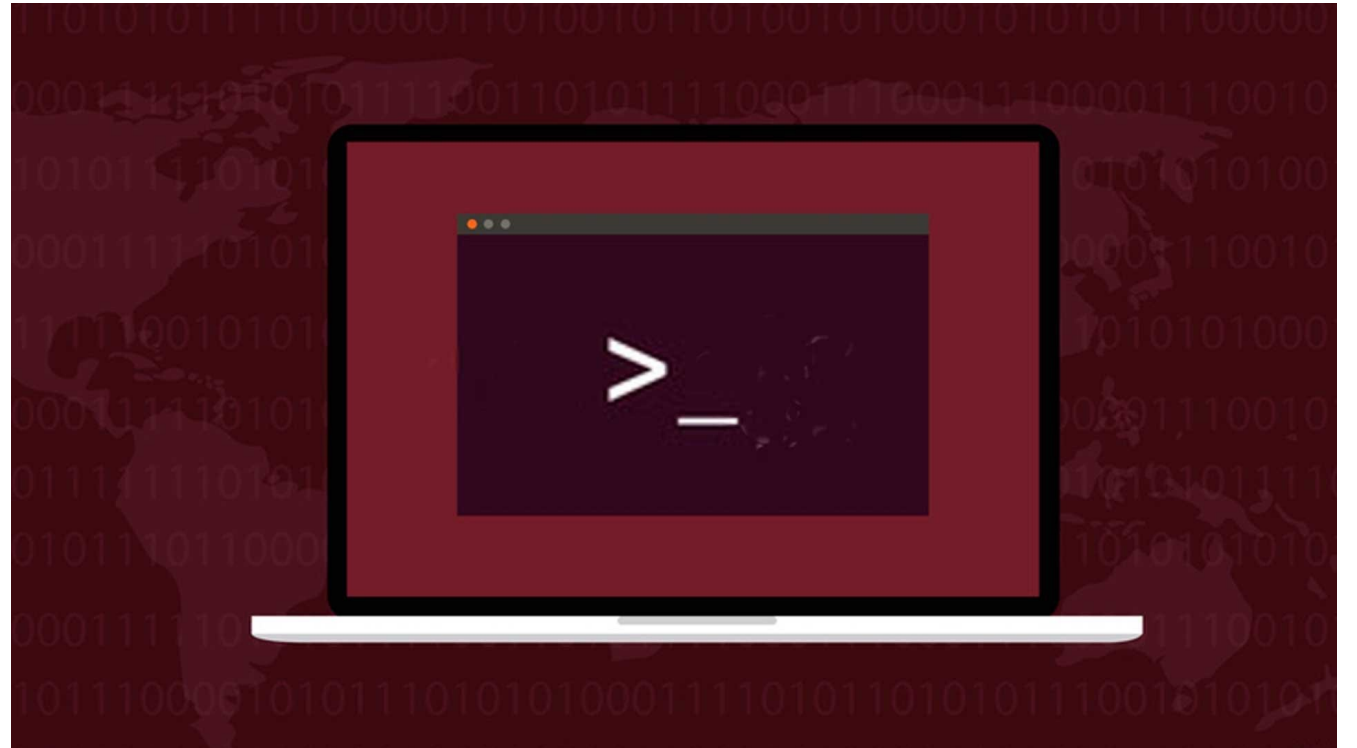


Açık Kaynak İşletim Sistemleri

Arş. Gör. Gökçe OĞUZ ERENLER

LINUX KOMUT SATIRI

- Bash ve Linux Komut Satırı
- Yardım Alma ve Man Kılavuz Sayfaları
- Bilgi Alma Komutları
- Dizinler/Klasörler Arasında Gezme ve İçeriğini Listeleme
- Dosya İşlemleri



BÖLÜM 1

Bash ve Linux Komut Satırı

Bash ve Linux Komut Satırı

- ✓ Linux işletim sistemini kullanacaksanız, öğrenmeniz gereken ilk kavram Linux **terminali** (konsol) ve **Linux komut satırı** kavramlarıdır.
- ✓ Kernel, bir işletim sisteminin kalbidir. En önemli görevi, donanımla yazılımın haberleşmesini sağlamaktır.
- ✓ Önemli bir başka görevi de sistem belleğini ve sistemde çalışan süreçleri (**process**) denetlemektir.
- ✓ Tüm eylemleri, kullanıcıdan gelen girdilerle birlikte sistemin işlemesi için kullanıcıyla sistem arasında bir aracı olmalıdır.
- ✓ Grafikselsel bir arayüz (GUI) de kullansak, sadece komutlarla da çalışsak fark etmez yapılan her işlem aslında arka planda komutların yorumlanıp çalıştırılması ile olmaktadır.
- ✓ Linux sistemlerde bu işi komut satırı dediğimiz ve bir **kabuk (shell)** programını temel alarak çalışan yapı sayesinde gerçekleştiririz.

Shell (Kabuk)

- ✓ Shell bir kabuk programıdır.
- ✓ **Görevi;** kernel ile kullanıcı arasında yer alarak, kullanıcıdan gelen komutları yorumlayıp kernele iletmektir.
- ✓ Kernel ile shell'in bu etkileşimi de sistem çağrıları (system calls) sayesinde olmaktadır.
- ✓ Linux sistemlerde birçok kabuk (shell) programı kullanılmaktadır fakat günümüzde en çok tercih edilen BASH kabuk programıdır.
- ✓ Sistemimizde kullanılan kabuk programını öğrenmek için echo \$SHELL komutunu kullanabiliriz.

```
ubuntu@ubuntu-VirtualBox:~$ echo $SHELL
/bin/bash
ubuntu@ubuntu-VirtualBox:~$
```

Ortam Değişkenleri(Çevre Değişkenleri)

- ✓Konsoldan girilen bir komutu çalıştırmak için sırasıyla bazı dizinlere bakılması gerekir.
- ✓Eğer verilen komutun çalıştırılabilir dosyası bu dizinlerin altındaysa çalıştırılır. Yani yola (PATH) ekliyse çalıştırılır, yoksa çalıştırılmaz.
- ✓Aşağıda görülen ve birbiriyle : işaretiyle ayrılmış ifadelerin oluşturduğu yapıya yol (PATH) ortam değişkeni denir.
- ✓Kullanacağımız komut **echo \$PATH** komutudur.

```
ubuntu@ubuntu-VirtualBox:~$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
```

Ortam Değişkenleri(Çevre Değişkenleri)

- ✓ Çalışma ortamınızda ortam değişkenlerini görmek için **printenv** ya da **env** komutunu kullanabilirsiniz.

```
ubuntu@ubuntu-VirtualBox:~$ env
SHELL=/bin/bash
SESSION_MANAGER=local/ubuntu-VirtualBox:@/tmp/.ICE-unix/1027,unix/ubuntu-VirtualBox:/tmp/.ICE-unix/1027
QT_ACCESSIBILITY=1
COLORTERM=truecolor
XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/etc/xdg
XDG_MENU_PREFIX=gnome-
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
GNOME_SHELL_SESSION_MODE=ubuntu
SSH_AUTH_SOCK=/run/user/1000/keyring/ssh
XMODIFIERS=@im=ibus
DESKTOP_SESSION=ubuntu
SSH_AGENT_PID=977
GTK_MODULES=gail:atk-bridge
PWD=/home/ubuntu
LOGNAME=ubuntu
XDG_SESSION_DESKTOP=ubuntu
XDG_SESSION_TYPE=x11
GPG_AGENT_INFO=/run/user/1000/gnupg/S.gpg-agent:0:1
XAUTHORITY=/run/user/1000/gdm/Xauthority
WINDOWPATH=2
HOME=/home/ubuntu
USERNAME=ubuntu
IM_CONFIG_PHASE=1
LANG=tr TR.UTF-8
```

Komut çıktısında da görülen ortam/çevre ya da **kabuk değişkenleri** ne ifade ediyor ?

PATH: Bir komut verildiğinde, komut programını oluşturan dosyanın aranacağı dizinler.

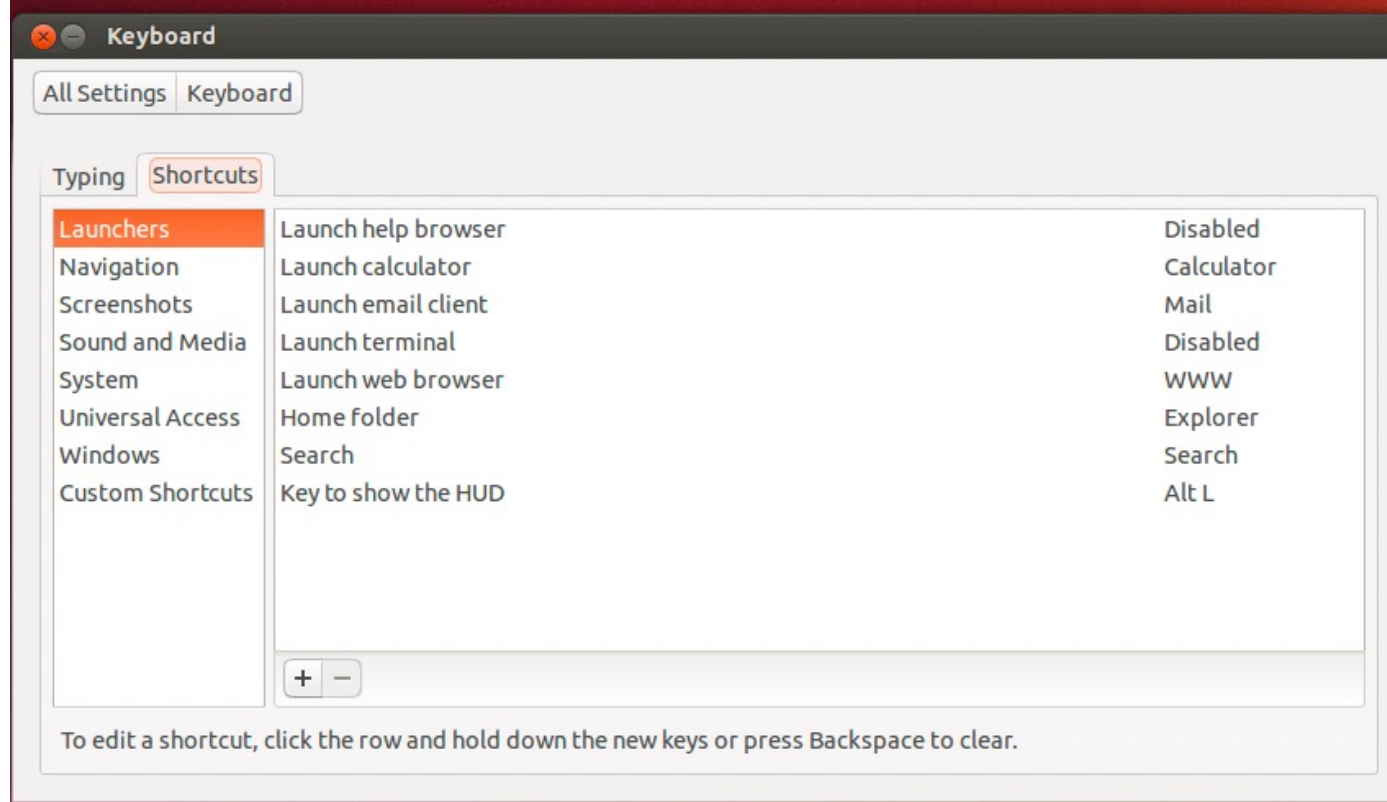
HOME: Kullanıcının kişisel dizininin (**ev dizini**) adını içeren değişken.

TERM: Grafik ekran görüntülerinin gönderileceği X sunucusu ile ilgili değişken.

SHELL: O anda çalışmakta olan kabuk programının adı ve yeri.

Terminal İçin Kısayol Tuşları Atama

✓ **Amaç;** konsolu ve dolayısıyla Linux sistemi verimli ve kolay bir şekilde kullanmaktır.



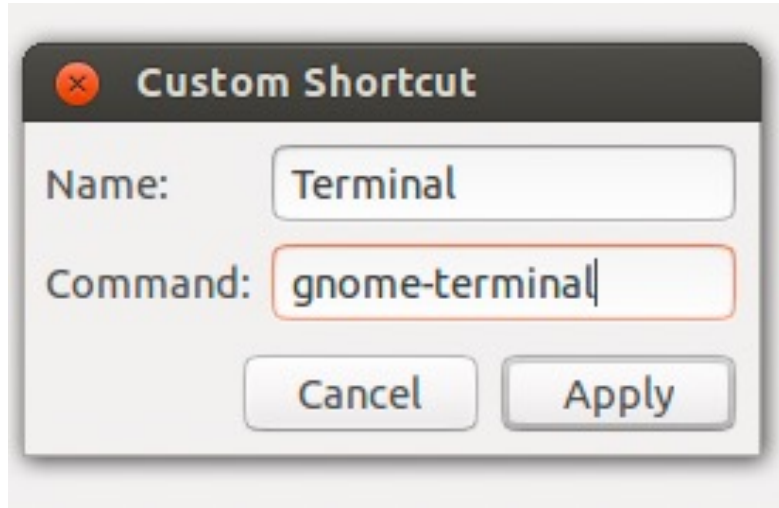
✓ Öncelikle, terminalin açılması için kullanabileceğimiz bir kısayol tuş kombinasyonu oluşturabiliriz.

✓ Bunun için sistem ayarları menüsünden klavyeyi (keyboard) seçiyoruz ve oradan da kısayollar menüsünü açıyoruz.

✓ Yeni kısayolu düzenlemek için, alt tarafta bulunan + işaretine tıklıyoruz.

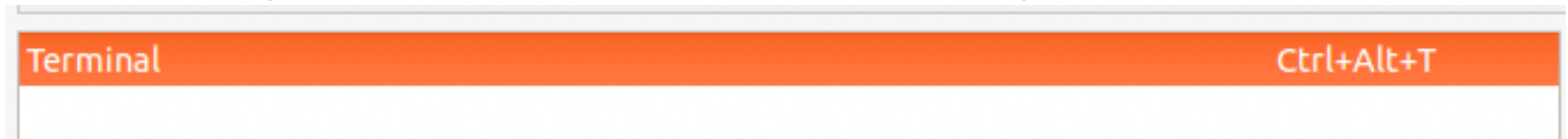
Terminal İçin Kısayol Tuşları Atama

- ✓ Açılan kutucuğu aşağıdaki gibi düzenliyoruz.



- ✓ Bu şekilde düzenledikten sonra da Ekle butonuna basıyoruz.
- ✓ Bu işlemi yaptıktan sonra ana ekran açılır ve orada Terminal ve Kapalı yazan bir satır yer aldığını görürüz.

- ✓ O satırın üzerine sağ tıklayarak çıkan New accelerator., uyarısından sonra ctrl+alt+T tuş takımını kullanırız ve terminal kısayolunu bu tuşlara atamış oluruz.
- ✓ Artık bu tuş kombinasyonunu kullanarak Linux terminalini kolayca açabiliriz.



Terminal İin Kısayol Tuşları Atama

alt+F2

- ✓ Programları alıřtırabileceğimiz komutları kullanabilmemizi saėlayan bir kutucuk aan alt+F2 tuşlarını kullanarak ve aılan kutucuėa gnome-terminal yazarak da komut satırını aabiliriz.
- ✓ Aslında ctrl+alt+T tuşlarına atadığımız kısayol da bu komuta ait kısayoldur.



Bash Shell Kısayollar

KISAYOL	GÖREVİ	KISAYOL	GÖREVİ
ctrl+A	İmleci satır başına götürür.	ctrl+K	İmlecin sağındaki her şey silinir.
ctrl+E	İmleci satır sonuna götürür.	esc+T	İmleçten önceki iki kelimeyi yer değiştirir.
ctrl+B	İmleç bir karakter geri gider.	ctrl+C	Komutu keser.
ctrl+F	İmleç bir karakter ileri gider.	ctrl+Z	Çalışan uygulamayı arka plana atar, bg komutu ile sürecin (process) arka planda çalışması devam eder, fg komutu ile de öne alınabilir.
alt+B	İmleç bir kelime geri gider.	ctrl+D	Terminali sonlandırır. (exit komutuyla aynıdır)
alt+F	İmleç bir kelime ileri gider.	ctrl+H	Backspace tuşu gibi, sola doğru karakterleri siler.
ctrl+XX	İmleç bulunduğu yer ile satır başı arasında gider gelir.	ctrl+R	Daha önce kullanılan komutlar arasında arama ve yeniden o komutu kullanma gibi işlemleri yapmanızı sağlar.
ctrl+L	Ekranı temizler ve imleç en üst satıra çıkar, (clear komutunun aynı)	ctrl+K	İmlecin sağındaki her şey silinir.
ctrl+U	İmlecin solundaki her şey silinir.	esc+T:	İmleçten önceki iki kelimeyi yer değiştirir.
Tab	Otomatik tamamlama sağlar. Bu durum hem dosya/dizin hem de komutlar için geçerlidir. Örneğin kullanacağınız komutu tam hatırlayamadınız ve ilk birkaç harfini yazdınız. Tab tuşuna bir kere bastığınızda o komutu tamamlayacaktır. Eğer tab tuşuna üst üste iki kez basarsanız bu sefer de o yazdığınız harflerle başlayan olası tüm komutlar çıkacaktır.		

Komut Satırında Aynı Anda Birden Fazla Komut Kullanımı

- ✓ Terminalde/konsolda aynı anda birden fazla komut kullanılabilir.
- ✓ Bunun için kullanılacak komutların noktalı virgül ile ayrılması yeterlidir.
- ✓ Örneğin **ls;ls /etc** komutunun çıktısı aşağıdaki gibi olacaktır.

```
ubuntu@ubuntu-VirtualBox:~$ ls;ls /etc
Belgeler      File_A  gokce      Müzik      Şablonlar
deneme        File_B  İndirilenler  Resimler  Videolar
facebook.com  Genel  Masaüstü   snap       www.gsstore.org
acpi
adduser.conf
alsa
alternatives
anacrontab
apache2
apg.conf
apm
apparmor
apparmor.d
appport
appstream.conf
apt
avahi
bash.bashrc
bash_completion
bash_completion.d
bindresvport.blacklist
binfmt.d
bluetooth
hostid
hostname
hosts
hosts.allow
hosts.deny
hp
ifplugd
init
init.d
initramfs-tools
inputrc
insserv.conf.d
ipp-usb
iproute2
issue
issue.net
kernel
kernel-img.conf
kerneloops.conf
ldap
profile
profile.d
protocols
pulse
python3
python3.8
rc0.d
rc1.d
rc2.d
rc3.d
rc4.d
rc5.d
rc6.d
rcS.d
resolv.conf
resolve.conf
rmt
rpc
rsyslog.conf
rsyslog.d
```

Komut Satırında Aynı Anda Birden Fazla Komut Kullanımı

- ✓ Terminalde birden fazla komut çalıştırmak için && operatörü de kullanılabilir.
- ✓ Bu operatör sistem güncelleme sırasında çok kullanılır.
- ✓ Örneğin **apt-get update && apt-get upgrade** komutu verildiğinde sırasıyla komutlar çalıştırılarak sistem güncellenecektir.
- ✓ **ls && cd /etc** komutunun çıktısı aşağıdaki gibi olacaktır.

```
ubuntu@ubuntu-VirtualBox:~$ ls && cd /etc
Belgeler      File_A  gokce      Müzik      Şablonlar
deneme        File_B  İndirilenler Resimler   Videolar
facebook.com  Genel  Masaüstü   snap       www.gsstore.org
ubuntu@ubuntu-VirtualBox:/etc$
```

- ✓ Bir diğer operatörümüz de || operatörüdür.
- ✓ Terminalde iki komut kullanırken komutlar arasına bu operatörü koyarsak, birinci komut başarısız olursa diğer komutu çalıştırmasını istediğimizi söylemiş oluruz.
- ✓ Örnek olarak **abc || cd /home** komutuna bakalım.

```
ubuntu@ubuntu-VirtualBox:/$ abc || cd /home
Command 'abc' not found, did you mean:
  command 'ab' from deb apache2-utils (2.4.46-1ubuntu1)
  command 'atc' from deb bsdgames (2.17-28build1)
  command 'asc' from deb asc (2.6.1.0-7build2)
  command 'axc' from deb afnix (2.9.2-2build1)
  command 'ajc' from deb aspectj (1.9.2-1)
  command 'arc' from deb arc (5.21q-8)
  command 'arc' from deb arcanist (0-git20190207-1)
  command 'bc' from deb bc (1.07.1-2build2)
  command 'alc' from deb amule-utils-gui (1:2.3.2+git20200530.3a77afb-1)
  command 'abx' from deb abx (0.0~b1-1build1)
  command 'cbc' from deb coinor-cbc (2.10.5+ds1-1)
  command 'aec' from deb libaec-tools (1.0.4-1)
  command 'ac' from deb acct (6.6.4-3)
  command 'abe' from deb abe (1.1+dfsg-3)
  command 'nbc' from deb nbc (1.2.1.r4+dfsg-9)
Try: sudo apt install <deb name>
ubuntu@ubuntu-VirtualBox:/home$
```

ignoreeof Ayarı

- ✓ Bash kabuğunun özelliklerinden biri de kabuktan çıkmak için kullanılan CTRL+D tuş takımıdır.
- ✓ Bu tuşlara bastığımızda kabuk kendini sonlandırır ve kabuktan çıkmış oluruz.
- ✓ Bu şekilde kabuktan yanlışlıkla çıkılmasını önlemek için yapılabilecek bir düzenleme vardır. Bu düzenlemeyi ignoreeof ile yaparız.
- ✓ Ignoreeof düzenlemesi, kabuktan CTRL+D tuş takımıyla çıkılmasını engeller.
- ✓ Bu düzenlemeyi birkaç farklı şekilde yapabiliriz.

ignoreeof Ayarı

- ✓ Örneğin `set -o ignoreeof` komutuyla düzenlemeyi yapabilir, geri eski haline getirmek için de `set +o ignoreeof` komutunu kullanabiliriz.
- ✓ Diğer bir komutumuz da `export IGNOREEOF=2` komutudur. Eski haline getirmek için de `export IGNOREEOF=0` kullanılabilir.
- ✓ Son yöntem de `ignoreeof=2` komutunu kullanmaktır.
- ✓ Bu düzenleme ile art arda iki kere `CTRL+D` tuş takımına basılırsa kabuktan çıkılmasını söylemiş oluyoruz.

ignoreeof Ayarı

✓ set -o ignoreeof düzenlemesi yapıldığında, sadece exit komutuyla kabuğu sonlandırabileceğimiz uyarısını verir.

```
ubuntu@ubuntu-VirtualBox:/$ set -o ignoreeof
ubuntu@ubuntu-VirtualBox:/$ Use "exit" to leave the shell.
ubuntu@ubuntu-VirtualBox:/$ Use "exit" to leave the shell.
ubuntu@ubuntu-VirtualBox:/$ Use "exit" to leave the shell.
ubuntu@ubuntu-VirtualBox:/$
```

✓ Diğer yöntem;

```
root@ubuntu-VirtualBox:~# ignoreeof=2
root@ubuntu-VirtualBox:~# Use "logout" to leave the shell.
root@ubuntu-VirtualBox:~# Use "logout" to leave the shell.
root@ubuntu-VirtualBox:~# logout
ubuntu@ubuntu-VirtualBox:~$
```

Bu düzenlemeyi yaptıktan sonra iki kere CTRL+D tuş takımına basınca yukarıdaki uyarılar çıkıyor, üçüncü kez aynı tuş takımını kullandığımızda kabuğu sonlandırıyor.

noclobber Tanımlaması

- ✓ Tanımlar küçük harflerle yazılır, sebebi de büyük harflerle yazılan değişkenlerden ayrılımları içindir.
- ✓ Bir tanımlı işletmek için `set -o` tanım, kaldırmak için de `set +o` tanım kullanılır.
- ✓ Bu tanımlardan biri olan `noclobber` şu işe yarar: Bir komutun çıktısı bir dosyaya gönderildiğinde, dosyanın eski içeriği aynı kalır.
- ✓ Bu tanım sayesinde yanlışlıkla bir dosyanın üzerine başka bilgilerin yazılması önlenir.

Örnek olarak;

- ✓ İlk önce **touch** komutuyla bir belge oluşturalım ve bu belgenin içine **echo** yardımıyla bir ifade yazalım.
- ✓ Sonra da **cat** komutuyla dosyayı okuyup kontrol edelim.

```
ubuntu@ubuntu-VirtualBox:~$ touch important
ubuntu@ubuntu-VirtualBox:~$ echo "test" > important
ubuntu@ubuntu-VirtualBox:~$ cat important
test
ubuntu@ubuntu-VirtualBox:~$
```

- ✓ Sonra **noclobber** tanımlamasını aktif edelim ve dosyada değişiklik yapıp yapamadığına bakalım.

```
ubuntu@ubuntu-VirtualBox:~$ set -o noclobber
ubuntu@ubuntu-VirtualBox:~$ echo "deneme" > important
bash: important: cannot overwrite existing file
ubuntu@ubuntu-VirtualBox:~$ set +o noclobber
ubuntu@ubuntu-VirtualBox:~$ echo "test2" > important
ubuntu@ubuntu-VirtualBox:~$ cat important
test2
ubuntu@ubuntu-VirtualBox:~$
```

BÖLÜM 2

Yardım Alma ve Man Kılavuz Sayfaları

help

- ✓ ls komutunun kullanımıyla ilgili bilgi almak için **ls --help** komutu kullanılabilir.

```
ubuntu@ubuntu-VirtualBox:~$ ls --help
Usage: ls [OPTION]... [FILE]...
List information about the FILES (the current directory by default).
Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.

Mandatory arguments to long options are mandatory for short options too.
  -a, --all                do not ignore entries starting with .
  -A, --almost-all        do not list implied . and ..
      --author              with -l, print the author of each file
  -b, --escape             print C-style escapes for nongraphic characters
      --block-size=SIZE    with -l, scale sizes by SIZE when printing them;
                           e.g., '--block-size=M'; see SIZE format below
  -B, --ignore-backups     do not list implied entries ending with ~
  -C                       with -lt: sort by, and show, ctime (time of last
                           modification of file status information);
                           with -l: show ctime and sort by name;
                           otherwise: sort by ctime, newest first
  -C                       list entries by columns
      --color[=WHEN]       colorize the output; WHEN can be 'always' (default
                           if omitted), 'auto', or 'never'; more info below
```

help

✓ **help <komut>** şeklinde de yardım alınabilir.

```
or available locally via: info (coreutils) is invocation
ubuntu@ubuntu-VirtualBox:~$ help cd
cd: cd [-L|[-P [-e]] [-@]] [dir]
    Change the shell working directory.

    Change the current directory to DIR.  The default DIR is the value of the
    HOME shell variable.

    The variable CDPATH defines the search path for the directory containing
    DIR.  Alternative directory names in CDPATH are separated by a colon (:).
    A null directory name is the same as the current directory.  If DIR begins
    with a slash (/), then CDPATH is not used.

    If the directory is not found, and the shell option `cdable_vars' is set,
    the word is assumed to be a variable name.  If that variable has a value,
    its value is used for DIR.
```

```
ubuntu@ubuntu-VirtualBox:~$ help ls
bash: help: no help topics match `ls'.  Try `help help' or `man -k ls' or `info
ls'.
ubuntu@ubuntu-VirtualBox:~$
```

- Fakat her komutun bu komut kalıbına uyan yardım konusu olmayabilir.
- Örneğin biraz önce **ls --help** şeklinde komut kullandık ve işe yaradı.
- Ancak diğer kullanımda yardım bilgisi olmayabilir.

man (Manuel Pages) Kılavuz Sayfaları

- Man sayfaları Linux sistemde temel yardım alma dosyalarıdır.
- Bir man sayfasının yapısı genellikle şu şekildedir;
 - **NAME:** Komutun ismi ve açıklaması.
 - **SYNOPSIS:** Komutun nasıl kullanılacağı.
 - **DESCRIPTION:** Komutun fonksiyonu hakkında açıklama.
 - **EXAMPLES:** Komutun nasıl kullanılacağıyla ilgili örnekler.
 - **SEE ALSO:** İlgili başlıklar.
- Kılavuz sayfaları, /usr/share/man altında bulunur.

```
ubuntu@ubuntu-VirtualBox:/usr/share/man$ ls
cs  es  fr.ISO8859-1  id  ko  man3  man6  nl  pt_BR  sr  zh_CN
da  fi  fr.UTF-8      it  man1  man4  man7  pl  ru     sv  zh_TW
de  fr  hu           ja  man2  man5  man8  pt  sl     tr
ubuntu@ubuntu-VirtualBox:/usr/share/man$
```


man (Manuel Pages) Kılavuz Sayfaları

➤ Kılavuz sayfalarının yapılandırması şu şekildedir:

- **man1:** Genel kullanıcı programları.
- **man2:** Sistem programları.
- **man3:** Kütüphane fonksiyonları. (C programlama ile ilgili)
- **man4:** Özel dosyalar.
- **man5:** Dosya biçimleri.
- **man6:** Oyunlar ve ekran koruyucular.
- **man7:** Diğer kategorilere girmeyen çeşitli komutlar.
- **man8:** Sistem yönetimi ve bakım.

whatis

- ✓ Bu komutu kullanarak bir uygulama ya da komutla ilgili yardımın hangi kılavuz sayfalarında olduğunu öğrenebiliriz.

```
ubuntu@ubuntu-VirtualBox:/$ whatis echo
echo (1) - display a line of text
ubuntu@ubuntu-VirtualBox:/$
```

- ✓ Joker karakter kullanarak sorgulama yapmak için -w parametresi kullanılmalıdır.
- ✓ Örneğin `whatis -w cal*` gibi.

```
ubuntu@ubuntu-VirtualBox:/$ whatis -w cal*
calibrate_ppa (8) - pnm2ppa calibration tool
ubuntu@ubuntu-VirtualBox:/$
```

BÖLÜM 3

BİLGİ ALMA KOMUTLARI

lsb_release

- ✓ Kullandığımız Linux dağıtımın adını öğrenmek için lsb_release -a komutunu kullanabiliriz.

```
ubuntu@ubuntu-VirtualBox:/$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:   Ubuntu 20.10
Release:      20.10
Codename:     groovy
ubuntu@ubuntu-VirtualBox:/$
```

- ✓ Benzer bir komut da cat /etc/*release komutudur. Bu komutla sistemimizle ilgili biraz daha ayrıntılı bilgi alabiliriz.

```
ubuntu@ubuntu-VirtualBox:/$ cat /etc/*release
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=20.10
DISTRIB_CODENAME=groovy
DISTRIB_DESCRIPTION="Ubuntu 20.10"
NAME="Ubuntu"
VERSION="20.10 (Groovy Gorilla)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 20.10"
VERSION_ID="20.10"
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
VERSION_CODENAME=groovy
UBUNTU_CODENAME=groovy
```

etc/issue

- ✓ Kullandığımız Linux'un adını görmek için `cat /etc/issue` komutu da kullanılabilir.

```
ubuntu@ubuntu-VirtualBox:/$ cat /etc/issue
Ubuntu 20.10 \n \l
```

uname

- ✓ `uname -a` komutu ile kernel (çekirdek) versiyonunu, kernel adı, hostname, çekirdek mimarisi gibi bilgilerin hepsi görülebilir.

```
ubuntu@ubuntu-VirtualBox:/$ uname -a
Linux ubuntu-VirtualBox 5.8.0-48-generic #54-Ubuntu SMP Fri Mar 19 14:25:20 UTC
2021 x86_64 x86_64 x86_64 GNU/Linux
```

date

- ✓ Sistemin o anki saat ve tarih bilgisi öğrenmek için **date** komutu kullanılır.

```
ubuntu@ubuntu-VirtualBox:/$ date
Paz 04 Nis 2021 22:54:57 +03
```

hostname

- ✓ Kullandığımız sistemde bilgisayara verdiğimiz adı yani host adını görmek için **hostname** komutunu kullanabiliriz.

```
ubuntu@ubuntu-VirtualBox:/$ hostname
ubuntu-VirtualBox
```

who-whoami

- ✓ Sistemde kim aktif, kim login olmuş, o an hangi kimlikle çalışıyoruz gibi bilgileri öğrenmek için **w**, **who**, **whoami**, **who am i** komutlarını kullanabiliriz.

```
ubuntu@ubuntu-VirtualBox:/$ w
 23:00:55 up 47 min,  1 user,  load average: 0,00, 0,04, 0,07
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU WHAT
ubuntu    :0                :0                22:13   ?xdm?  4:40   0.00s /usr/libexec/gd
```

- ✓ w: Hangi kullanıcının o anda hangi uygulamayı/komutu çalıştırdığı bilgisi.
- ✓ who: Sistemde hangi kimlikle çalışıldığı.
- ✓ whoami: Kullanıcının hangi kimlikle çalıştığını gösterir.

```
ubuntu@ubuntu-VirtualBox:/$ who
ubuntu    :0                2021-04-04 22:13 (:0)
ubuntu@ubuntu-VirtualBox:/$ whoami
ubuntu
```

```
ubuntu@ubuntu-VirtualBox:/$ sudo -i
root@ubuntu-VirtualBox:~# whoami
root
root@ubuntu-VirtualBox:~# who
ubuntu    :0                2022-03-16 10:43 (:0)
root@ubuntu-VirtualBox:~# whoami
root
root@ubuntu-VirtualBox:~#
```

whereis

✓ Bir dosyayla ilgili çalıştırılabilir dosyanın, kaynak ve yardım dosyalarının konumunu öğrenmek için **whereis** komutu kullanılır.

- **Parametresiz kullanıldığında;** hakkında bilgi almak istediğimiz dosyanın çalıştırılabilir dosyasının ve yardım sayfasının bulunduğu yeri çıktı olarak verir.
- **-b parametresiyle komut verirse;** sadece çalıştırılabilir dosyanın yerini,
- **-m ile kullanıldığında ;** sadece man sayfasının yerini,
- **-s ile komut verilirse;** varsa kaynak kodunun konum bilgisini verir.

```
ubuntu@ubuntu-VirtualBox:/$ whereis ls
ls: /usr/bin/ls /usr/share/man/man1/ls.1.gz
ubuntu@ubuntu-VirtualBox:/$ whereis -b ls
ls: /usr/bin/ls
ubuntu@ubuntu-VirtualBox:/$ whereis -m ls
ls: /usr/share/man/man1/ls.1.gz
ubuntu@ubuntu-VirtualBox:/$ whereis -s ls
ls:
```

which

✓ Parametre olarak verilen bir komutun tam yol bilgisini öğrenmek için kullanılan komut **which** komutudur.

```
ubuntu@ubuntu-VirtualBox:/$ which cat
/usr/bin/cat
ubuntu@ubuntu-VirtualBox:/$
```

dmidecode

- ✓ Sistemin donanım bileşenleri ve BIOS ile ilgili bilgiyi görmek için dmidecode komutunu kullanabiliriz.
- ✓ Bu bilgiler, sistem DMI (Desktop Management Interface) tablosunun okunmasıyla elde edilir. Komutun genel kullanım kalıbı şöyledir:

dmidecode -t <donanım>

- ✓ Bu şekilde -t parametresiyle aşağıdaki bilgiler öğrenilebilir:

➤ system

➤ memory

➤ baseboard

➤ cache

➤ chassis

➤ connector

➤ processor

➤ slot

dmidecode

- ✓Örneğin **dmidecode -t system** komutu, aşağıdakine benzer bir çıktı döner.

```
root@ubuntu-VirtualBox:~# dmidecode -t system
# dmidecode 3.2
Getting SMBIOS data from sysfs.
SMBIOS 2.5 present.

Handle 0x0001, DMI type 1, 27 bytes
System Information
    Manufacturer: innotek GmbH
    Product Name: VirtualBox
    Version: 1.2
    Serial Number: 0
    UUID: c1400118-6ddf-0a4b-9c9d-f461d74c5705
    Wake-up Type: Power Switch
    SKU Number: Not Specified
    Family: Virtual Machine

root@ubuntu-VirtualBox:~#
```

- ✓Eğer BIOS bilgisini görmek isterseniz **dmidecode -t 0** komutunu kullanmalısınız.

```
root@ubuntu-VirtualBox:~# dmidecode -t 0
# dmidecode 3.2
Getting SMBIOS data from sysfs.
SMBIOS 2.5 present.

Handle 0x0000, DMI type 0, 20 bytes
BIOS Information
    Vendor: innotek GmbH
    Version: VirtualBox
    Release Date: 12/01/2006
    Address: 0xE0000
    Runtime Size: 128 kB
    ROM Size: 128 kB
    Characteristics:
        ISA is supported
        PCI is supported
        Boot from CD is supported
        Selectable boot is supported
        8042 keyboard services are supported (int 9h)
        CGA/mono video services are supported (int 10h)
        ACPI is supported
```


fdisk-l

- ✓ Aslında diskleri bölümllemek için kullanılan **fdisk** komutu **-l** parametresiyle **fdisk -l** şeklinde kullanılarak sistemdeki disk bölümleri görülür.

```
Disk /dev/sda: 10 GiB, 10737418240 bytes, 20971520 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 2375B0DF-C001-4C6C-B16A-B3ADD6E6AE0E

Device            Start      End    Sectors  Size Type
/dev/sda1         2048      4095     2048    1M BIOS boot
/dev/sda2         4096  1054719  1050624  513M EFI System
/dev/sda3      1054720 20969471 19914752  9,5G Linux filesystem
```

df

- ✓ Disk kullanımı ile ilgili ayrıntılı bilgiyi **df** komutu yardımıyla öğrenebiliriz.

```
root@ubuntu-VirtualBox:~# df
Filesystem      1K-blocks    Used Available Use% Mounted on
tmpfs            203160      1380   201780    1% /run
/dev/sda3       9735476 8470412   750812   92% /
tmpfs           1015788        0  1015788    0% /dev/shm
tmpfs            5120         4     5116    1% /run/lock
tmpfs            4096        0     4096    0% /sys/fs/cgroup
/dev/sda2       524272     7988   516284    2% /boot/efi
tmpfs           203156     116   203040    1% /run/user/1000
root@ubuntu-VirtualBox:~#
```

- ✓ Daha düzgün bir çıktı (MB, GB bazında) almak için **df -h** komutu kullanılır.

```
root@ubuntu-VirtualBox:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
tmpfs            199M  1,4M  198M   1% /run
/dev/sda3        9,3G  8,1G  733M  92% /
tmpfs            992M    0  992M   0% /dev/shm
tmpfs            5,0M  4,0K  5,0M   1% /run/lock
tmpfs            4,0M    0  4,0M   0% /sys/fs/cgroup
/dev/sda2        512M  7,9M  505M   2% /boot/efi
tmpfs            199M  116K  199M   1% /run/user/1000
root@ubuntu-VirtualBox:~#
```

- ✓ Dosya sistemi bilgisiyle birlikte liste almak için de **df -T** komutu kullanılmalıdır. (ext4, vfat vs.)

```
root@ubuntu-VirtualBox:~# df -T
Filesystem      Type 1K-blocks    Used Available Use% Mounted on
tmpfs           tmpfs  203160      1380   201780    1% /run
/dev/sda3       ext4  9735476 8480796   740428   92% /
tmpfs           tmpfs  1015788        0  1015788    0% /dev/shm
tmpfs           tmpfs   5120         4     5116    1% /run/lock
tmpfs           tmpfs   4096        0     4096    0% /sys/fs/cgroup
/dev/sda2       vfat  524272     7988   516284    2% /boot/efi
tmpfs           tmpfs  203156     116   203040    1% /run/user/1000
root@ubuntu-VirtualBox:~#
```


du

- ✓ Bir dizinin altında bulunan alt dizin, dosya ya da klasörlerin diskte kapladıkları alanın boyutunu öğrenmek için **du** komutundan faydalanılır.
- ✓ Komut çıktısının istediğimiz sonucu vermesi için çeşitli parametrelerle birlikte kullanılabilir.

```
root@ubuntu-VirtualBox:~# du -h /etc/apt
4,0K    /etc/apt/auth.conf.d
12K     /etc/apt/trusted.gpg.d
84K     /etc/apt/apt.conf.d
4,0K    /etc/apt/sources.list.d
4,0K    /etc/apt/preferences.d
128K    /etc/apt
root@ubuntu-VirtualBox:~#
```

free

- ✓ Bu komutla kullanılan bellek miktarı görülebilir.
- ✓ -m parametresiyle **free -m** şeklinde kullanılırsa, çıktı daha düzenli (MB) olacaktır.

```
root@ubuntu-VirtualBox:~# free -m
              total        used        free      shared  buff/cache   available
Mem:           1983         839          74         16        1070        971
Swap:           448           26        421
root@ubuntu-VirtualBox:~#
```

modinfo

- ✓ Linux kernel modülleriyle ilgili bilgileri listelemek için **modinfo** komutu kullanılır.
- ✓ Örneğin bluetooth ile ilgili çekirdek modülü bilgisi için **modinfo bluetooth** komutunu kullanmak gerekir.

```
root@ubuntu-VirtualBox:~# modinfo bluetooth
filename:      /lib/modules/5.8.0-48-generic/kernel/net/bluetooth/bluetooth.ko
alias:         net-pf-31
license:       GPL
version:       2.22
description:   Bluetooth Core ver 2.22
author:        Marcel Holtmann <marcel@holtmann.org>
srcversion:    41F055D5DFB55E828E7128D
depends:        ecdh_generic
retpoline:     Y
intree:        Y
name:          bluetooth
vermagic:      5.8.0-48-generic SMP mod_unload modversions
signature:     PKCS#7
```

- ✓ Modül listesini görmek isterseniz **lsmod** komutunu kullanabilirsiniz

BÖLÜM 4

Dizinler/Klasörler Arasında Gezme ve İçeriğini Listeleme

Dizinler/Klasörler Arasında Gezme ve İçeriğini Listeleme

- ✓ Linux dünyasında her şey (programlar, dosyalar, aygıtlar) bir dizin altında alt dizin/dosya görünümünde yer alır.
- ✓ Bütün dizinler, hiyerarşik şekilde kök dizine (root dizin) bağlıdır. Root dizin / (slash) ile ifade edilir.
- ✓ Linux sistemlerde root kullanıcı da dahil her kullanıcının kendi ev (home) dizini vardır.
- ✓ Konsol/terminal, varsayılan olarak kullanıcının kendi ev dizininde çalışmaya başlar.
- ✓ Root kullanıcının ev dizini / root dizinidir.
- ✓ Sisteme yeni kullanıcı eklendiği zaman, bu kullanıcıların da ev dizinleri (kişisel dizinleri) /home dizini altında /home/user1, /home/user2 şeklinde yer alacaktır.
- ✓ Dizinler arasında gezmek, içeriklerini listelemek, hangi dizinde çalıştığımızı görmek için kullandığımız komutlar vardır.

cd

- ✓ cd (change directory) komutu ile dizinler arasında geçiş yapabiliyoruz.
- ✓ Örneğin cd /usr/share komutu ile /share dizinine geçebiliriz.
- ✓ Herhangi bir dizindeyken ana dizinimize dönmek için cd komutunu kullanmanız yeterlidir.

```
ubuntu@ubuntu-VirtualBox:~$ cd /usr/share
ubuntu@ubuntu-VirtualBox:/usr/share$ cd
ubuntu@ubuntu-VirtualBox:~$
```

- ✓ Bir alt dizinden üst dizine geçmek istediğimizde ise cd .. (ya da cd ../) komutunu kullanabiliriz.
- ✓ Eğer iki üst dizine geçmek istersek cd ../../, üç dizin üste geçmek için de cd ../../../ komutlarını kullanmamız gerekir.

Pushd-popd

- ✓ İki ayrı klasör/dizin açıp, bu dizinler arasında istediğimiz gibi geçiş yapmamıza yarayan komutlar pushd ve popd komutlarıdır.
- ✓ Bu işlemi bir nevi dizin öteleme-çekme şeklinde yapıyor diyebiliriz.

```
ubuntu@ubuntu-VirtualBox:~$ cd /home
ubuntu@ubuntu-VirtualBox:/home$ pwd
/home
ubuntu@ubuntu-VirtualBox:/home$ pushd /etc
/etc /home
ubuntu@ubuntu-VirtualBox:/etc$ pwd
/etc
ubuntu@ubuntu-VirtualBox:/etc$ popd
/home
ubuntu@ubuntu-VirtualBox:/home$
```

```
ubuntu@ubuntu-VirtualBox:/home$ pushd /etc
/etc /home
ubuntu@ubuntu-VirtualBox:/etc$ pushd
/home /etc
ubuntu@ubuntu-VirtualBox:/home$ pushd
/etc /home
ubuntu@ubuntu-VirtualBox:/etc$
```

Is Komutu

- ✓ Dizinler altında bulunan alt dizin, klasör ve dosyaları listelemek için ls komutunu kullanırız.
- ✓ Komutu parametresiz kullanabileceğimiz gibi birçok parametreyle beraber de kullanabiliriz.
- ✓ Parametresiz kullanım aşağıdaki gibi bir çıktı verir.

```
ubuntu@ubuntu-VirtualBox:~$ ls
Belgeler  Dosya2  Genel  İndirilenler  Resimler  test
deneme    facebook.com  gokce  linuxcmd     snap      testB
deneme2   File_A   gokce.txt  Masaüstü     Şablonlar  Videolar
```

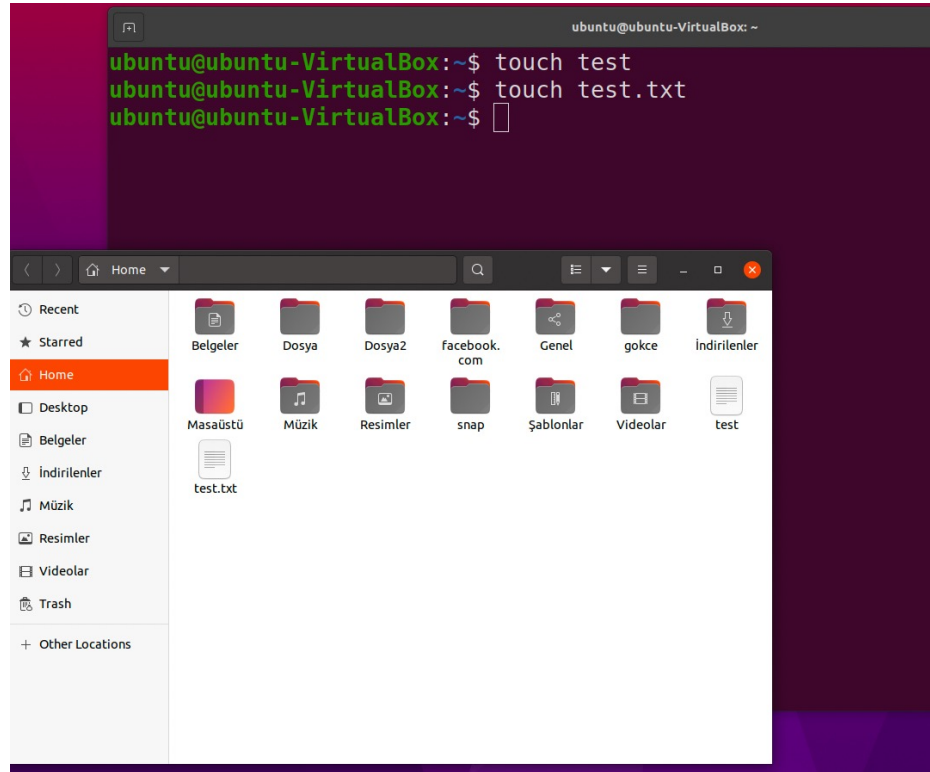
- ✓ Eğer ls -l şeklinde komut verirse, daha ayrıntılı bir çıktı alırız.
- ✓ Bu ayrıntılı listede, dosyaların erişim izinleri, sahibi, oluşturulma tarihi gibi bilgiler yer alır.

```
ubuntu@ubuntu-VirtualBox:~$ ls -l
total 76
drwxr-xr-x  4 ubuntu ubuntu 4096 Nis  15 10:17 Belgeler
-rwx----- 1 ubuntu ubuntu   0 Mar 26 12:53 deneme
-rw-rw-r--  1 ubuntu ubuntu   0 Nis  15 10:21 deneme2
drwxrwxr-x  2 ubuntu ubuntu 4096 Nis  15 10:22 Dosya
drwxrwxr-x  2 ubuntu ubuntu 4096 Nis  15 10:18 Dosya2
drwxrwxr-x  2 ubuntu ubuntu 4096 Mar 30 12:15 facebook.com
-rw-rw-r--  1 ubuntu ubuntu   25 Mar 29 22:36 File_A
-rw-rw-r--  1 ubuntu ubuntu   17 Mar 29 22:20 File_B
drwxr-xr-x  2 ubuntu ubuntu 4096 Şub  25 11:28 Genel
drwxrwxr-x  2 ubuntu ubuntu 4096 Nis  15 09:59 gokce
-rw-rw-r--  1 ubuntu ubuntu   0 Nis  15 09:59 gokce.txt
-rw-rw-r--  1 ubuntu ubuntu   29 Nis  15 10:07 important
drwxr-xr-x  2 ubuntu ubuntu 4096 Mar 29 15:16 İndirilenler
```

BÖLÜM 5

DOSYA İŞLEMLERİ

touch



```
ubuntu@ubuntu-VirtualBox:~$ ls -l test
-rw-rw-r-- 1 ubuntu ubuntu 0 Nis 15 11:24 test
ubuntu@ubuntu-VirtualBox:~$ touch test
ubuntu@ubuntu-VirtualBox:~$ ls -l test
-rw-rw-r-- 1 ubuntu ubuntu 0 Nis 15 11:26 test
ubuntu@ubuntu-VirtualBox:~$
```

✓ touch test komutuyla test isimli metin dosyamızı oluşturabiliriz, (isterseniz test.txt adı da verilebilir).

✓ Bu komutu sadece metin dosyası oluşturmak için değil, aynı zamanda var olan bir dosyanın tarih etiketini güncellemek için de kullanabilirsiniz.

cat

- ✓ Dosya içeriğinin terminal ekranından okunmasında kullanılan cat komutu, bunun haricinde bir metin dosyasına yazı yazmak için de kullanılır.
- ✓ Mesela boş olan test dosyasının içine yazı yazmak için `cat > test` komutunu verelim (> operatörünün kullanıldığına dikkat edin) ve istediğimiz ifadeleri yazarak `ctrl+D` ile yazımızı sonlandıralım.
- ✓ Böylece, konsoldan yazdığımız şeyler dosyamızın içine kaydolacaktır.
- ✓ Dosya içeriğini okumak için de `cat test` komutunu kullanabiliriz.

```
ubuntu@ubuntu-VirtualBox:~/Belgeler$ cat > test
deneme
deneme2
ubuntu@ubuntu-VirtualBox:~/Belgeler$ cat test
deneme
deneme2
ubuntu@ubuntu-VirtualBox:~/Belgeler$
```

cat

- ✓ > operatörü dosyalara yönlendirme yapılacağı zaman kullanılıyor. Bu operatörü birçok farklı şekilde kullanılabilir.
- ✓ Verilen ifadeleri, komut çıktılarını vs. bir dosyaya yönlendirme yapabildiği gibi, bir dosyanın içeriğini argüman olarak da alabilir.
- ✓ Var olan bir text dosyası içine herhangi bir şey yazmak için, dosyanın boş olması gerekir.
- ✓ Eğer dosya boş değilse eski içerik silinecek ve yeni içerik yazılacaktır. Tabi bu durum > operatörünü kullanırken böyledir.
- ✓ Eğer » operatörünü kullanırsak, dosya içeriği silinmeden, bizim girdiğimiz ifade dosyaya en son satırdan itibaren eklenecektir.

cat

```
ubuntu@ubuntu-VirtualBox:~/Belgeler$ cat > test
deneme
deneme2
ubuntu@ubuntu-VirtualBox:~/Belgeler$ cat test
deneme
deneme2
ubuntu@ubuntu-VirtualBox:~/Belgeler$ cat >> test
Ali
Veli
ubuntu@ubuntu-VirtualBox:~/Belgeler$ cat test
deneme
deneme2
Ali
Veli
ubuntu@ubuntu-VirtualBox:~/Belgeler$ cat > test
1
2
3
ubuntu@ubuntu-VirtualBox:~/Belgeler$ cat test
1
2
3
ubuntu@ubuntu-VirtualBox:~/Belgeler$
```

- ✓ touch test ile yeni bir metin dosyası oluşturduk.
- ✓ cat > test komutunu verdik ve istediğimiz ifadeleri yazdıktan sonra ctrl+D ile sonlandırdık.
- ✓ cat test ile dosya içeriğine baktık.
- ✓ Cat >> test ile dosyamıza yeni ifadeler yazdık ve ctrl+D ile sonlandırdık.
- ✓ cat test ile yeni yazdığımız ifadelerin dosya içeriğine eklendiğini gördük.
- ✓ cat > test (tek > kullanıldığına dikkat ediniz) ile dosyamıza yeni içerik eklemek istedik.
- ✓ cat test ile dosyamızın içeriğini tekrar kontrol ettiğimizde, eski içeriğin silindiğini ve en son girdiğimiz içeriğin yazıldığını gördük

echo

- ✓ Normalde bu komut ile istenilen ifadeler terminal ekranına çıktı olarak gönderilir.
- ✓ Bunun yanında bir dosya oluşturup içine istenilen string'in/ifadenin yazılması için de kullanılabilir.

```
3
ubuntu@ubuntu-VirtualBox:~/Belgeler$ echo "linux" > linuxcmd
ubuntu@ubuntu-VirtualBox:~/Belgeler$ ls
linuxcmd test test2
ubuntu@ubuntu-VirtualBox:~/Belgeler$ cat linuxcmd
linux
ubuntu@ubuntu-VirtualBox:~/Belgeler$
```

- ✓ Is komutu ile içinde bulunduğumuz dizinin içeriğini görürüz.
- ✓ Eğer echo komutunu echo * şeklinde kullanırsak ls komutu ile aynı çıktıyı alırız.

```
ubuntu@ubuntu-VirtualBox:~/Belgeler$ ls
linuxcmd test test2
ubuntu@ubuntu-VirtualBox:~/Belgeler$ echo *
linuxcmd test test2
ubuntu@ubuntu-VirtualBox:~/Belgeler$
```

paste

- ✓ İki ya da daha fazla dosyada bulunan karşılıklı satırları aralarına tab boşluğu koyarak yan yana listelemek için paste komutu kullanılır.

```
ubuntu@ubuntu-VirtualBox:~/Belgeler$ paste test test2
1      a
2      b
3      c
       d
ubuntu@ubuntu-VirtualBox:~/Belgeler$
```

- ✓ Bu iki dosyanın içeriğini birleştirilip yeni bir dosyaya yönlendirilebilir.

```
ubuntu@ubuntu-VirtualBox:~/Belgeler$ paste test test2 > test3
ubuntu@ubuntu-VirtualBox:~/Belgeler$ cat test3
1      a
2      b
3      c
       d
ubuntu@ubuntu-VirtualBox:~/Belgeler$
```

grep

- ✓ Dosyalar içerisinde ya da bir komut çıktısı içinde bir ifade (string) aramak için grep komutunu kullanırız.
- -v: Komutun davranışını tersine çevirir, yani aranılan kelimeyi içermeyen satırlar listelenir.
- -i: Arama sırasında büyük/küçük harf ayrımı yapılmaz.
- -r: Verilen dizinin alt dizinlerinde de arama yapar.
- -n: Aranılan kelimenin geçtiği satır numarasını da gösterir.
- -c: Belirtilen dizinde aradığınız kelimenin kaç kere geçtiğini gösterir.
- -l: Şablona uygun satırların bulunduğu dosya adlarını listeler.

```
ubuntu@ubuntu-VirtualBox:~$ grep ubuntu /etc/passwd
ubuntu:x:1000:1000:Ubuntu,,,:/home/ubuntu:/bin/bash
ubuntu@ubuntu-VirtualBox:~$ grep -c ubuntu /etc/passwd
1
ubuntu@ubuntu-VirtualBox:~$
```


find

- ✓ Zaman zaman adının yalnızca bir kısmını bildiğiniz ama bulunduğu dizini bir türlü hatırlayamadığınız dosyalar ve dizinler olacaktır.
- ✓ Bu dosyaları bulmak ve bunlarla ilgili çeşitli işlemler yapmak için find komutu kullanılır.
- ✓ Kullanım kalıbı **find [dosya_yolu] [aranacak_ifade]** şeklindedir

```
ubuntu@ubuntu-VirtualBox:~$ find /etc -name *.conf
/etc/gtk-3.0/im-multipress.conf
/etc/modules-load.d/modules.conf
/etc/modules-load.d/cups-filters.conf
/etc/pam.conf
/etc/gdm3/custom.conf
/etc/pnm2ppa.conf
/etc/adduser.conf
/etc/speech-dispatcher/speechd.conf
/etc/speech-dispatcher/modules/espeak-ng-mbrola-generic.conf
/etc/speech-dispatcher/modules/llia_phon-generic.conf
/etc/speech-dispatcher/modules/dtk-generic.conf
/etc/speech-dispatcher/modules/ivona.conf
/etc/speech-dispatcher/modules/ivona.conf
```

exec

- ✓ Bulduğumuz dosyalarla ilgili çeşitli işlemler yapabilmek için find komutuyla birlikte -exec kullanılır.
- ✓ Mesela ilk önce `find /home/... -name *.jpg` komutuyla depo klasöründeki jpg dosyalarını bulalım.

```
gokceoguzerenler@gokceoguzerenler-VirtualBox:~$ find /home/gokceoguzerenler/Pictures -name *.jpg  
/home/gokceoguzerenler/Pictures/doga.jpg
```

- ✓ find komutunu -exec ile beraber kullanalım ve bulduğu jpg dosyalarını silmesini isteyelim.
- ✓ Bunun için `find /home/... -name *.jpg -exec rm {} \;` komutunu kullanmalıyız.

```
gokceoguzerenler@gokceoguzerenler-VirtualBox:~$ find /home/gokceoguzerenler/Pictures -name *.jpg -exec rm {} \;  
gokceoguzerenler@gokceoguzerenler-VirtualBox:~$ find /home/gokceoguzerenler/Pictures -name *.jpg  
gokceoguzerenler@gokceoguzerenler-VirtualBox:~$
```

KAYNAKLAR

- Linux Komut Satırı
 - Kemal DEMİREZ

