میگیرند.

Symbol	Meaning
-	Regular file
d	Directory
l	Link
c	Special file
s	Socket
p	Named pipe

سیستم فایل لینوکس ساختاری است که اطلاعات شما را در کامپیوتر ذخیره میکند. فایلها در یک ساختار درختی از دایرکتوری ها ذخیره میشوند. هر دایرکتوری میتواند حاوی فایلها و دایرکتوری های دیگری باشد. در صورتی که بخواهید ساختار سیستم فایل لینوکس را دقیق تر توصیف کنید، آن بیشتر شبیه یک درخت وارونه است. در بالاترین نقطه، دایرکتوری ریشه قرار دارد که بوسیله یک اسلش تنها نشان داده میشود. در زیر آن دایرکتوری های عمومی و سیستمی سیستم عامل لینوکس قرار میگیرند. مانند `home`، `dev`، `bin` و `tmp`. هر کدام از این دایرکتوری ها محتوی دایرکتوری های دیگری هستند. تصویر زیر ساختار درختی سیستم فایل لینوکس را نشان میدهد.



شکل ۳-۱: ساختار سیستم فایل لینوکس

برخی از دایرکتوری های مهم سیستم فایل لینوکس در زیر توضیح داده شده اند.

۳-۱۰ معرفی دایرکتوری های سطح بالای سیستم فایل های رایج در لینوکس

- `/`: این دایرکتوری Root یا ریشه مبنای سیستم فایل لینوکس است. از نظر منطقی کلیه دایرکتوری ها و فایل های لینوکس صرف نظر از محل فیزیکی آنان در این دایرکتوری قرار دارند.
- `/bin`: این دایرکتوری شامل برنامه های اجرایی است که جزیی از سیستم عامل لینوکس هستند. بسیاری از فرمان های لینوکس مانند `cat`, `cp`, `ls`, `more`, `tar` و... در این دایرکتوری قرار دارند.
- `/boot`: این دایرکتوری حاوی هسته لینوکس و دیگر فایل هایی است که مورد نیاز برنامه GRUB boot Manager است. (این هسته و فایل های دیگر می توانند در هر دایرکتوری نیز قرار داد ولی به طور مرسوم آنها را در این دایرکتوری قرار می دهند.)
- `/dev`: این دایرکتوری شامل فایل های ویژه جهت نمایش وسایل متصل شده به سیستم هستند.
- `/etc`: این دایرکتوری حاوی بیشترین پرونده های پیکربندی سیستم و اسکریپت های `iniliations` است.
- `/home`: محل قراردادی دایرکتوری خانگی کلیه کاربران. مانند دایرکتوری خانگی `/home/Roohollah` برای کاربری به نام Roohollah.
- `/lib`: دایرکتوری حاوی فایل های کتابخانه برای کلیه برنامه های ذخیره شده در دایرکتوری های `/bin` و `/sbin` از جمله ماژول های قابل بارگذاری درایو که مورد نیاز راه اندازی لینوکس است.

- **/media** : یک دایرکتوری برای سوار کردن سیستم های فایل روی رسانه های قابل حمل مانند درایوهای CD-ROM ، دیسک های فلاپی و درایوهای zip. این دایرکتوری شامل دایرکتوری **/media/floppy** برای سوار کردن دیسک های فلاپی و دایرکتوری **/media/cdrom** برای سوار کردن سی دی رام است. اگر یک ضبط کننده CD داشته باشید بجای دایرکتوری آن دایرکتوری **/media/cdrecorder** در اختیار شما گذاشته می شود.
- **/mnt** : یک دایرکتوری برای سیستم های فایل سوار شده موقتی
- **/opt** : این دایرکتوری یک ناحیه ذخیره سازی برای بسته های نرم افزار برنامه های بزرگ کاربردی ارائه می دهد. مانند برنامه های کاربردی **Gnome , KDE** نصب شده در این دایرکتوری.
- **/proc** : یک دایرکتوری خاص ساکن در حافظه حاوی اطلاعات مختلف درباره کارهای در حال اجرا در سیستم لینوکس.
- **/root** : دایرکتوری خانگی برای کاربر ریشه (دقت کنید با دایرکتوری ریشه تفاوت دارد)
- **/sbin** : حاوی فایل های اجرایی معرف فرمان هایی که بطور نوعی برای کارهای سرپرستی سیستم استفاده میشود و توسط کاربر **root** بکار می رود. فرمان هایی مانند **shutdown , Halt** در دایرکتوری **/sbin** قرار می گیرند.
- **/srv** : این دایرکتوری شامل اطلاعاتی برای خدمات مانند **Web** و **FTP** ارائه شده در این سیستم است.
- **/sys** : یک دایرکتوری خاص حاوی اطلاعات درباره وسایل کامپیوتر که هسته لینوکس با آن ها سر و کار دارد.
- **/tmp** : یک دایرکتوری موقت قابل استفاده برای هر کاربر لینوکس به عنوان یک دایرکتوری **scratch** به این معنا که محتویات آن حائز اهمیت نبوده و هر بار با راه اندازی سیستم محتویات آن پاک می شود.
- **/usr** : دایرکتوری حاوی زیر دایرکتوری ها برای بسیاری از برنامه های مهم مانند **X Window System** که در دایرکتوری **usr/x11R6** و کتابچه راهنمای **Online**.
- **/var** : این دایرکتوری حاوی فایل های مختلف سیستم مانند فایل های گزارش های سیستمی و نیز دایرکتوری هایی برای نگه داری سایر اطلاعات مانند فایل هایی برای چاپ و پیام های الکتریکی است.

۳-۱۱ LILO و GRUB و فرایند راه اندازی لینوکس

در لینوکس دو مدیر بوت رایج وجود دارد. لیلو (LILO=Linux Loader) مدیر بوت سنتی لینوکس و گراب (Grub=Grand Unified Unix Bootloader) مدیر بوتی جدیدتر می باشد. هر کدام از این برنامه ها ابتدا مقداری اطلاعات پیکربندی را دریافت کرده و سپس هسته لینوکس یا سیستم عامل دیگری را بارگذاری می کند و ادامه فرایند بوت را به آن می سپارد.

بعد از اینکه بایوس کنترل را به بوت لودر سپرد ، بوت لودر هم کنترل را در نهایت به سیستم عامل شما مثلا به لینوکس می سپارد . بدیهی است که بوت لودرهای لینوکس براحتی قابل پیکربندی و اختصاصی شدن هستند و اطلاعات نخستین سیستم عامل قابل بوت و زمان وقفه و پارتیشن هارد دیسک محل استقرار (Master Boot Record) MBR یا سیستم عامل ها با ویرایش فایل های **/etc/lilo.conf** یا **/boot/grub/menu.lst** قابل تغییر است .

اولین چیزی که لینوکس پس از آغاز عملیات بوت انجام میدهد تغییر وضعیت سیستم یا سوئیچ کردن به وضعیت **protected mode** یا همان سیستم امن میباشد . البته باید دانست که سیستم های عامل قدیمی همچون داس بدون تغییر وضعیت سیستم از لحاظ امنیتی به وضعیت بدون حفاظ **real mode** بوت میشوند .

همانطوریکه می بینید سیستم های عامل پیشرفته مثل لینوکس با انجام اینکار کنترل سیستم را کاملا بدست گرفته و از اختیار بایوس کاملا خارج می سازند .

در مرحله بعدی لینوکس به جستجو و شناسائی سخت افزارهای موجود بر روی سیستم می پردازد و این کار را هر بار که شما سیستم خود را خاموش و سپس روشن نمائید انجام خواهد داد چرا که اطلاعات مربوط به سخت افزار سیستم مثل نوع مادربرد ، هاردیسک ، چیپست کارت گرافیک ، ماوس و ابزارهای شبکه و در هر بار بوت شدن لازم و ضروری اند و لینوکس نمی تواند و نباید این اطلاعات را بخاطر بسپارد چون هر کس ممکن است در طول دومرحله بوت کامپیوتر خود نوعی از سخت افزار را به آن افزوده یا از آن بکاهد .

۳-۱۲ محیط X Windows چیست؟

معمولا برای راحتی تلفظ، به این محیط، X نیز گفته میشود که بر پایه GUI بنا نهاده شده و یک محیط گرافیکی قوی برای راحتی کار در Linux است. در این محیط شما می توانید چندین پنجره Terminal را به طور همزمان در یک صفحه داشته باشید، به طوری که در هر پنجره، یک کاربر Login کرده باشد. معمولا در محیط X و هر محیط گرافیکی دیگر، میتوان از ماوس برای راحتی بیشتر استفاده کرد. خیلی از برنامه ها مانند بازیها و نرم افزارهای گرافیکی و کاربردی برای محیط X نوشته شده است. Linux از دو محیط متنی و گرافیکی تشکیل شده است. معمولا محیط متنی به دلیل کاربر پسند نبودن مورد توجه و پسند کاربران مبتدی قرار نمی گیرد، ولی این محیط برای کاربران حرفه ای تر که از Linux برای مصارف شبکه استفاده می کنند، رضایت بخش است، زیرا دوام سرویسهای نصب شده و در حال استفاده در محیط متنی، بسیار بیشتر از دوام سرویسها در محیط گرافیکی است.

همراه با لینوکس دو محیط گرافیکی با نامهای GNOME و KDE ارائه می شود. همراه با این محیطها ابزارهایی برای تنظیم گزینه هایی مانند رنگها، شکل پنجره ها، ماوس، تصویر پس زمینه، منوها و... وجود دارد. به طوری که هر کاربر می تواند تنظیمات جداگانه خود را از سایر کاربران داشته باشد.

هنگامی که فرایند ورود به سیستم خاتمه یافت، محیط گرافیکی به طور خودکار اجرا می شود.

یکی از محیطهای گرافیکی Linux که محیطی راحت و دوست داشتنی را برای کاربران خود فراهم میکند، KDE نام دارد. این محیط بسیار شبیه دسکتاپ در ویندوز است. البته بر خلاف ویندوز، Linux دارای چندین دسکتاپ در یک محیط گرافیکی است. به این نکته باید توجه کرد که سرعت محیطهای گرافیکی Linux، در حد ویندوز یا حتی کمی کندتر از آن است، زیرا GUI در ویندوز به صورت ذاتی به همراه سیستم عامل است ولی در لینوکس محیطهای گرافیکی به عنوان برنامه هایی جدا از سیستم عامل هستند و گاهی بر روی لینوکس بار اضافی ایجاد می کنند.

۳-۱۳ پروژه KDE و GNOME

پروژه GNOME (GNU Network Object Model Environment) دو امکان را فراهم می کند:

محیط میز کار GNOME، یک دسکتاپ ویژوال و جالب برای کاربران مختلف،

و دوم زیرساخت های برنامه سازی، به معنی چارچوبی گسترده برای ساخت کاربردهایی که با سایر برنامه های دسکتاپ یکپارچه می شوند. برخی از خصوصیات GNOME عبارتند از:

قابل استفاده: قابل استفاده بودن در واقع به مفهوم ایجاد نرم افزار نیست که برای همگان چه کاربران و چه توسعه دهندگان نرم افزار، قابل استفاده باشد. این نکته همواره مورد توجه خاص جامعه GNOME می باشد.

دسترسی آسان و بین المللی: GNOME به زبانهای مختلف توسعه یافته، مستند سازی شده و مورد استفاده قرار می گیرد. گروه GNOME در تلاش است این اطمینان را بدهد که تمامی بخش های نرم افزار را بتوان به کلیه زبان ها ترجمه کرد.

مورد پسند برنامه نویسان (Developer-Friendly): توسعه دهندگان نرم افزار برای استفاده از GNOME تنها از یک زبان استفاده نمی کنند، شما می توانید از C، C++، Python، Perl، Java و یا حتی C# برای ایجاد کاربردهای سطح بالایی که با سایر برنامه های دسکتاپ یونیکس و لینوکس یکپارچه می شوند، استفاده کنید.

سازمان یافتگی: GNOME تلاش می کند که یک انجمن سازمان یافته با ساختاری متشکل از صداها عضو و تشکیلات منظم باشد. توزیع های GNOME توسط تیم مخصوص این کار و طبق برنامه هر ۶ ماه یکبار پخش می شود.

پشتیبانی: GNOME توسط شرکت های با نفوذ در لینوکس و یونیکس پشتیبانی می شود، برخی از این شرکت ها عبارتند از: HP، Sun، Canonical، RedHat، Novell، MandrakeSoft و شرکت Sun.

پروژه KDE (K Desktop Environment) در اواخر سال ۱۹۹۶ به وجود آمد. اهداف KDE عبارت بودند از آماده سازی بستری مناسب برای نوشتن نرم افزار برای یونیکس و گنو/لینوکس؛ و فراهم کردن محیط گرافیکی جذاب برای ایستگاههای کاری یونیکس و گنو/لینوکس.

KDE در ابتدا یک پروژه عمدتاً آلمانی بود، که به مرور زمان گسترش یافت و امروزه بدل به شبکه‌ای از مهندسان نرم‌افزار معتقد به نرم‌افزار آزاد در سراسر جهان شده است. KDE که مخفف "K Desktop Environment" می‌باشد، با پیشرفت خیره‌کننده‌اش طی سال‌های اخیر، تبدیل به تاج طلایی پروژه‌های نرم‌افزار آزاد گردیده است، به گونه‌ای که یکی از جدیدترین نسخه‌های آن یعنی نسخه ۱/۳ هم از نظر زیبایی و چشم‌نوازی، و هم از نظر قابلیت و کارایی با محیط‌های ویندوز و مکینتاش رقابت می‌کند، و به اعتقاد بسیاری، حتی از آن‌ها پیشی گرفته است.

محیط میزکار KDE شامل برنامه‌ها و نرم‌افزارهای گوناگون و متنوعی می‌باشد. نرم‌افزارهای شبکه، برنامه FTP، پست الکترونیکی (KMail) و گفتگوی مستقیم (Kopete)؛ نرم‌افزارهای گرافیکی و ویرایش تصویر (Kontour)، پخش موسیقی (noatun)، پخش فیلم (aktion!) و کار با دوربین‌های دیجیتال (Kamera) و پویشگر (Kooka) و چاپگر (Kdeprint)، نرم‌افزارهای مدیریت سیستم همانند مرکز کنترل، مرورگر صفحات وب (Konqueror)، نرم‌افزارهای نمابر (KFax)، طراحی صفحات وب (Quanta)، محیط برنامه‌نویسی (KDevelop)، تعدادی بازی (Kdegames)، نرم‌افزارهای آموزشی (Kdeedu)، و یک مجموعه کامل نرم‌افزارهای اداری (KOffice) شامل واژه‌پرداز (KWord)، صفحه‌گسترده (KSpread)، نمایش KPresenter و بسیاری نرم‌افزارهای دیگر، محیطی کامل را برای کاربران فراهم کرده‌اند.

با توجه به سرعت پیشرفتی که KDE از خود نشان داده است، پیش‌بینی آینده آن کاری بس دشوار است. به تازگی دولت آلمان طی پروژه‌ای با نام Kroupware، اقدام به اضافه کردن پاره‌ای امکانات به KDE کرده است. سایر دول اروپایی نیز همگی مشغول بررسی KDE می‌باشند. شرکت Apple برای ساخت مرورگر جدید خود به نام Safari، از مرورگر Konqueror، استفاده کرده و پیشرفت‌های خود به این مرورگر را در اختیار پروژه KDE قرار داده است. همگی نشانه‌ها بیانگر آن است که آینده درخشانی در انتظار این پروژه می‌باشد و سناریوی "یونیکس سخت است" دیگر صادق نیست. ترکیب KDE/لینوکس آماده فتح رایانه‌های رومیزی می‌باشد.

۳-۱۴ چرا ابونتو (ubuntu)

Ubuntu واژه ای آفریقایی است که به معنای "نوع دوستی و انسانیت نسبت به دیگران" می‌باشد. اما در دنیای نرم‌افزار Ubuntu سیستم عاملی مناسب برای لپ‌تاپ‌ها، کامپیوترهای خانگی و سرورهاست.

محبوبیت (برای هر چیزی و برای مدت طولانی) چیزی اتفاقی نیست. حتی اگر از دنیای لینوکس بیرون بیایم و به ویندوز و یا مک نگاه کنیم می‌بینیم جدای از بحث تحمیل سیستم عامل به دیگران، قابلیت‌هایی بسیار ساده دارند که همه می‌توانند به راحتی از آن استفاده کنند (ابونتو!). ابونتو یکی از توزیع‌هایی است که فلسفه‌ی استفاده از گنو/لینوکس را تغییر داده است. یعنی اینکه به جای اینکه کاربران لینوکس را بخواهند، لینوکس کاربران را بخواهد (همان هدفی که سیستم‌عامل‌های تجاری به دنبال آن هستند!) به همین دلیل سعی کرده است که نیازمندی‌های کاربران عادی را به طور بسیار عالی برطرف کند. توضیح اینکه بیشتر [توزیع‌های](#) گنو/لینوکس نیازمندی‌های کاربران متخصص رایانه را به خوبی برطرف کرده‌اند. اما برای کاربران عادی و سطح پایین کار کردن با واسط خط فرمان و یا کامپایل کردن برنامه‌ها می‌تواند همانند یک کابوس وحشتناک باشد.

به همین دلیل ابونتو توسط شرکت کانونیکال برای این مقصد ساخته شد. ابونتو بیشتر سعی می‌کند از واسط گرافیکی استفاده کند تا متنی. همچنین سعی جامعه‌ی آن بر فراهم ساختن نسخه‌های باینری برای بیشتر برنامه‌های منتشر شده در جهان می‌باشد.

یکی از دلایل اصلی که کاربران از ابونتو استفاده می‌کنند، همین بزرگ بودن جامعه‌ی آن و فراهم کردن نیازمندی‌های کاربران به طور کامل است. همچنین هر نرم‌افزاری که نوشته می‌شود می‌توانید نسخه‌ی باینری برای آن را در مخازن گوناگون پیدا کنید. این مزیت بسیار خوبی است. مزیتی که هزینه کردن وقت و فضا را کاهش می‌دهد. برای یک کاربر غیربرنامه‌نویس هیچ الزامی وجود ندارد که کد منبع یک برنامه‌ی کاربردی را مطالعه کند و یا آن را کامپایل کند.

دلیل بسیار مهم دیگر استفاده از یک محیط گرافیکی در کل سیستم عامل است. برای مثال در ابونتو محیط گرافیکی پیشفرض گنوم (GNOME) ، در کوبونتو کی.دی.ای (KDE) و در Xubuntu XFCE است. این کار دو مزیت بزرگ دارد :

- صرفه جویی در فضای مورد استفاده
- هماهنگی با کلیه برنامه در محیط گرافیکی

برای کسی که دوست ندارد از GNOME استفاده کند ، نیازی نیست که گنوم روی سیستم عاملش نصب باشد و بی جهت فضای سیستم را بگیرد. همچنین بیشتر توزیع های دیگر هر دو محیط گرافیکی (و گاهی هر سه !) را به همراه دارند. این مطلب درست است که می توان آن را سفارشی کرده و یا هر کدام را که خواستیم نصب کنیم اما در این توزیع ها هماهنگی کامل ملاک قرار نگرفته است. در صورتی که شما تنها از یک محیط گرافیکی استفاده کنید ، می توانید این هماهنگی را ایجاد کنید.

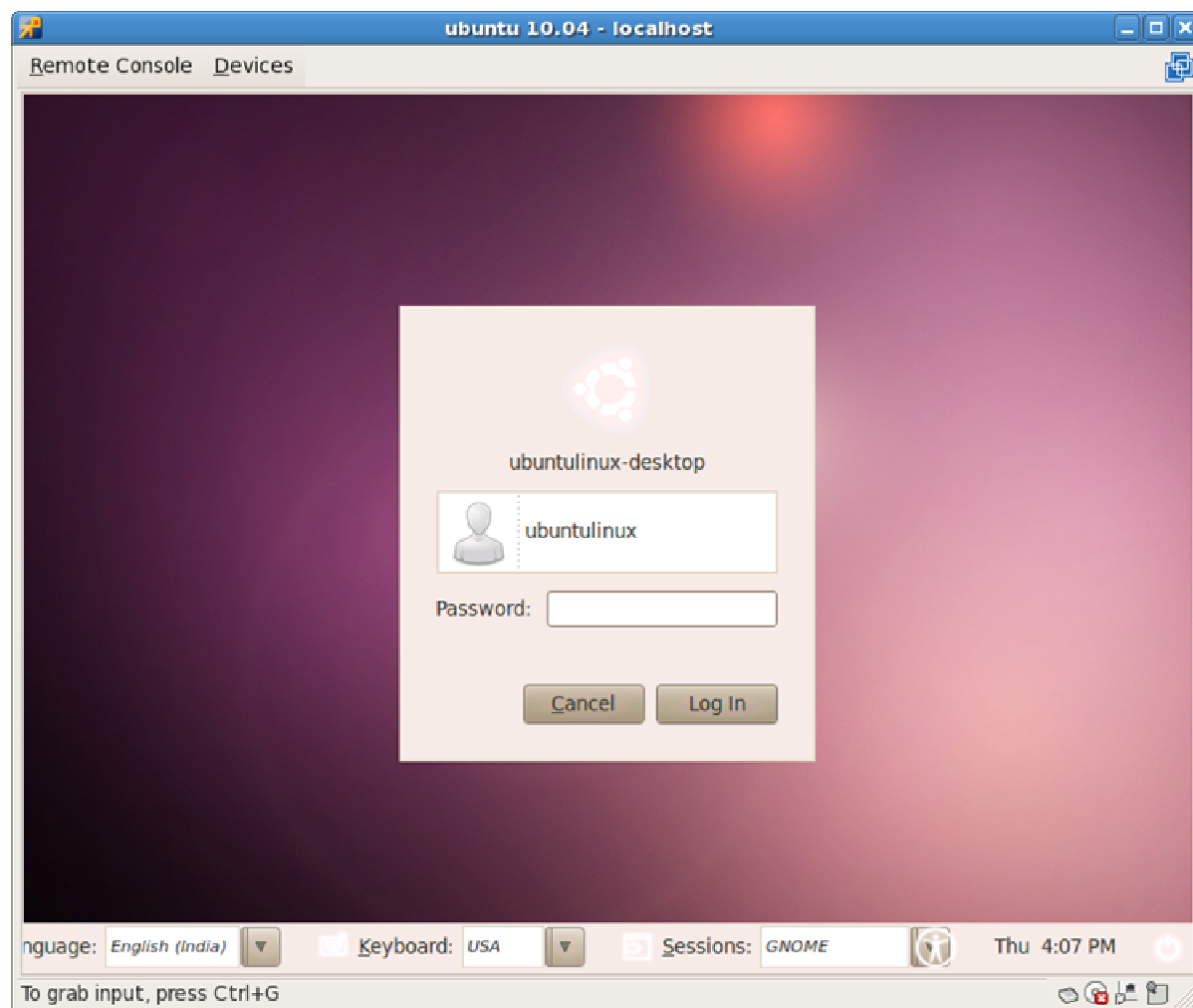
ویژگی دیگری که ابونتو را ساده کرده است چیزی است که از [Debian](#) به ارث برده است. یعنی همان راحتی نصب پکیج های deb و قدرتمندی آن ها است. به طوری که به راحتی کافی است مخزن مورد نیاز را به لیست مخازن خود اضافه کنید و هر برنامه ای را که می خواهید با دستور زیر نصب کنید :

برنامه مورد نظر `sudo apt-get install`

به هر حال در میان جامعه ی کاربران گنو/لینوکس معروف است که تنها یک دلیل خوب و معقولانه برای پیوستن به یک جریان یا توزیع کافی است !

۳-۱۵ ورود به سیستم

تصویر یک صفحه ورود به سیستم گرافیکی را نشان می دهد. در پایین صفحه چند منو مشاهده می کنید. البته هنگام ورود به سیستم در حالت عادی نیازی به کار کردن با این منوها ندارید ولی کاربرد این منوها به ترتیب زیر می باشد:



Session: به طور معمول هنگامی که از طریق صفحه ورود گرافیکی به سیستم وارد می شوید، محیط گرافیکی لینوکس بارگذاری می شود. این محیط بسته به انتخاب شما ممکن است GNOME یا KDE باشد. منوی Session برای تغییر محیط گرافیکی استفاده می شود. مثلاً هنگامی که هر دو سیستم را روی کامپیوتر خود دارید و گاهی از KDE استفاده می کنید و گاهی از GNOME. گزینه Last در این منو محیط گرافیکی استفاده شده و گزینه safe یک محیط مبتنی بر پوسته فرمان را بارگذاری می کند.

• **Language:** با این گزینه می توانید زبانی غیر از زبانی که قبلاً استفاده کرده اید را انتخاب کنید. البته این گزینه به شرطی کار خواهد کرد که شما بسته های نرم افزاری زبان های دیگر را نصب کرده باشید.

Restart, Shut down: در صورتی که بجای ورود به سیستم، مایل به خاموش کردن یا راه اندازی مجدد کامپیوتر خود هستید، یکی از این گزینه ها را انتخاب کنید.

لینوکس می تواند همزمان به تعداد زیادی کاربر سرویس دهد. وقتی مدیر سیستم یک کاربر جدید تعریف می کند یک Account اختصاصی با نام loginname یا username ایجاد می شود. هر کاربر تنها با داشتن نام و کلمه ی عبور password خود می تواند به محیط سیستم وارد شود و تنها اجازه ی دسترسی به فایل های خود را داشته باشد. در اینجا فرض بر این است که شما کامپیوتری در اختیار دارید که سیستم عامل لینوکس روی آن نصب و یک نام کاربری به شما اختصاص داده شده باشد.

ابتدا نام کاربری خود را وارد کنید. (توجه کنید که لینوکس نسبت به حروف کوچک و بزرگ حساس است یا اصطلاحاً casesensitive است). پس از وارد کردن نام (چه صحیح وارد کرده چه غلط) سیستم از شما کلمه عبور درخواست می کند. آن را وارد کنید. اگر وارد نشدید:

الف - نام کاربری را چک کنید

ب - پسورد را چک کنید

ج - به حروف کوچک و بزرگ دقت کنید

ج د- از مدیر سیستم بپرسید آیا چنین کاربری تعریف شده یا نه

پس از وارد کردن Username و Password صحیح صفحه نمایش لینوکس، (بسته به انتخاب GNOME یا KDE) را مشاهده میکنید. (GNOME محیط پیش فرض desktop است)

۳-۱۶ آشنایی با بخشهای مختلف محیط های گرافیکی

سیستم عامل لینوکس از دو محیط گرافیکی (Interface GUI (Graphical User Interface و محیط متنی CLI (Command Line Interface) تشکیل شده است

برای رفتن به محیط GUI با فشار دادن دکمه های Ctrl+Alt+F1 و یا Ctrl+Alt+F2...F6 میتوانیم به آن دستیابی پیدا کنیم و با فشار دادن Ctrl+Alt+F7 میتوانیم به محیط گرافیکی وارد شویم

۳-۱۶-۱ محیط گرافیکی GNOME

هنگامی که شما از Xwindows در GNOME استفاده می کنید، چندین آیتم می بینید از جمله :

☀️ محیط کاری (Desktop) :

فضای اصلی کار شماست که بیشترین فضای صفحه را اشغال کرده است. کلیه پنجره ها روی این محیط قرار می گیرند. مدارک، پوشه ها، میانبرها و دیگر داده ها نیز روی این محیط قرار می گیرند.

☀️ پنل (Panel) :

میله بلندی است که سر تا سر پایین محیط کاری را پر کرده است. تعدادی از میانبرها و اپلت ها (Applet) روی آن قرار دارد.

Applet از کلمه mini application به معنی درخواستهای کوچک گرفته شده است.

پنل، قابل تغییر و پیکربندی است. شما می توانید ابزارها و اپلت هایی را روی پنل بگذارید و از آنها استفاده کنید. در ضمن، می توانید با کلیک کردن روی علامتهای فلش مانند در دو طرف پنل، آنرا در سمت چپ یا راست پنهان کنید.

☀️ دکمه منوی اصلی :

معمولاً اولین دکمه از سمت چپ روی پنل است که به شکل یک جای پا است. (آرم GNOME) این منو شامل زیر منوها، اپلت ها و ابزارهای کاربردی است. بیشتر برنامه های لینوکس را نیز می توان از آنجا یافت. البته می توان چیزهایی نیز به آن افزود.

☀️ میانبرها (Launchers) :

آیکونهایی هستند که روی محیط های کاری می آیند و شما با یک کلیک روی آنها می توانید به برنامه یا محیط مربوطه دسترسی پیدا کنید. مثلاً اگر روی آیکون Netscape کلیک کنید به این مرورگر وب دسترسی پیدا خواهید کرد.

☀️ پوشه Home :

این پوشه دسترسی شما را به فهرستهای مخصوص کاربران که در /home/youruser قرار دارد، امکان پذیر می کند.

☀️ اضافه کردن اپلت ها :

روی پنل کلیک راست کنید و Applet را انتخاب کنید. در پنجره جدید، اگر روی ابزار مورد نظر خود کلیک کنید، آن ابزار به پنل اضافه خواهد شد.

☀ استفاده از راهنما (Help) :

در روی پنل دکمه ای وجود دارد که یک علامت سوال (?) روی آن دیده می شود. با کلیک روی آن می توانید به راهنمای GNOME دسترسی پیدا کنید. این راهنما یکی از مفیدترین ابزارهای است که بسیار به شما کمک می کند.

☀ محیط کاری چندگانه :

اگر صفحه کاری شما کمی شلوغ شد به طوری که شما را گیج می کرد، یعنی مثلا اگر در هنگام کار مجبور شدید چندین پنجره را با هم باز کنید، می توانید کارتان را در چندین صفحه کاری تقسیم کنید. در روی پنل در سمت راست، یک مربع وجود دارد که به چهار قسمت تبدیل شده است. با کلیک روی هر کدام از این قسمتها، به صفحه مربوطه می روید. در حقیقت، شما به طور پیش فرض، چهار صفحه کاری دارید که می توانید این تعداد را کم و زیاد کنید.

☀ سوییچ کردن بین محیط های کاری :

در لینوکس ابزاری به نام switchdesk وجود دارد که این امکان را برای شما ایجاد می کند که در هنگام کار در یکی از محیطهای گرافیکی، به یکی از محیطهای دیگر بروید. مثلا شما در GNOME کار می کنید و می خواهید به KDE بروید. باید با استفاده از این ابزار، رابط گرافیکی را به KDE تغییر داده و از سیستم خارج شوید. حال اگر دوباره وارد سیستم شوید، خود را در رابط گرافیکی KDE خواهید یافت. برای اجرای این برنامه از منوی اصلی GNOME، گزینه Programs سپس گزینه System و سپس گزینه desktop switching tool را انتخاب کنید.

☀ پایان کار در GNOME :

در منوی اصلی GNOME دکمه ای به نام Log Out وجود دارد که شما با استفاده از این دکمه می توانید عمل shutdown یا reboot کردن را انجام دهید. همچنین می توانید از محیط گرافیکی که در آن هستید خارج شوید.

🍌🍌 ۳-۱۶-۲ محیط گرافیکی KDE:

شکل ظاهری آن تقریبا شبیه GNOME است. هنگام ورود به این محیط ، desktop, panel و... را می بینید. مفاهیم desktop, panel, محیط کاری چندگانه ، پوشه ها و... در قسمت قبل توضیح داده شد. این مفاهیم، دقیقا در مورد KDE هم صادق است.

پس در اینجا چند ویژگی متفاوت KDE را می گوئیم :

☀ Floppy , Trash , CD-Rom :

آیکونهایی هستند که بر روی صفحه کاری دیده می شوند. آیکون CD-Rom و Floppy دستیابی شما را به فلاپی درایو و سی دی آسان می سازند. یعنی با یک کلیک روی آنها می توانید محتویات CD یا Floppy را مشاهده کنید. آیکون Trash هم همان سطل آشغال KDE است که آن را در ویندوز با نام Recycle Bin می شناسید. برای حذف یک پوشه یا فایل، می توانید آنرا با ماوس گرفته و به درون سطل آشغال بیندازید.

☀ آیکون منوی اصلی K :

آیکونی روی پنل شما وجود دارد که یک حرف بزرگ K روی آن دیده می شود. در این منو می توانید انواع ابزارها و برنامه ها و زیرمنوهای موجود در KDE را بیابید. این برنامه ها و زیرمنوها قابل تغییرند یعنی قابل حذف و اضافه هستند.

☀ راهنمای KDE

همانند GNOME ، KDE هم یک راهنمای مفید و پرکاربرد دارد که می تواند کمک خوبی برای شما باشد. این راهنما شامل راهنمای اقسام مختلف KDE ، راهنمای پویش سیستم، سفارشی کردن محیط کاری و دیگر کارهای سیستم شماست.

اضافه کردن میانبرها، اپلت ها و برنامه هایی به پنل : برای اضافه کردن چیزی به پنل هم می توانید مانند GNOME ، از روی پنل عمل کنید و هم می توانید از منوی اصلی K گزینه Panel سپس Add Application سپس برنامه مورد نظر را انتخاب کنید. با این انتخاب، این برنامه به پنل اضافه می شود.

دو آیکون مهم منوی اصلی K :

• Center KDE Control :

مرکز کنترل KDE است که با استفاده از آن می توانید در محیط کاری تغییراتی بوجود آورید. در ضمن، بسیاری از اطلاعات مهم و اصلی مربوط به سیستم، تنظیمات سخت افزار و برنامه ها و... را در این مکان خواهید یافت.

• Logout , Lock Screen :

با استفاده از Lock Screen می توانید صفحه کامپیوتر را قفل کنید و تا زمانی که رمز را وارد نکنید، صفحه به همان حالت باقی می ماند. Logout هم برای Shut Down و یا Reboot کردن کامپیوتر و یا خارج شدن از محیط KDE استفاده می شود.

۳-۱۶-۳ چند نکته :

(۱) استفاده از ماوس در KDE و GNOME شبیه کار آن در ویندوز است مثلا کلیک راست روی یک آیکون، امکان هرگونه تغییرات روی آن و نمایش اطلاعات مربوط به آن را می دهد. یا می توانید جای آیکونهای روی دسکتاپ را با کشیدن و رها کردن آنها با ماوس تغییر دهید.

(۲) یکی از تفاوتهای KDE با GNOME این است که برای اجرای یک برنامه که آیکون آن روی صفحه دسکتاپ بود در GNOME می بایست روی آن آیکون دوبار کلیک می کردید. اما در KDE ، تنها با یک بار کلیک روی هر آیکون، آن برنامه اجرا می شود.

(۳) همانطور که گفتیم KDE هم مانند GNOME دارای صفحه کاری چندگانه می باشد. برای تعویض سریع این صفحه ها می توانید از ترکیب کلیدهای Ctrl+Fx استفاده کنید. (منظور از Fx کلیدهای F۱, F۲, F۳ و... می باشد که باید به جای X ، شماره صفحه کاری را بزنید.) مثلا با استفاده از Ctrl+F۲ می توانید به صفحه کاری ۲ بروید و به همین ترتیب برای رفتن به صفحات دیگر عمل کنید.

(۴) اگر آیکون Help روی پنل نبود، میتوانید از منوی اصلی K و با انتخاب گزینه KDE Help آنرا اجرا کنید.

۳-۱۷ استفاده از دستورات در خط فرمان

شما می توانید هم در محیط گرافیک لینوکس (x-windows) و هم در محیط متنی کار کنید. سیستم شما در محیط گرافیکی فعال است لیکن اغلب دستورات را در محیط متنی به نام پنجره ی ترمینال (terminal windows) تایپ و اجرا می کنیم.

باز کردن یک پنجره ترمینال (ورود به محیط Console):

از مسیر Application → Accessories → Terminal یک ترمینال باز کنید. در این مسیر می توانید با کلیک راست روی آیکن ترمینال یک ایکن روی نوار وظیفه و یا روی دسک تاپ خود اضافه کرده تا به سرعت در دسترس باشد.

۳-۱۷-۱ دو نوع کاربر (دو سطح دسترسی)

- مجری سیستم (root) که وظایف نا محدود دارد
- کاربران عادی

در ویندوز administrator توانایی انجام هر گونه تنظیمی مثل اضافه کردن، حذف کاربر، دادن permission به یک فلدر و یا فایل خاص و ... دارد در سیستم لینوکس administartor را root می نامیم. بدین مفهوم که root قادر به انجام اختیارات گفته شده می باشد.

سطر فرمان برای این کاربران با شکل متفاوت ظاهر می شود :

[~root@hostname]

برای مجری سیستم به صورت مقابل

\$ [~username@hostname]

برای کاربران عادی به شکل

root نشاندهنده اسم user می باشد

~ دایرکتوری که در آن هستیم می باشد

* hostname نام کامپیوتری که میزبان به آن متصل شده.

۳-۱۷-۲ بررسی نشست ورود (به) و خروج (از) سیستم

هنگامی که وارد سیستم لینوکس میشوید، برای سیستم دارای یک هویت خاص هستید. این هویت شامل نام کاربری شما، نام گروه شما، شماره کاربری شما و شماره گروه شماست. همچنین لینوکس اطلاعات زمان ورود به سیستم، مدت حضور، مدت بیکاری و محل ورود شما به سیستم را نگهداری میکند. (حواستان را جمع کنید!) برای بدست آوردن اطلاعات در مورد هویت کاربری خودتان در جلوی اعلان فرمان دستور زیر را تایپ کنید. خروجی آن در زیر آن نشان داده شده است:

id

uid=500(Alan) gid=500(Alan) groups=500(Alan)

خروجی فرمان نشان میدهد که نام کاربر Alan بوده که عضو گروه Alan است و شماره های کاربری و گروه آن ۵۰۰ میباشد.

از فرمان زیر برای نمایش معماری سیستم استفاده میشود

arch

از فرمان زیر برای نمایش اسم **login** استفاده میشود

logname

از دستورات زیر برای خروج از حساب کاربری استفاده میشود

logout

exit

reboot

از دستورات زیر برای راه اندازی مجدد سیستم استفاده میشود

init 6

از فرمان زیر برای خاموش کردن سیستم استفاده میشود

shutdown

init 0

۳-۱۷-۳ **CommandEditingTricks** ویرایش دستورات خط فرمان

بوسیله فشار دادن **ctrl+a** در خط فرمان مکان نما به ابتدای خط انتقال پیدا می کند

بوسیله فشار دادن **ctrl+e** در خط فرمان مکان نما به انتهای خط انتقال پیدا می کند

بوسیله فشار دادن **ctrl+u** از ابتدای خط فرمان تا آن مکان را پاک می کنیم

با فشار دادن **ctrl+k** از مکان نما تا انتهای خط فرمان را پاک می کنیم

با فشار **ctrl+arrow** از چپ به راست یک حرف انتقال می یابد

۴-۱۷-۳ **uname,hostname** دستور

با استفاده از برخی دستورات مانند **uname** می توانید به اطلاعات اولیه در مورد سیستم لینوکس خود دست پیدا کنید. این اطلاعات می تواند شامل موارد زیر باشد:

- سیستم عامل (لینوکس)

- نام کامپیوتر

- شماره نسخه لینوکس

- معماری پردازنده

در مثال زیر، با استفاده از دستور **uname -a** این اطلاعات استخراج شده است :

uname -a

GNU/Linux Sun Aug 3 20:15:59 PDT 2003 i686 ۱# ۲,۴,۲۱ Linux Roohollah

نام کامپیوتر در مثال بالا **memphis** است. از این نام برای شناسایی کامپیوتر در محیطهای اشتراکی مانند شبکه استفاده می شود. شماره ۲,۴,۲۱ نشاندهنده شماره نسخه هسته لینوکس است.

با استفاده از دستور **hostname** می توانید نام کامپیوتر خود را تغییر دهید. به مثال زیر توجه کنید:

hostname نام جدید

hostname Pishbahar

این دستور نام کامپیوتر را از **Roohollah** به **Pishbahar** تغییر می دهد. در صورتی که این دستور را به تنهایی تایپ نمایید:

hostname

نام فعلی کامپیوترتان را به شما نشان خواهد داد:

Memphis

۳-۱۷-۵ دستور تاریخ

date

نکته: در کوچک و بزرگ بودن حروف دقت شود .

date -u

دستور مقابل زمان را به وقت گرینویچ نشان می دهد .

۳-۱۷-۶ دستور cal

از این فرمان برای نمایش تقویم استفاده می شود .

-بدون وارد کردن گزینه خاص تقویم جاری را نشان می دهد .

cal 7 1982

دستور مقابل تقویم ماه و سال داده شده را می دهد .

نکته : در این دستور برای نشان دادن ماه ها باید از اعداد استفاده کنید .

تمرین: نتیجه دستورات زیر را بیان کنید

cal -y

cal -3

cal 2005

۳-۱۷-۷ پوسته چیست؟

یک برنامه یونیکس است که دستورات وارد شده از طریق صفحه کلید را اجرا می کند.

یونیکس پوسته های متعددی دارد : **c** ، **korn** ، **bourne** و **c** .

لینوکس از پوسته مجانی دیگری به نام **bash** استفاده می کند که بهترین خصوصیات در پوسته های **korn** و **bourne** را دارا می باشد .

بدون توجه به نوع پوسته ارتباط شما با یونیکس از طریق پوسته انجام می شود .

انواع پوسته های موجود در زیر آمده است :

– بون – کورن – cshell – bash – tcsh – zsh

با وارد کردن نام یک پوسته می توانیم از یک پوسته به پوسته دیگر وارد شویم.

برای دانستن اینکه در کدام پوسته کار می کنید فرمان روبرو را وارد کنید

اگر پوسته **bash** باشد فرمان زیر ظاهر می شود:

```
echo $SHELL
```

```
/bin/bash
```

۳-۱۷-۸ استفاده از چند دستور هم زمان

برای جدا کردن چند دستور که می خواهیم در یک خط بنویسیم از جدا کننده ";" استفاده می کنیم:

```
Cal ; date
```

۳-۱۷-۹ دستور clear و فراخوانی مجدد فرمان (history)

اگر به جواب فرامین قبلی نیاز ندارید از این فرمان استفاده کنید و صفحه را پاک کنید .

پس از اینکه یک دستور را تایپ کردید، این دستور بطور کامل در تاریخچه پوسته فرمان ذخیره میشود . برای نمایش محتویات تاریخچه پوسته استفاده کنید . در صورتی که پس از آن یک عدد **history** فرمان میتوانید از دستور اضافه کنید، به تعداد آن عدد دستورات تایپ شده را نشان خواهد داد:

```
history 5
```

سابقه سطر فرمان :

اغلب فرمانی واحد را در زمانی کوتاه بار ها وارد می کنید . پوسته ها آخرین فرامینی که استفاده کرده اید را در خود نگه میدارند و شما بدون تایپ مجدد می توانید آنها را فراخوانی کنید . از کلید های **scroll up** و **scroll down** برای آوردن دستورات قبلی و بعدی استفاده می شود .

```
history -c
```

از این فرمان به منظور پاک کردن فرمانهای اجرا شده قبل استفاده میشود

۳-۱۷-۱۰ دستورات کمک (man و Info, Help, what is)

--help نام دستور

Info دستور

man دستور

توضیحاتی در مورد دستور ها به کاربر ارائه می دهد .

– اسم دستور و شرح مختصری از آن

– نحوه به کار بستن دستور

– توضیح

– گزینه ها

– عیوب و نواقص شناخته شده در دستور

Man مخفف Manual و Info مخفف Information می باشد

نام دستور Whatis

اطلاعات مختصری در مورد دستور می دهد

۳-۱۷-۱۱ خروج از یونیکس (UNIX) و لینوکس (Linux)

برای پوسته های مختلف دستورات مختلفی وجود دارد :

پوسته های **c** با تایپ **logout** و پوسته های **bash**، **korn**، **bourne** با تایپ **ctrl+d** و **exit** خازج می شود .

۳-۱۷-۱۲ تغییر رمز عبور

دستور `passwd` را تایپ می کنید. پس از زدن `enter` از شما پسورد قبلی (`current password`) و پسورد جدید را می خواهد.

`passwd`

۳-۱۷-۱۳ هسته لینوکس و دستور `cat`

قلب سیستم عامل لینوکس هسته یا `kernel` نام دارد. هسته لایه بین شما و برنامه‌های کاربردی سیستم را با سخت افزار کامپیوتر تشکیل می‌دهد. با استفاده از فایل‌های موجود در مسیر `proc` می‌توانید اطلاعات فراوانی را از هسته سیستم خود بدست آورید.

برای هر پروسه‌ای که اکنون در کامپیوتر شما در حال اجراست، یک دایرکتوری در مسیر `proc` ایجاد می‌شود که نام آن شماره پروسه مربوطه است.

برای نمایش محتویات فایل‌های موجود در دایرکتوری `proc`، می‌توانید از دستور `cat` استفاده نمایید. برای مثال دستور:

`cat /proc/version`

* در جلوی فرمان `cat` می‌توانید نام چند فایل را بنویسید تا محتویات همه را به ترتیب نشان دهد.

`cat /etc/shells /proc/version`

شماره نسخه هسته را به شما نشان خواهد داد. بجز فایل `version`، فایل‌های دیگری نیز وجود دارند که می‌توانید اطلاعات جالبی از آنها بدست آورید:

- `cpuinfo`: نوع پردازنده نصب شده در کامپیوترتان بعلاوه سرعت، خانواده و اطلاعات دیگر مربوط به آن را نشان می‌دهد.
- `devices`: ابزارهای بلاکی و کاراکتری موجود در کامپیوترتان را بعلاوه شماره های آنها نمایش می‌دهد.
- `ioports`: آدرس‌های درگاه‌های `I/O` روی کامپیوترتان را نمایش می‌دهد.
- `meminfo`: حاوی اطلاعاتی در مورد مصرف حافظه و فضای `swap` است. می‌توانید ببینید که کلاً چه میزان حافظه وجود دارد و چه میزان از آن مصرف شده است.
- `modules`: لیست ماژول‌هایی که در حال حاضر در هسته نصب شده‌اند را نشان می‌دهد.
- `mounts`: لیست فایل سیستم‌هایی که در حال حاضر متصل هستند را نشان می‌دهد.
- `partitions`: حاوی نام پارتیشن‌های دیسک سخت شما و تعداد بلاک‌های موجود روی هر پارتیشن است. همچنین شماره ابزار `major` و `minor` آن نیز نمایش داده می‌شود.
- `pci`: لیست ابزارهای `pci` موجود در کامپیوترتان را نشان می‌دهد. همچنین می‌توانید اطلاعاتی مانند شماره `bus`، نام ابرارها، شماره‌های وقفه و ... را نیز بدست آورید.
- `swaps`: اطلاعات پارتیشن‌های `swap` متصل شده به سیستم فایل را نمایش می‌دهد. این اطلاعات شامل اندازه و مقدار فضای مصرف شده آنها نیز می‌باشد.
- `net/dev`: اطلاعات مربوط به رابط‌های شبکه (مانند کارتهای شبکه) فعال روی کامپیوتر را نشان می‌دهد.

`cat > file name`

کاربرد این دستور را با مثال بررسی می‌کنیم:

-۱-

`cat filename1 filename2 filename3`

در این حالت دستور `cat` محتوای فایل‌های نوشته شده را به ترتیب در خروجی استاندارد چاپ می‌کند.

-۲

```
cat filename→
.....
.....
```

پایان ورود اطلاعات به فایل (Ctrl+Z)

در این حالت فایلی با نام filename تولید می شود.

-۳

```
cat filename1 filename2 filename3 >allfile
```

با این دستور محتوای فایل های ذکر شده به فایلی با نام allfile اضافه می شود.

۳-۱۷-۱۴ دستور df, du و فرمت کردن فلاپی

دستور df میزان فضای استفاده شده ,

df

پارامتر **-h** در دستور df ، میزان فضای استفاده نشده پارتیشن را بر حسب MG و یا نشان می دهد

du که به معنی data universal است برای دیدن مقدار فضای هر دایرکتوری بصورت کیلوبایت می باشد

```
du /opt
```

همچنین با استفاده از **-s** میتوانیم مجموع فضای اشغال شده یک دایرکتوری را مشاهده کنیم

```
du -s /etc
```

از فرمان زیر به منظور فرمت کردن floppy disk استفاده میکنیم

```
mkfs -t ext2 /dev/fd0
```

۳-۱۷-۱۵ دستورات واریسی فایل :

برای مشاهده کل فایل از دستور cat استفاده می شود.

برای مشاهده فایل به صورت صفحه به صفحه از دستور **more** و **less** استفاده می کنیم.

دستور **head** اولین خط یک فایل را نشان می دهد و دستور **tail** آخرین خط فایل را نشان می دهد .

```
head -n
```

فایل مورد نظر

n تعداد خطوطی که این دستور نمایش می دهد را تعیین می کند .

مثال:

```
tail -5 یا head -5
```

```
head -20 /etc/passwd > mypasswd
```

با این دستور فایلی شامل بیست خط اول فایل پسورد بسازید. سپس با دستور زیر محتویات آن را ببینید.

این دستور فهرستی از پوسته های قابل دسترسی را نشان می دهد .

```
more /etc/termcap
```

- کلید space برای عوض کردن صفحه

- Q برای خارج شدن

```
less /etc/termcap
```

- کلید حرکت رو به پایین خط به خط فایل را رو به جلو نمایش می دهد

- کلید حرکت رو به بالا خط به خط فایل را به عقب می برد
- کلید page down یا space برای رفتن به صفحه بعدی
- کلید page up یا back space برای رفتن به صفحه قبل
- وقتی علامت ' : ' پایین صفحه بود با q بسته می شود .

با استفاده از دستور **sort** می توانیم فایل را به صورت مرتب شده مشاهده کنیم

👉👉 -n مرتب کردن بر اساس شماره (مخفف numerical).

👉👉 -f مرتب کردن بر اساس حروف الفبا

👉👉 -r مرتب کردن به ترتیب نزولی

```
sort -n subnet
```

```
sort -f subnet
```

```
sort -r subnet
```

```
cat mypasswd
```

و مرتب کنید:

```
sort mypasswd
```

دوباره ببینید:

```
cat mypasswd
```

توجه کنید که فایل تغییری نمی کند .اطلاعات پس از خوانده شدن در حافظه اصلی (RAM) مرتب شده و نشان داده می شود.
تمرین: نتیجه دستورات زیر را بیان کنید

```
history |more
```

```
history |less
```


۳-۱۷-۱۶ تغییر مسیر :

دستورات از طریق صفحه کلید به سیستم داده شده و خروجی آنها روی صفحه مانیتور ظاهر می شود. بنابراین سیستم بصورت پیش فرض صفحه کلید را بعنوان ورودی استاندارد و صفحه نمایش را بعنوان خروجی استاندارد در نظر می گیرد. اما می تواند با استفاده از کاراکترهای < , > , < > و < > ورودی و خروجی استاندارد را تغییر داد. یکی از کاربردهای این کاراکترها در چنین حالتی، در فرمان cat می باشد. بکارگیری فرمان cat در این حالت برای تغییر در محتویات فایلها و کپی محتوا یا جایگزین کردن محتوای یک فایل با فایل دیگر می باشد.

مثال :

```
cat > filename
cat >> filename
```

تمرین : فرق دو دستور بالا را بررسی کنید .

۳-۱۷-۱۷ به کار بستن پیکر بندی های unix :

ملحقات کامپیوتر از طریق واسطهای الکترونیکی به کامپیوتر متصل می شوند. ۲ نوع معروف این واسط ها عبارتند از IDE^۱ و SCSI^۲. Unix برای نام گذاری ابزار هایی که از طریق این دو واسط به کامپیوتر وصل می شوند از دو روش نام گذاری مجزا استفاده می کند. ابزار هایی مانند هارد دیسک ، CD-ROM ، از واسط های IDE استفاده می کنند ، unix ، ۲ حرف اول نام آنها را با hd نشان می دهد و برای وسایل جانبی که با واسط های SCSI به کامپیوتر وصل می شوند ، ۲ حرف اول نام آنها sd می باشد. یک حرف "a" یا "b" بعد از "hd" یا "sa" قرار می گیرد که نشانه primary یا secondary بودن وسیله جانبی است .

/dev/hda1	/dev/sda1
/dev/hda2	/dev/sda2
/dev/hda3	/dev/sda3

۳-۱۷-۱۸ دستور pwd :

این فرمان بمنظور نمایش مسیر جاری سیستم بکار می رود. کافی است در خط فرمان، آنرا تایپ کرده و Enter بزنید.

۳-۱۷-۱۹ دستور cd :

cd directory name

برای تغییر مسیر از این دستور استفاده می شود.

cd\ : انتقال به دایرکتوری ریشه

cd.. : انتقال به دایرکتوری بالا

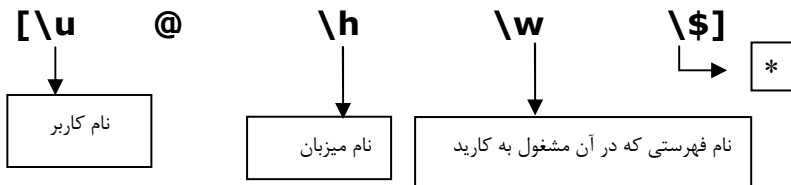
cd : انتقال به home directory برای هر کاربر

¹ - Integrated Drive Electronics
² - Small Computer System Interface

۳-۱۷-۲۰ به فرم دلخواه در آوردن خط فرمان :

PS1 متغیری است در محیط unix که اطلاعات مربوط به چگونگی نمایش خط فرمان را نگه می دارد. با تغییر دادن این متغیر می توانید سطر فرمان را به فرمی که میخواهید در آورید.

echo \$PS1 ↴



*بسته به اینکه کاربر ریشه است یا کاربر عادی شکل این علامت # یا \$ می باشد.

\!	شماره فعلی تاریخچه فرمان را نشان میدهد.
\#	شماره دستور آخرین دستور را نشان میدهد.
\t	نمایش زمان
\h	نام میزبان
\d	نمایش تاریخ
\u	نمایش نام کاربر
\W	نمایش فقط نام فهرست جاری
\n	نمایش new line
\w	نمایش فهرست جاری
\\	نمایش /
\\$	اگر کاربر ریشه بود # در غیر اینصورت \$ را نشان می دهد

مثال :

```
export PS1="[ \t \w] \$]"
ps1="\d \t >"
ps1="\w >"
ps1="$PWD"
```

۳-۱۷-۲۱ دستور ls:

این فرمان جهت لیست گیری و مشاهده مشخصات فایل ها و پوشه های درون یک مسیر بکار می رود. کاربرد option های این فرمان عبارتند از:

👉👉 -f باعث می شود فایلها و پوشه های مخفی هم در نتیجه لیست گیری نمایش یابند.

👉👉 -a مانند سوئیچ f فایلها و پوشه های مخفی را هم نمایش می دهد با این تفاوت که نتیجه بر حسب حروف الفبا بصورت -f باعث می شود فایلها و پوشه های مخفی هم در نتیجه لیست گیری نمایش یابند.

👉👉 -i برای هر فایل یا پوشه شماره inode آن نیز در ستون اول سمت چپ ظاهر خواهد شد.

👉👉 -l لیست کامل تمام مشخصات فایل یا پوشه را بصورت زیر در قالب ده فیلد نمایش می دهد.

Type	rwX	rwX	rwX	link-number	user	group	size	time-date	name
1	2	3	4	5	6	7	8	9	۱۰

- 1 : فیلد اول نوع فایل یا پوشه را تعیین می کند. این فیلد می تواند یکی از حروف زیر باشد:

- ✓ - فایل معمولی
- ✓ b فایل ویژه بلاکی
- ✓ C فایل ویژه کاراکتری
- ✓ p فایل pipe
- ✓ l فایل لینک
- ✓ d پوشه

- 2 : مجوزهای دسترسی به فایل یا پوشه برای کاربر مالک
- 3 : مجوزهای دسترسی به فایل یا پوشه برای کاربر گروه
- 4 : مجوزهای دسترسی به فایل یا پوشه برای کاربر دیگران
- 5 : این عدد بیانگر شماره لینک فایل می باشد
- 6 : نام کاربر مالک
- 7 : نام کاربر گروه
- 8 : اندازه فایل بر حسب بایت
- 9 : تاریخ و ساعت ایجاد فایل یا پوشه
- 10: نام فایل یا پوشه

👉👉 -m نام فایلها و پوشه ها در کل صفحه بصورت ستونهای کنار هم، نمایش می یابد و هر نام از نام بعدی با یک کاراکتر ویرگول (,) جدا شده است.

👉👉 -n مانند l فقط بجای نام مالک و گروه، شماره آنها را نمایش می دهد.

👉👉 -r باعث مرتب نشان داده شدن نتیجه لیست گیری بر حسب حروف الفبا بصورت نزولی می شود.

👉👉 -s سایز هر فایل یا پوشه را بر حسب بلاکهای ۵۱۲ بایتی در ستون اول نمایش می دهد.

👉👉 -t لیست اسامی فایل ها یا پوشه های موردنظر را بصورت مرتب شده بر حسب زمان نمایش می دهد.

نکته: در صورتیکه در فرمان ls مسیر لیست گیری نوشته نشود، فرمان روی مسیر جاری عمل خواهد کرد.

نکته: در UNIX تعدادی کاراکتر عمومی وجود دارد که می توان از آنها بعنوان الگوی جستجو در برخی از فرامین استفاده کرد.

این کاراکترها عبارتند از:

👉👉 * هر کاراکتر به هر تعداد

👉👉 ? هر کاراکتر به تعداد حداکثر یکی

👉👉 [] هر مجموعه کاراکتری که داخل کروشه ذکر شده است.

II که به معنی long list میباشد همان کار فرمان ls را انجام میدهد

II

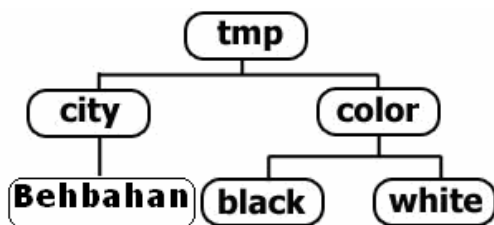
مثال :

```
ls f*
ls [aAtT]*
ls [a-f]?
```

۳-۱۸ کار کردن با فایل ها و فهرست ها و سطوح دسترسی:

اساسی ترین واحد فایل سیستم فایل نام دارد. در unix حتی با درایو ها، floppy drive و cd-rom نیز مانند فایل رفتار می شود.

Mkdir این فرمان (making directory) برای ایجاد پوشه بکار می رود. دارای دو سوئیچ p و m می باشد.



ساختار درختی روبرو را در نظر بگیرید:

[tmp] mkdir

مثال:

city

فرمان فوق سبب ایجاد پوشه city داخل tmp خواهد شد.

[tmp]mkdir -p city/Behbahan

مثال:

فرمان فوق کاربرد سوئیچ p از فرمان mkdir می باشد، در صورتیکه بخواهیم پوشه ها را بصورت متداخل ایجاد نمائیم. یعنی این سوئیچ به کاربر امکان ایجاد دایرکتوری را می دهد حتی اگر هنگام ایجاد یک زیر فهرست، فهرست والد ایجاد نشده باشد هم آنرا ایجاد خواهد کرد. در این مثال پوشه City داخل tmp ایجاد شده و همزمان پوشه Behbahan هم داخل city ایجاد خواهد شد.

[tmp] cd color

مثال:

[tmp][color] mkdir black white

در مثال فوق برای ایجاد پوشه های black و white داخل پوشه color که قبلاً آنرا ایجاد کرده ایم، ابتدا با استفاده از فرمان cd وارد پوشه color می شویم تا بتوانیم پوشه های زیرمجموعه آن را بسازیم. همانگونه که مشاهده می کنیم می توان چند پوشه همزاد را بصورت همزمان ایجاد کرد، کافیه نام آنها را با فاصله جلوی فرمان mkdir بنویسیم.

نکته: در UNIX برای فایل ها و پوشه ها سه نوع مجوز وجود دارد که عبارتند از R برای خواندن، W برای نوشتن و X برای اجرا. به هر یک از این مجوزها یک عدد اختصاص می یابد که در برخی فرامین از این اعداد استفاده خواهد شد.

اجرا X	نوشتن و تغییر W	خواندن R
1	2	4

کاربرد سوئیچ m فرمان mkdir جهت ایجاد پوشه ای با مجوزهای دلخواه بکار می رود. طریقه بکارگیری آن بدین صورت است: mxxx که هر یک از X ها یک عدد است که از حاصلجمع اعداد مجوزها برای هر کاربر سیستم (بترتیب اولویت ابتدا owner سپس group و بعد others) بدست می آید.

[tmp] mkdir -m754 fruits

مثال:

7 ← برای کاربر مالک: 7=4+2+1 یعنی کاربر مالک برای پوشه fruits تمام مجوزها اعم از خواندن، نوشتن و اجرا را داراست.

5 ← برای کاربر گروه: $5=4+1$ یعنی کاربر گروه برای پوشه fruits مجوز خواندن و اجرا را دارد.

4 ← برای کاربر دیگران: کاربر دیگران فقط مجوز خواندن را دارد.

نکته: در صورت عدم بکارگیری سوئیچ m، پوشه با مجوزهای پیش فرض ایجاد خواهد شد. مجوزهای پیش فرض برای پوشه عبارتند از:

$\begin{array}{c} rwx \\ \underbrace{\hspace{1cm}} \end{array}$ مالک (خواندن، نوشتن و اجرا)	$\begin{array}{c} r-x \\ \underbrace{\hspace{1cm}} \end{array}$ گروه (خواندن و اجرا)	$\begin{array}{c} r-x \\ \underbrace{\hspace{1cm}} \end{array}$ دیگران (خواندن و اجرا)
---	--	--

۳-۱۷ تغییر دادن حالت دسترسی به یک فایل :

از فرمان **chmod** برای تغییر دادن حالت دسترسی به فایل یا پوشه استفاده می کنیم. این روشی ساده و مؤثر برای کنترل دسترسی به فایلها و پوشه ها می باشد. جهت استفاده از این فرمان باید مالکیت فایل به شما تعلق داشته باشد. این فرمان را می توان به دو روش نمادین یا مطلق بکار برد.

روش نمادین:

سطر فرمان در حالت نمادین بشکل زیر است:

chmod [who][+/-/=[permission] names

پارامترهای حالت نمادین بشرح زیر هستند:

👉👉 **who** : تعیین کننده کاربری است که می خواهیم مجوز خاصی را از او گرفته یا به او بدهیم. بجای **who** باید از یکی از

حروف **u** برای کاربر مالک، **g** برای کاربر گروه و **o** برای کاربر دیگران استفاده کنیم. در صورتیکه بخواهیم برای تمام

کاربران بصورت یکجا مجوزها را تغییر دهیم بجای **who** از حرف **a** استفاده کنیم. همچنین می توان حروف را بصورت توأم بکار

برد.

👉👉 **+/-/=** : از این کاراکترها برای گرفتن و دادن مجوزها استفاده می کنیم. بدین صورت که کاراکتر **+** برای افزودن مجوز،

کاراکتر **-** برای گرفتن مجوز و کاراکتر **=** عملیات دادن و گرفتن مجوز را بصورت توأم انجام می دهد.

👉👉 مجوزها (permissions) : از حروف **r** برای مجوز خواندن، **w** برای مجوز نوشتن و **x** برای مجوز اجرا استفاده می

کنیم.

روش مطلق:

در این روش برای تغییر دادن اجازه های دسترسی به یک فایل یا پوشه بصورت زیر عمل می کنیم:

chmod mode names

در این روش بجای عبارت **mode** باید یک عدد سه رقمی قرار دهیم که هر یک از ارقام آن از حاصلجمع عدد متناظر با مجوزها برای هر

یک از سه کاربر سیستم بدست می آید. بدست آوردن این عدد مشابه روش سوئیچ **m** از فرمان **mkdir** می باشد.

مثال :

```
chmod u+x file1
chmod ug=r file1
chmod a=rx file1
chmod 440
chmod 347
```

۳-۱۸ تنظیم اسامی مستعار

در لینوکس این امکان وجود دارد تا برای آسانی بیشتر، اسامی مستعاری را بجای **alias** فرمان اصلی تعیین کنید. برای اضافه کردن اسامی مستعار باید از دستور استفاده کنید. به مثالهای زیر توجه کنید:

```
alias p="pwd; ls -CF"
alias rm="rm -I"
```

در مثال نخست حرف **p** دستور **pwd** را اجرا کرده و پس از آن دستور **ls -CF** اجرا خواهد شد که محتویات دایرکتوری جاری را چاپ خواهد کرد. در مثال دوم، دستور **rm** طوری تنظیم شده است تا فقط با گزینه **I** اجرا شود. در صورتی که دستور **alias** را به تنهایی تایپ کنید، لیستی از اسامی مستعاری که تنظیم کرده اید نمایش داده میشود. توجه داشته باشید که اسامی مستعار در یک فایل پیکربندی ذخیره شده و با بستن پوسته فرمان از بین نمی روند.

۳-۱۹ دستورات پردازش فایل

تغییر مسیر پیغام های خطا:

سیستم عامل **unix** خطاهای احتمالی که در اثر اجرای یک فرمان بروز می کند را روی کانال استاندارد خطا یعنی **standard error** می نویسد. اگر بخواهیم پیغام های خطای حاصل از اجرای فرامین بجای نمایش روی خروجی، در یک فایل موردنظر ثبت و ذخیره شود باید از فرمان تغییر کانال خطای استاندارد استفاده کنیم. سیستم عامل کانالی با عدد **2** را برای خطا استفاده می کند. کانال صفر به ورودی استاندارد، کانال یک به خروجی استاندارد و کانال دو به خطا تخصیص یافته اند.

مثال:

```
ls abcds 2 > err
```

نکته: برای دیدن محتوای این فایل از دستور **cat** استفاده می کنیم.

حذف فایل:

```
rm -[fri] filename(s)
```

کاربرد دستور فوق برای حذف کردن فایلها می باشد. پس از اجرای فرمان پیغامی جهت تأیید عمل حذف پرسیده می شود و در صورت تأیید کاربر فایل حذف خواهد شد.

👉 **i** - برای پاک کردن فایل ها از کاربر تایید گرفته می شود

👉 **f** - فایل هایی که برای آنها اجازه **w** ندارد را بدون گرفتن تایید (**force**) پاک می کند.

👉 **r** - تمام فایل و زیر فهرست های یک دایرکتوری را به صورت بازگشتی (**recursive**) پاک می کند.

تمرین:

```
rm f*
rm [aAtT]*
rm [a-f]?
```

دستور کپی **cp**:

این فرمان فایل یا پوشه مبدأ را در مسیر مقصد کپی می کند.

```
cp -[fri] source destination
```

مثال:

```
cp test1 apple/test2
cp /dev/t* /test
```

کاربرد سوئیچ های این فرمان بترتیب زیر می باشد:

👉 **i** - برای جایگزین کردن فایل ها از کاربر تایید گرفته می شود

🍌🍌 -f فایل هایی که برای آنها اجازه W ندارد را بدون گرفتن تایید جایگزین می کند.
 🍌🍌 -r تمام فایل و زیر فهرست های یک دایرکتوری را به صورت بازگشتی (recursive) جایگزین می کند.

حذف فهرست ها :

برای حذف باید فهرست از قبل تهی باشد (removing directory).
 نام فهرست rmdir

اگر از سوئیچ p استفاده شود در صورتیکه دایرکتوری بالا هم خالی نباشد آن را نیز پاک می کند.

مثال :

```
rmdir d2/d3 d1
rmdir -p d2/d3
```

دستور mv :

از این فرمان هم برای انتقال و هم برای تغییر نام فایلها و پوشه ها استفاده می شود.

```
mv f1 f2 f3 dir1
```

تغییر مسیر f1 و f2 و f3 از مسیر جاری به dir1

```
mv olddir newdir
```

تغییر نام دایرکتوری از olddir به newdir

```
mv file1 file2
```

تغییر نام فایل از file1 به file2

نکته: در صورتیکه نوع مبدأ و مقصد فرمان mv یکسان باشد (یعنی هر دو فایل باشند یا هر دو پوشه باشند) بشرط یکسان بودن مسیر موردنظر در فرمان، کاربرد تغییر نام برقرار می شود. اما اگر مبدأ فایل و مقصد پوشه باشد، یا مسیر بکار رفته در پارامترهای فرمان متفاوت باشند، کاربرد آن انتقال خواهد بود.

دستور od :

این فرمان نیز برای مشاهده محتویات فایلها بکار می رود، اما با بکارگیری آن همراه با سوئیچ های ذکر شده نتایج زیر حاصل خواهد شد:

- 🍌🍌 -c بایتهای محتویات فایل را بصورت حروف اسکی نشان می دهد.
- 🍌🍌 -d کلمات محتویات فایل را در مبنای ده نشان می دهد.
- 🍌🍌 -o کلمات محتویات فایل را در مبنای هشت (اکتال) نشان می دهد.
- 🍌🍌 x کلمات محتویات فایل را در مبنای شانزده نشان می دهد.

مثال :

```
od -c filename1
```

یافتن فایل ها :

با دستور find می توان موقعیت فایلهای خاصی را که نام و مشخصات دیگری از آنها را می دانیم، بیابیم. پارامترهای این فرمان بدین ترتیب هستند که می توان مسیر جستجو را برای فرمان find تعیین کرد که در صورت عدم ذکر آن، جستجو در کل حافظه سیستم انجام خواهد شد.

پارامتر **type** برای تعیین نوع فایل یا پوشه ای است که بدنبال آن می گردیم. می توان یکی از موارد زیر را بکار برد.

b	: فایل ویژه بلاکی	👉👉
p	: فایل پایپ	👉👉
f	: فایل معمولی	👉👉
l	: فایل لینک	👉👉
d	: پوشه	👉👉

مثال :

```
find /t1 -name black -type d
find -size 10240c
find /t1 -size 10
```

دستور **WC** :

با اجرای این فرمان می توان تعداد حروف، کلمات و خطوط یک یا چند فایل را مشاهده نمود. سوئیچ های این فرمان بترتیب کاربردهای زیر را دارند:

-c	شمارش تعداد حروف	👉👉
-w	شمارش تعداد کلمات	👉👉
-l	شمارش تعداد خطوط	👉👉

مثال :

```
wc filename1 filename2
```

دستور **grep** :

```
grep -[vcln] pattern files
```

این دستور خطوطی از محتوای فایل که شامل عبارت موردنظر هستند را پیدا کرده و روی صفحه نمایش می دهد. چنانچه الگوی موردنظر را در بیش از یک فایل جستجو کنیم خطوط پیدا شده را با ذکر نام فایل روی صفحه نمایش می دهد. پارامترهای این فرمان عبارتند از:

-v	این پارامتر خطوطی را نشان می دهد که الگوی موردنظر در آنها وجود نداشته باشد.	👉👉
-c	این پارامتر تعداد دفعات تکرار یک رشته را در یک پرونده نشان می دهد.	👉👉
-l	این پارامتر باعث می شود فقط اسامی فایلهایی که حاوی رشته هستند نشان داده شود.	👉👉
-n	شماره خط را الگو را قبل از نمایش آن روی صفحه می نویسد.	👉👉

مثال :

```
grep -v "the" .bash
```

pipes:

در بسیاری از موارد لازم است که از خروجی یک دستور بعنوان ورودی دستور دیگر استفاده نمائیم. با استفاده از علامت " | " که **pipe** نامیده می شود کاربر به سیستم عامل می گوید که خروجی یک دستور بعنوان ورودی دستور بعدی در نظر گرفته شود و بدین ترتیب چند دستور با یکدیگر تلفیق می شوند.

برای ایجاد یک سه راهی در میان فرامین **Pipe** می توان از فرمان **tee** استفاده کرد. این فرمان بدون هیچگونه تغییر ورودی استاندارد خود را به خروجی استاندارد منتقل کرده و همزمان یک کپی از داده های ورودی را در فایلهای اختصاص داده شده از سوی کاربر ایجاد می کند.

مثال :

```
command1 | tee file | command2
```



```
cat apple | grep "canvas" | tee mine | wc -l
```

در این مثال تعداد خطوطی از فایل **apple** که حاوی رشته **canvas** باشند بعنوان خروجی نشان داده شده، علاوه بر آن یک کپی از خروجی فرمان **Grep** در فایل **mine** نیز ذخیره می گردد.

دز توضیح فرمان فوق می توان گفت ابتدا فرمان **cat apple** محتوای فایل **apple** را همانگونه که قبلاً آموختید نمایش می دهد. سپس فرمان **grep "canvas"** روی خروجی فرمان قبل اثر کرده و در خطوطی که حاصل از اجرای فرمان قبلی هستند بدنبال آنهایی می گردد که حاوی رشته **canvas** باشند. بنابراین خروجی این فرمان خطوطی از فایل **apple** است که رشته **canvas** در آنها وجود داشته باشد. فرمان نهایی یعنی **wc -l** کارش شمارش تعداد خطوط یک فایل است. ورودی این فرمان، خروجی فرمان قبل است. پس تعداد خطوط فرمان قبل را نشان می دهد و در نتیجه تعداد خطوط فایل **apple** که دارای رشته **canvas** هستند در خروجی نمایش داده خواهد شد.

فرمان **tee** همانگونه که ذکر کردیم یک سه راهی در میان فرامین **Pipe** ایجاد کرده و خروجی حاصل از فرمانهای پیش از خود را در فایل مشخص شده در فرمان **tee** ذخیره می نماید.

۳-۲۰ حساب کاربری

برای ساختن یک **user** از فرمان **useradd** استفاده میکنیم

```
useradd Roohollah
```

با توجه به اینکه لینوکس یک سیستم عامل فوق العاده **secure** است برای **user** نیز باید یک پسورد داشته باشیم. از فرمان **passwd** همراه با نام **user** میتوانیم پسورد را تعیین کنیم

```
passwd Roohollah
```

سپس دودفعه باید پسورد مورد نظر را وارد کنید. توجه کنید که با توجه به **security** بالا لینوکس پسورد حتی با ستاره هم نشان داده نمی شود

اگر بخواهیم یک اطلاعات اضافی مثل نام و نام خانوادگی و یا شماره تلفن به یک **user** اختصاص دهیم از پارامتر **-C** میتوانیم استفاده کنیم

```
useradd -c " Roohollah Pishbahar" Roohollah
```

اگر بخواهیم تغییری در **user** که درست کرده ایم بدهیم از فرمان **usermod** که به معنی **modify** است استفاده کنیم

```
usermod -c " AsreDanesh web site's" Roohollah
```

پروفایل هر **user** جدیدی که درست میکنیم در دایرکتوری **/home** ذخیره میشود. **user** ها ساخته شده همراه با اسم **user**، شماره **UID-User ID**، **GID-Group ID**، **home directory** و **shell** که زمانی که کاربر وارد میشود در گردش میفتد را میتوانیم در شاخه زیر مشاهده کنیم

```
cat /etc/passwd
```

با فرمان زیر میتوانیم **user** مورد نظر را حذف کنیم

```
userdel Roohollah
```

توجه کنید با اجرای فرمان بالا حساب کاربری کاربر حذف میشود اما اگر در دایرکتوری `/home` برویم اسم کاربر را مشاهده می کنیم. به منظور حذف کامل حساب کاربر میتوانیم از فرمان زیر استفاده کنیم

```
userdel -r Roohollah
```

با فرمان زیر می توانیم یک گروه برای کاربران درست کنیم

```
groupadd -r project
```

از فرمان زیر به منظور اضافه کردن کاربر به گروه استفاده میکنیم

```
usermod -G project Roohollah
```

در مسیر زیر کاربرانی که عضو گروه ما هستند قابل دیدن میباشند

```
cat /etc/group
```

با فرمان پایین میتوانیم یک کاربر را عضو دو گروه مختلف کنیم. برای این منظور یک گروه جدید میسازیم سپس کاربر را عضو هر دو گروه می کنیم

```
groupadd -r proj
usermod -G proj,project Roohollah
```

از فرمان زیر در جهت اینکه کاربر عضو کدام گروه است استفاده میشود

```
groups Roohollah
```

برای عوض کردن مالک یک فایل در گروه از فرمان زیر استفاده میکنیم. توجه کنید اسم فایل ۱۲۳ و اسم کاربر Roohollah و اسم گروه project در نظر گرفته شده است

```
chown Roohollah.project 123
```

اگر بخواهیم یک کاربر عادی را مدیر یک گروه کنیم از فرمان زیر استفاده میکنیم

```
gpasswd -A Roohollah project
```

حال کاربری که مدیر فایل شده قابلیت این را دارد که کاربر جدید به گروه اضافه و یا از گروه خارج کند

مدیر گروه جدید از فرمان زیر به منظور اضافه کردن کاربر به گروه استفاده میکند

```
gpasswd -a ali project
```

مدیر گروه جدید از فرمان زیر به منظور حذف کردن کاربر از گروه استفاده میکند

```
gpasswd -d ali project
```

۳-۲۰-۱ دستور **who, users, w**

بوسیله فرمان **who** نام کاربران وارد شده در سیستم نشان داده میشود

who
who am i

پارامترهای دستور **Who**

b - آخرین باری که سیستم بوت شده کی بوده ؟

p - چه پروسس هایی الان فعال هستند ؟

u - زمان هایی که یوزر خاص مورد نظر ما لوگین می شده کی بوده ؟

با استفاده از دستور **w** اطلاعات بیشتری در مورد کاربران نشان داده می شود

w

```
19:51:32 up 6:18, 2 users, load average: 0.44, 0.57, 0.87
```

USER	TTY	FROM	LOGIN@	IDLE	JCPU	PCPU	WHAT
Roohollah	tty7	:0	13:33	6:18m	34:57	0.42s	gnome-session
Roohollah	pts/0	:0.0	19:37	0.00s	0.32s	0.00s	w

توضیح سطر اول: مدت زمانی که کامپیوتر روشن شده-تعداد کاربرها-بارگزاری متوسط توسط پردازنده

توضیح سطر دوم: کاربرانی که وارد سیستم شده اند- مکان ورود- زمان ورود- مدت زمان ورود - میزان اشغال سی پی یو - برنامه در حال اجرا

تفاوت **JCPU** و **PCPU**:

JCPU: زمان کل پردازش هایی که به **tty** ضمیمه شدند (شامل **job** هایی که در گذشته در پس زمینه انجام میشدن همیشه ولی شامل اونایی که الان دارن در پس زمینه لینوکس انجام میشن هست).

PCPU: زمان استفاده شده بوسیله پروسس فعلی که در ستون **what** از خروجی دستور **w** نام پروسس نمایش داده می شود

با استفاده از دستور **users** لیست کاربرانی که مجوز ورود دارند نمایش داده می شود

users

با استفاده از دستور **finger** اطلاعات کاملی راجع به یک کاربر نمایش داده می شود

finger نام کاربر

۳-۲۱ مدیریت پردازش

یک پروسس، یک برنامه در حال اجراست. هر پروسس بوسیله یک شماره واحد شناخته می شود. به این شماره PID یا شماره مشخصه پروسس گفته می شود. پروسسهای شماره صفر و یک ویژه سیستم می باشند. پروسس شماره صفر Kernel سیستم عامل و پروسس شماره یک، پروسس Init نام دارد. این پروسس وظیفه برپاسازی ساختار پروسسها را بر عهده دارد. در سیستم عامل unix تمام پروسسها توسط یک پروسس دیگر ایجاد می شوند که به آن پروسس والد (parent) گویند.

دستور PS :

این فرمان PID و اطلاعات دیگری در مورد پروسسهای در حال اجرا را نمایش می دهد. اگر هیچ سوئیچی را در سطر فرمان به PS اختصاص ندهید اطلاعات زیر را مشاهده خواهید کرد:

👉👉 PID شماره مشخصه پروسس.

👉👉 tty نام ترمینالی که پروسس را اجرا کرده است.

👉👉 Time زمان مصرف شده برای اجرای فرمان.

👉👉 Command نام فرمان

اگر فرمان PS همراه با سوئیچ -p بکار گرفته شود، می توان اطلاعات پروسسهای با شماره خاص را مشاهده کرد.

مثال: ps -p 100,107,267

👉👉 اگر فرمان PS همراه با سوئیچ -f بکار گرفته شود، یک لیست کامل از اطلاعات مختلف درباره پروسسهای فعال را نمایش می دهد. این اطلاعات شامل موارد زیر است:

👉👉 نام کاربری که پروسس را فعال کرده (UID)

👉👉 شماره مشخصه پروسس (PID)

👉👉 شماره پروسس والد (PPID)

👉👉 شماره process utilization

👉👉 ساعت شروع پروسس (stime)

👉👉 ترمینالی که پروسس روی آن فعال شده است.

👉👉 مدت زمانی که اجرای پروسس طول کشیده است. (Time)

👉👉 دستوری که اجرا شده. (command)

بوسیله فرمان **tty** میتوانیم تشخیص دهیم کاربر در کدام ترمینال در حال فعالیت میباشد

tty

۳-۲۲ کار با DVD,CD,Flopy,...

محتویات CD,dvd و یا floppy disk در دایرکتوری media موجود می باشد و برای دسترسی به آن می بایست آن را mount کنیم

مثلا برای دسترسی به CD ROM می بایست دستور زیر را وارد کنیم

mount /media/cdrom

در صورتی که بخواهیم CD را از داخل CD ROM خارج کنیم به دلیل mount بودن این عمل میسر نمی باشد و ابتدا باید CD را un mount کنیم

بدین منظور از فرمان زیر استفاده می کنیم

umount /media/cdrom

برای خارج کردن CD از CD ROM از فرمان **eject** استفاده می کنیم

eject

۳-۲۳ تنظیمات IP در لینوکس

در هر شبکه بطور معمول نخستین لایه یعنی سخت افزار از یک کارت شبکه یا اترنت تشکیل شده و برای اینکه این کارت بعنوان یک رابط در محیط شبکه بکار گرفته شود بایستی اولاً آدرس واحدی تحت عنوان IP address به آن اختصاص یافته و ثانیاً "بسته ها یا packet های اطلاعاتی برای رسیدن به این رابط مسیر دهی شوند که به این عمل routing گفته میشود.

در اغلب توزیعهای لینوکس راههای خاصی برای انجام تنظیمات رابط یا کارت شبکه گنجانده شده که در اینجا روش ساده استفاده از فرامین متنی در محیط shell بررسی می شود :

باید دانست که هسته یا کرنل سیستم عامل لینوکس بطور پیش فرض از یک رابط مجازی یا (lo loopback) interface بعنوان یک ابزار مجازی برای ارتباط با خود سیستم استفاده می کند .

این ابزار مجازی بطور معمول IP 127.0.0.1 و نام localhost را برمی گزیند و در هنگام بوت سیستم بطور از پیش تعریف شده فعال میگردد .

قسمت اصلی سخت افزار شبکه یک رایانه کارت شبکه (NIC(Network Interface Card) می باشد که در محیط گنو/لینوکس معمولاً با اسامی eth0 یا eth1 و ... بسته به تعداد کارتهای شبکه متصل به سیستم آدرس دهی و نامگذاری میشود.

اگر بر روی سیستم ما دو کارت شبکه موجود بود و تمایل داشتیم یکی از آنها را غیرفعال و یا فعال کنیم از فرمانهای زیر استفاده می کنیم

برای غیر فعال کردن از

ifdown eth0

برای فعال کردن از

ifup eth1

برای تنظیم IP یک کارت شبکه با فرض اینکه این کارت اولین و تنها کارت شبکه نصب شده در سیستم است تنها چیزی که لازم داریم برنامه (ifconfig(interface configure) می باشد و می بایست بعنوان مثال از دستور زیر در محیط shell با مجوز کاربر ریشه استفاده کنیم :

ifconfig eth0 192.168.3.8 broadcast 192.168.3.255 netmask 255.255.255.0

همانطوریکه ملاحظه می شود در اینجا یک IP کلاس C شامل آی پی اختصاص یافته به کارت شبکه ۱۹۲,۱۶۸,۳,۸ و یک پوشانه زیر شبکه (subnetmask) کلاس C شامل ۲۵۵,۲۵۵,۲۵۵,۰ به کارت شبکه ما eth0 اختصاص می یابد. و آی پی broadcast نیز شامل IP خاصی است که کلیه کامپیوترهای شبکه به آن پاسخ می دهند و معمولاً آخرین شماره آی پی یک کلاس شبکه می باشد.

در ادامه برای بررسی صحت عملیات انجام شده و عملکرد کارت شبکه از دستور **ifconfig** به تنهایی استفاده کرده و پاسخ سیستم را که چیزی مشابه زیر خواهد بود مشاهده می کنیم :

ifconfig

```
eth0    Link encap:Ethernet  HWaddr 00:21:70:b2:1b:5f
        inet addr:192.168.0.1  Bcast:192.168.0.255  Mask:255.255.255.0
        UP BROADCAST MULTICAST  MTU:1500  Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
        Interrupt:30 Base address:0x2000

lo      Link encap:Local Loopback
        inet addr:127.0.0.1  Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
        UP LOOPBACK RUNNING  MTU:16436  Metric:1
        RX packets:8 errors:0 dropped:0 overruns:0 frame:0
        TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:480 (480.0 B)  TX bytes:480 (480.0 B)
```

از فرمان **ifconfig** برای دیدن و تغییر IP address کارت شبکه که به سیستم ما تعلق گرفته است استفاده می کنیم. در این گزارش حتی آدرس سخت افزاری کارت شبکه eth0 هم مشخص شده 00:21:70:b2:1b:5f که در نوع خود با توجه با سادگی دستورات داده شده نتایج جالبی به نظر میرسد.

برای تغییر آدرس یک کارت شبکه:

```
Ifconfig eth0 192.168.0.1 netmask 255.255.255.0
```

۳-۲۴ پیکربندی پوسته فرمان

برای اینکه بتوانید بطور موثرتری از پوسته فرمان خود استفاده کنید، میتوانید آنرا بنا به خواسته خود تنظیم کنید. برای این منظور باید فایل‌های پیکربندی پوسته فرمان خود را ویرایش کنید.

تعدادی فایل پیکربندی وجود دارد که نحوه رفتار پوسته فرمان شما را تعیین میکند. برخی از این فایل‌ها برای تمام کاربران و پوسته‌ها مشترک بوده و برخی مخصوص یک کاربر خاص هستند. فایل‌های پیکربندی زیر فایل‌هایی هستند که هر کاربر پوسته فرمان در لینوکس از آنها استفاده میکند :

- `etc/profile` : این فایل اطلاعات محیط کاربری هر کاربر را ذخیره میکند. این فایل هنگامی اجرا میشود که شما به سیستم وارد شده و پوسته فرمان آغاز به کار میکند. این فایل مقادیر پیش‌گزینه مسیر، شکل اعلان فرمان، حداکثر تعداد فایلی که شما میتوانید ایجاد کنید و مجوزهای پیش‌گزینه برای فایل‌هایی که ایجاد میکنید را تعیین میکند. همچنین این فایل متغیرهای محیطی مانند محل صندوق پستی و اندازه فایل‌های تاریخچه را تنظیم میکند.
- `etc/bashrc` : این فایل برای هر کاربری که پوسته `bash` را اجرا میکند، اجرا میشود. این فایل حالت اعلان فرمان را تنظیم میکند. مقادیر این فایل میتواند توسط فایل `bashrc` که در دایرکتوری خانگی هر کاربر وجود دارد، تحت تاثیر قرار گیرد.
- `~/bashrc` : این فایل حاوی اطلاعات مربوط به `bash` هر کاربر میباشد. این فایل هنگامی خوانده میشود که به سیستم وارد میشود و هر گاه که یک پوسته جدید باز میکنید. اینجا بهترین مکان برای ذخیره متغیرهای محیطی و فرمانهای مستعار خاص خودتان است.
- `~/bash_profile` : این فایل برای وارد کردن اطلاعات خاصی که هر کاربر در استفاده از پوسته بکار میبرد میباشد. این فایل تنها یکبار اجرا میشود. هنگامی که کاربر به سیستم وارد میشود. این فایل تعدادی از متغیرهای محیطی را مقداردهی کرده و فایل `bashrc` مربوط به کاربر را اجرا میکند.
- `~/bash_logout` : این فایل هر گاه که شما از سیستم خارج میشوید اجرا میشود. این فایل فقط صفحه نمایش را پاک میکند.

برای تغییر فایل‌های `etc/profile` و `etc/bashrc` باید با کاربر ریشه وارد سیستم شده باشید. هر کاربر میتواند اطلاعات موجود در فایل‌های `bash_profile`، `bashrc` و `bash_logout` موجود در دایرکتوری‌های خود را تغییر دهد.

۳-۲۵ نوشتن برنامه در محیط لینوکس

نوشتن برنامه در سیستم عامل لینوکس همانند سیستم عامل ویندوز می‌باشد با این تفاوت که سیستم عامل لینوکس خود حاوی کامپایلر `C++` می‌باشد.

برای نوشتن برنامه باید ابتدا در یک محیط برنامه را ویرایش کرد و بعد با پسوند `CPP`، آن را ذخیره کرد. محیط‌های متفاوتی در لینوکس برای نوشتن برنامه موجود می‌باشد. یکی از این محیط‌ها `gedit` می‌باشد. برای دستیابی به این محیط در ترمینال (Terminal) کلمه `gedit` را تایپ کرده و کلید `Enter` را فشار دهید. یکی از قابلیت‌های این محیط تنظیم محیط با زبان مورد نظر که در اینجا `C++` می‌باشد. بعد از این تنظیم کامپایلر کلمات کلیدی و رزرو شده را تشخیص داده و به نوشتن برنامه کمک خواهد کرد اگر سیستم عامل لینوکس شما حاوی محیط گرافیکی نباشد می‌توانید بجای `gedit` از محیط `Vi` استفاده نمایید. برای دستیابی به این محیط از دستور `Vi` در ترمینال استفاده نمایید.

بعد از آن که برنامه را با پسوند `CPP` نمودیم. باید آن را با دستور زیر کامپایل نماییم.

`g++ CPP.اسم برنامه`

مثلا اگر برنامه به نام `exam` م یباشد بصورت زیر کامپایل می‌شود.

`G++ exam.CPP`

یک برنامه خیلی ساده به طور مثال برنامه خوش آمد گویی را تایپ کنید . مثال زیر یک برنامه بسیار ساده می باشد .

```
#include

Int main()

{

    Cout << "welcome to c++";

    Returne 0;

}
```

سپس این برنامه را با نام دلخواه و با پسوند **cpp** ذخیره کرده و از محیط ادیتور خارج شوید . مرحله کد نویسی تمام شده حالا باید برای برنامه نوشته شده را کامپایل و اجرا کنید. برای این کار یک کنسول یا ترمینال باز کنید . برای کامپایل کردن برنامه از دستور **g++** استفاده می کنیم.

```
g++ -o t2 test1.cpp
```

دستور مقابل از چند ستون تشکیل شده است . ستون اول **g++** مربوط به دستور کامپایل کردن می باشد. در ستون دوم سوئیچ **-o** برای ساختن فایل خروجی یا اجرای بعد از کامپایل کردن مورد استفاده قرار گرفته است. در ستون سوم **t2** اسم فایلی است که بعد از کامپایل کردن ایجاد می شود و یک فایل اجرایی می باشد . و در ستون آخر **test1.cpp** اسم فایلی است که شما برنامه **c++** را در اون ذخیره کردید بعد از اجرای دستور بالا ابتدا برنامه شما کامپایل شده و سپس یک فایل با اسم **t2** ساخته میشود .

نکته: اگر در برنامه شما خطا وجود داشته باشد فایلی برای اجرا ساخته نمیشود و در مرحله کامپایل شماره خطهای دارای خطا را نمایش می دهد . اگر برنامه بدون خطا اجرا شود کامپایل به صورت زیر در کنسول اجرا شده و یک فایل ساخته می شود.

```
G++ -o t2 test1.cpp
```

```
In file included from /usr/lib/gcc/i386-redhat-
```

```
Linux/3.4.2/../../../../include/c++/3.4.2/backward/iostream.h:31;
```

```
From test1.cpp:1:
```

```
/usr/lib/gcc/1386-redhatg-
```

```
Linux/3.4.2/../../../../include/c++/3.4.2/backward-warning.h:32:2:warning:#warning this
file includes at last one deprecated or antiquated header.please
```

Consider using one of the 32 headers found in section 17.4.12 of the c++ standard.

Examples include substituting the header for the header for c++ includes ,or instead of the

Deprecated header. To disable this warning use-Wno-deprecated.

```
Test1.cpp:6:2: warning:no newline at end of file
```


حالا می توانید فایل ساخته شده را اجرا کنید و نتیجه کار را ببینید.

./t2

Welcome to c++

ستون اول t2. برنامه را به اجرا در می آورد.

مشاهده می کنید که بعد از اجرای برنامه پیغام خوش آمد گویی c++ welcome to را چاپ می کند .

۳-۲۲ سرانجام پنگوئن محبوب توروالدز چه خواهد شد؟



«ابتدا تورانادیده میگیرند، سپس مسخره‌ات میکنند و بعد با تو می‌جنگند. ولی در نهایت پیروزی از آن توست»

«کاذبی»

۳-۲۳ دستور کار:

۱. سعی کنید که به شاخه‌های /root و /home/user2 وارد شوید و محتویات آنها را مشاهده کنید. هر دفعه با چه پیامی مواجه می‌شوید؟ همین کار را در مورد شاخه‌های /bin و /lib انجام دهید.
۲. در محیط وارد شده و با امتحان کردن دستورات و برنامه‌های زیر با کار هریک آشنا شوید:
--help , help , info, man , pwd , mv, rm , mkdir , cd , cp , ls, who, chmod , grep tar , mc , mcedit , wget , make , gcc , ifconfig
۳. تفاوت دستورهای -help و help و info و man را توضیح دهید.
۴. فرمان ls دارای گزینه‌هایی است که برخی از آنها عبارتند از : -a و -l . از راهنمای سیستم عملکرد هر کدام از گزینه‌های زیر را بیابید و آنها را آزمایش کنید. سپس نتیجه هریک را در کاربرگ خود بیاورید.
۵. با فرمان mkdir یک شاخه جدید ایجاد کنید. توسط پارامتر -p این فرمان سعی کنید شاخه و زیرشاخه جدیدی را همزمان ایجاد کنید. دستور را در کاربرگ یادداشت نمایید.
۶. دستور who|grep john? چه عملی را انجام می‌دهد؟
۷. تفاوت دستور who|sort و who>sort در چیست؟
۸. تفاوت make depend و make را توضیح دهید.
۹. عملکرد تابع getpid را از صفحات راهنما به وسیله دستور man بیابید. با استفاده از این دستور در پرونده pid.c برنامه‌ای بنویسید که شماره PID خود را چاپ کند. این برنامه را با استفاده از gcc ترجمه و به دفعات اجرا کنید. چه نتیجه‌ای می‌گیرید؟

۳-۲۴ سوالات

۱. لینوکس و ویندوز را باهم مقایسه کنید(نقاط ضعف و قوت لینوکس نسبت به ویندوز؟)
۲. آیا می‌توان در محیط لینوکس با محیط دات نت برنامه نویسی و سپس آن را اجرا کرد؟ پاسخ خود را توضیح دهید.

۳-۲۵ منابع:

www.technotux.org
www.farsilinux.org
www.linuxdoc.org
www.iranitux.com
www.ospdev.net
www.gnome.org
www.IRITN.com
tafazoli.iut.ac.ir
www.atcce.com
www.linux.com
www.linux.org
www.subnet.ir
www.suma.ir



Linux
many flavors, many choices. Freedom

تحقیق

چگونه می‌توان بخشی از سیستم عامل لینوکس را فارسی نمود؟ مزیت‌ها و معایب این عمل را بیان کنید.