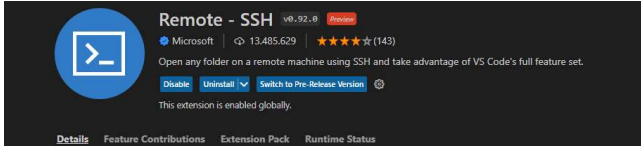


Linux Plus 1

30 Kasım 2022 Çarşamba 19:08

VS Code da çalışmak görsel açıdan daha rahat olacağı için VS Code kullanacağız. Öncelikle Extensionlardan SSH-Remote eklentisini indireceğiz.



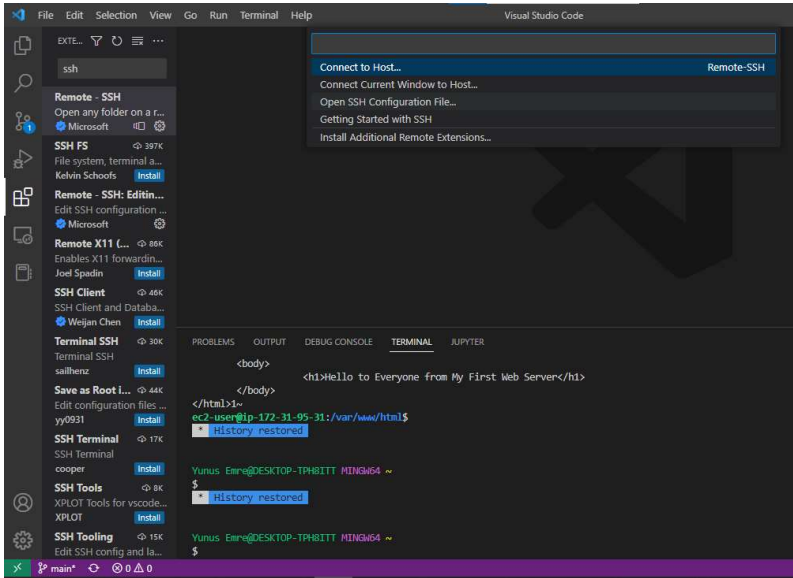
SSH eklentisinde, instance ın connect kısmındaki ssh linkini dosya şeklinde açıyoruz. Daha sonra EC2 da ayağa kaldırdığımız ubuntu instance ın SSH client linkini kopyalıyoruz:

ssh -i "first-key.pem" ubuntu@ec2-18-212-164-217.compute-1.amazonaws.com

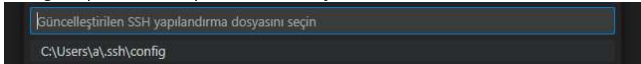
Bu linkte; ssh komutu, sonrasında pem.key imizin nerede olduğu ve hangi kullanıcı adı ile işlem yaptığımız ve hangi hosta bağlanacağımızın verileri var.

Yüklediğimiz extension ile bu verileri bir dosya şeklinde sunacağız, bu sayede her defasında pem.key yazmadan dosyadan bağlanacağımız yeri seçip bağlanabiliyoruz.

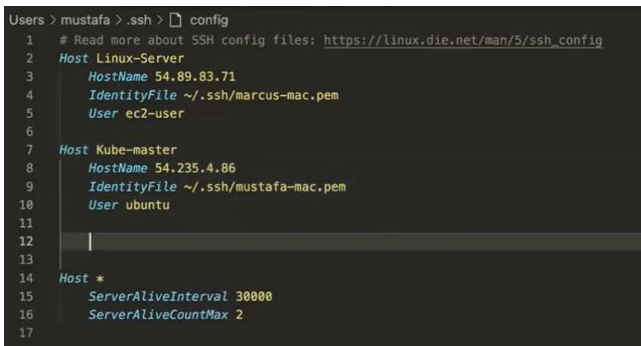
Sonra VS Code sol alt köşede çıkan yeşil butona basıyoruz ve Open SSH Configuration File.. a tıklıyoruz. Bağlantı bilgilerini bu dosyanın içine yazacağız.



config dosyamız neredeyse onun adresi çıkacaktır:



Gelen ekranda önceden bağlandığımız serverlar olacaktır, bunları silebiliriz.



Biz buraya bağlanmak istediğimiz server için kod bloğunu oluşturacağız.



```
ServerAliveInterval 30000
ServerAliveCountMax 2
```

en alttaki yıldızlı kısım yoksa onu da ekleyecek, varsa bir daha eklemeye gerek yok.
Host a Linux-Plus diye bir isim verdik (fark etmez ne olduđu)
HostName kısmını üstte belirttiğim ssh linkinden alıyoruz:

ec2-18-212-164-217.compute-1.amazonaws.com

IdentityFile karşısına ise pem.key dosyamız neredeyse onun adresini yazıyoruz.

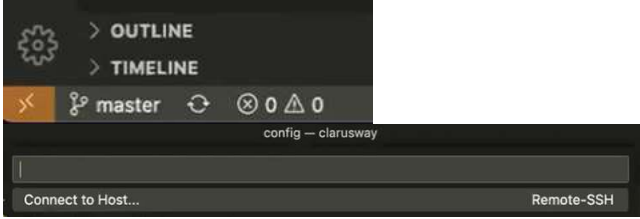
Windows kullananlar için buradaki adresin yazım formatı farklı (VS Code da alttaki terminal kısmında Powershell açıp .ssh klasörüne gidip (.pem oradaysa) adresi oradan alabilirsiniz)
Format şu şekilde olacak:

```
IdentityFile C:\Users\burak\.ssh\first-key.pem
```

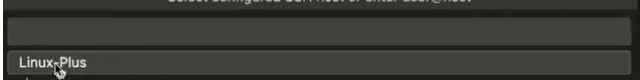
User karşısına ise ssh linkindeki kullanıcı adını yazıyoruz.

Config dosyasının içinde üstteki kısımları yazarken Linux un case sensitive olduğunu unutmayın! Her şeyin tamamıyla doğru yazıldığından emin olun. Küçük hatalardan ötürü bağlantı gerçekleşmiyor.

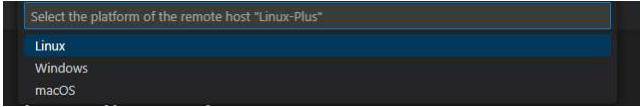
Daha sonra dosyayı save ediyoruz ve sonra sol alttaki butona basarak gelen menüden Connect to Host diyoruz:



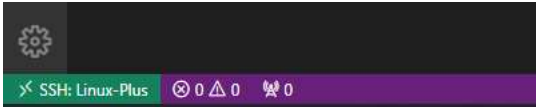
Kaydettiğimiz ismiyle Linux-plus şeklinde çıktı. Ona tıklıyoruz:



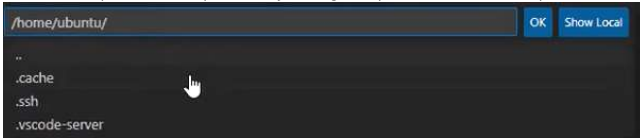
Yeni açılan pencerede Linux u seçiyoruz.



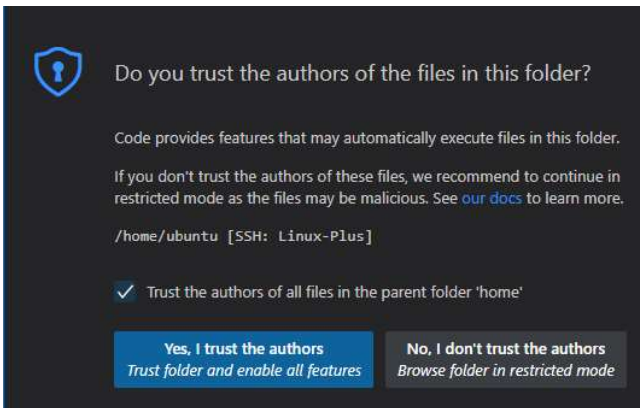
Sol altta SSH: Linux-Plus yazısını gördük ve EC2 ile bağlantı kurduğumuzu anladık



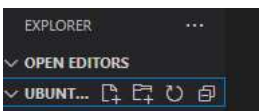
Sol tarafta open folder diyerek karşımıza gelen pencerede OK a tıklıyoruz:



Aşağıdaki ekranda Yes e tıklıyoruz:



Artık ubuntu'nun içindeyiz:



```

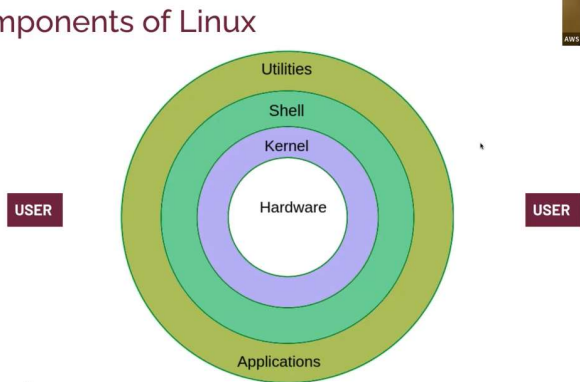
> .cache
> .ssh
> .vscode-server
$ .bash_logout
$ .bashrc
$ .profile
≡ .wget-hsts

```

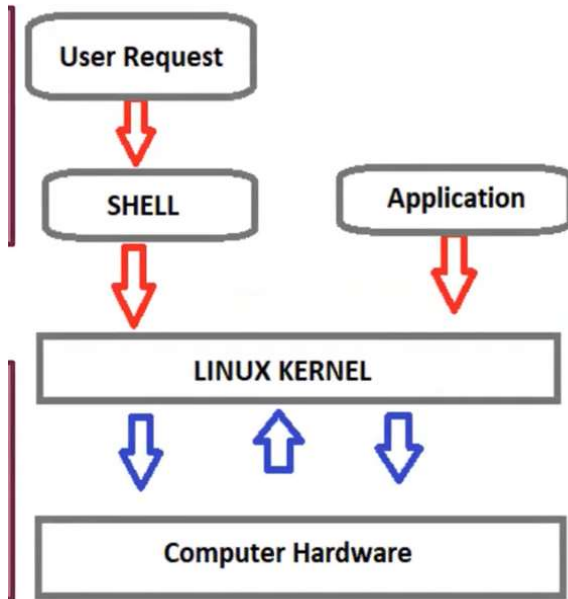
Linux plus derslerine bağlantıyı kurmuş şekilde gireceğiz.

Önceki derslerin kısa bir tekrarını yapıyoruz:

Components of Linux

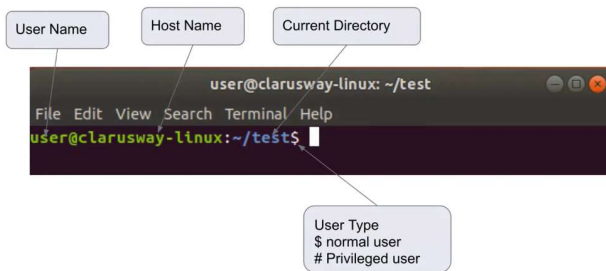


Linux bu şekilde tasarlanmış bir işletim sistemi. Kernel Shell ile Hardware arasında veri transferi yapıyor demıştik. Biz inputları application ile yaparken artık Shell katmanı ile komuta etmeyi öğreniyoruz.



Shell bir programlama dili içeriyor. Python gibi script yazacağız. Linux script deli ya da Bash script.

Prompt:



Temel bash komutları:

Basic Shell Commands

COMMAND	DESCRIPTION
---------	-------------

pwd	show current path
ls	lists directory contents
cd	change (current) directory
mkdir	create a new directory
rmdir	delete an empty directory
touch	create a file
rm	delete a file

COMMAND	DESCRIPTION
cp	copy a file to another location
mv	move a file to another location
cat	show file contents
echo	print message to screen
clear	clear the terminal screen

Bu komutları ezberlemeyeceğiz demiştik, help ten faydalanacağız veya google da arama yapacağız.

- Man Pages
- Info Pages
- whatis command
- apropos command
- --help option

IT de intihal güzel kabul edilir. Başkasının çalışmasını alıp kendi eklemeleriniz yapıyor ve kendi çalışmanızı oluşturuyorsunuz.

man komutunu kullanırken :

man <command>

man ve aradığımız komut neyse onu yazıyorduk.

info kumutu için de aynı şey geçerli:

info <command>

info komutu daha çok işin hikaye ve detay kısmını gösterir. Özellikle yeni kod oluşturmak isteyenler için.

whatis komutu man komutunu girdiğimizde birinci satırda çıkan açıklamayı bize gösteren bir komuttur:

```
clarusway@DESKTOP-UN6T2ES:~$ whatis ls
ls (1)                - list directory contents
clarusway@DESKTOP-UN6T2ES:~$ whatis pwd
pwd (1)               - print name of current/working directory
clarusway@DESKTOP-UN6T2ES:~$ whatis mv
mv (1)                - move (rename) files
clarusway@DESKTOP-UN6T2ES:~$
```

Bir komutun tam ismini hatırlamıyorsanız, kenarından köşesinden bir kısmını hatırlıyorsanız apropos komutunu kullanırsınız:

```
clarusway@DESKTOP-UN6T2ES:~$ apropos pwd
pwd (1)                - print name of current/working directory
pwdx (1)               - report current working directory of a process
unix_chkpwd (8)        - Helper binary that verifies the password of the current user
clarusway@DESKTOP-UN6T2ES:~$
```

```
clear ubuntu@desktop:~$
```

--help komutu ilk yardımcımız olur hep.

which komutu ile komutun nerede tutulduğunu görüyoruz:

```
ubuntu@ip-172-31-94-96:~$ which ls
/usr/bin/ls
```

bin klasörü; binaries, komutların yani koşulabilir dosyaların tutulduğu kısımdır. ls /usr/bin komutuyla listelediğimizde bin içeriğini görüyoruz:

```
ubuntu@ip-172-31-86-37:~$ ls /usr/bin
NF          egrep      lsipc      resizepart  systemd-cat
VAuthService eject      lslocks    resolvectl  systemd-cgls
[''         elfedit    lslogins   rev          systemd-cgtop
aa-enabled  enc2xs     lsnew      rgrep        systemd-cryptenroll
aa-exec     encguess   lsmod      rlogin       systemd-delta
aa-features-abi env         lsns       rm           systemd-detect-virt
acpi_listen ensubst    lsof       rmdir        systemd-escape
add-apt-repository eqn         lspci      rm nano      systemd-hwdb
addpart     ex         lspgpot    routeF       systemd-id128
```

Linux komutları \$PATH variables a bakarak bulur. Bir sonraki derste \$PATH ve diğer variables lara bakcaz.

```
ubuntu@ip-172-31-94-96:~$ echo $PATH
/home/ubuntu/.vscode-server/bin/6261075646f05b99068d3688932416f2346dd3b/bin:remote-cli:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
ubuntu@ip-172-31-94-96:~$
```

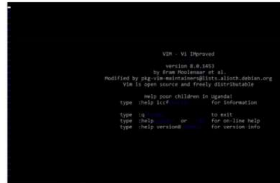
Text editörlerimiz Vim ve Nano ydu.

Vim editöründe command mode ve insert mode vardı:

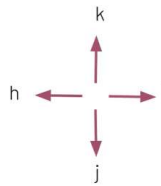
Vi/Vim Editor

- Vim is a powerful text editor used in CLI (command line interface).
- Vim is an editor to create or edit a text file.

- When you start Vim, you are **placed in Command mode**. In this mode, you can move across the screen, delete text and copy text.
- You cannot write text in command mode. To write text into a file, there is a dedicated insert mode. When you want to write something on a file, you must enter the insert mode.



Vim Command	Description
i	Enter insert mode
Esc	Enter command mode
x or Del	Delete a character
X	Delete character is backspace mode
u	Undo changes
Ctrl + r	Redo changes
yy	Copy a line
dd	Delete a line
p	Paste the content of the buffer
o	insert a blank line under the current cursor position.
:%s/foo/bar/g	Search and replace all occurrences
Esc + :w	Save changes
Esc + :wq or Esc + ZZ	Save and quit Vim



Nano editörde açar açmaz yazı yazabiliyorduk, command mode yoktu:

Nano Editor

GNU nano is a small and friendly text editor.

Besides basic text editing, nano offers features like:

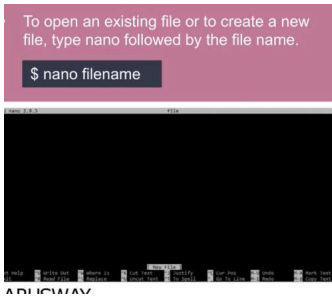
- undo/redo
- syntax coloring
- interactive search-and-replace
- auto-indentation
- line numbers
- word completion



Nano Editor

Unlike vi, nano is a **modeless editor**, which means that you can start typing and editing the text immediately after opening the file.

Nano Command	Meaning
Ctrl G	Get Help
Ctrl X	Exit



Ctrl O	Write Out
Ctrl R	Read File
Ctrl W	Where Is
Ctrl \	Replace
Ctrl K	Cut Text
Ctrl U	Uncut Text
Ctrl J	Justify
Ctrl T	To Spell
Ctrl C	Cur Pos
Alt U	Undo
Alt E	Redo

İş hayatında vim ve nano dan ziyade vs code kullanılıyor.

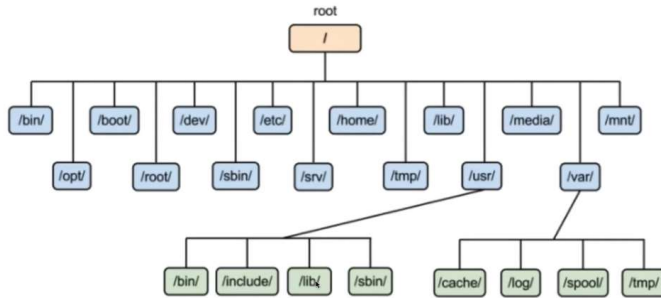
Files, Dosyalar:

Linux'te her şey dosyadan oluşur, komutlar, bağladığımız cihazlar etc.

Linux sistemi için dosya veya klasör arasındaki fark, klasör diğer dosyaların ismini tutan bir başka dosyadır.

Dosya sisteminin başında / işaretiyle simgelenen root klasörü vardır.

ROOT Directory (/)



Dosyaların simge ve renkleri:

Symbol	Meaning
-	Regular file
d	Directory
l	Link
c	Character Device File
s	Socket File
p	Named Pipe
b	Block Device

```
-rw-rw-r-- Regular File
drwxr-xr-x Directory File
lrwxrwxrwx Link File
crw-rw-r-- Character Device File
brw-rw-r-- Block Special File
srw-rw-rw- Socket File
prw-rw-r-- Named Pipe File
```

On most Linux versions ls is aliased to color-ls by default. This feature allows to see the file type without using any options to ls.

Color	Meaning
Blue	directories
Red	compressed archives
White	text files
Pink	images
Cyan	links
Yellow	Devices
Green	Executables
flashing red	broken links

```
clarusway@DESKTOP-UN6TZES:~$ ls
archive.tar  images.jpg  linux.txt  linux.txt
clarusway@DESKTOP-UN6TZES:~$
```

flashing red - broken links; linklediğimiz dosya silindiye kırmızı yanıp söner.

head komutu dosyanın ilk 10 satırını default olarak ya da -sayı belirtirsek baştan o sayı kadar satır gösterir.

head output the first ten lines of a file.

head -n output the first n lines of a file.

tail komutu ise head ile aynı şekilde fakat sondan satırları gösterir.

tail output the last ten lines of a file.

tail -n

output the last n lines of a file.

cat komutu istediğimiz dosyanın içeriğini bize gösterir:

cat

Display a file on the screen.

öte yandan farklı dosyaların içeriğini birleştirip bir başka dosyaya da yazdırabilir.

```
clarusway@DESKTOP-UN6T2ES:~$ cat file1 file2 file3
this is file1
this is file2
this is file3
clarusway@DESKTOP-UN6T2ES:~$ cat file1 file2 file3 > all
clarusway@DESKTOP-UN6T2ES:~$ cat all
this is file1
this is file2
this is file3
```

cat > dosyaismi.txt komutu ile sol tarafta bize prompt açar ve istediğimizi yazarak o dosyayı oluşturabiliriz. ctrl+d ile prompttan çıkabiliriz:

```
clarusway@DESKTOP-UN6T2ES:~$ cat > winter.txt
It is very cold today!
clarusway@DESKTOP-UN6T2ES:~$
```

The **Ctrl d** key combination will send an **EOF (End of File)** to the running process ending the cat command.

cat >> dosyaismi.txt ile de aynı yöntemle satır ekleyebiliriz.

more komutuyla default olarak 5'er satır şeklinde dosya içeriğini görüntüleyebiliriz. more -sayı şeklinde de kullanabiliriz.

more

view (but not modify) the contents of a text file one screen at a time.

```
clarusway@DESKTOP-UN6T2ES:~$ more -2 quotes.txt
1. Cherish your visions and your dreams as they are the children of your soul, the blueprints of your ultimate achievements.
--More-- (32%)
```

less komutu ise more komutuna benzer ancak hacimli dosyalarda dosyanın sadece görüntülemek istediğimiz kadar kısmı yüklenir bu da sistemi yavaşlatmaz. more da ise dosyanın tamamını yükler ancak istediğimiz kısmını gösterir.

less

Similar to more, less command allows you to view the contents of a file and navigate through file. The **main difference** between more and less is that **less** command is **faster** because it **does not load the entire file at once**.

```
clarusway@DESKTOP-UN6T2ES:~$ less quotes.txt
1. Cherish your visions and your dreams as they are the children of your soul, the blueprints of your ultimate achievements.
2. Start by doing what's necessary; then do what's possible; and suddenly you are doing the impossible.
3. The difficult we do immediately. The impossible takes a little longer.
4. We are what we repeatedly do. Excellence, then, is not an act, but a habit.
quotes.txt (END)
```

tac komutu cat in tam tersidir. dosya içeriğini ters çevirip yazdırır, dosyayı değiştirmez:

tac

concatenate and print files in reverse.

```
clarusway@DESKTOP-UN6T2ES:~$ cat count.txt
one
two
three
four
five
clarusway@DESKTOP-UN6T2ES:~$ tac count.txt
five
four
three
two
one
clarusway@DESKTOP-UN6T2ES:~$
```

Bize verilen bir görevde örn 7000 satırlık bir dosyanın sadece belirli bir kısmıyla ilgileniyor olacağız, bu komutları ilgilendiğimiz kısmını süzmek için kullanacağız.

bir dosyayı bulmak için dosyayı isim bazında, yer bazında find komutu ile aratabiliriz:

find

search for files in a directory hierarchy.

find [starting-point...] [expression]

find

Find all the files whose name is clarusway.txt in a current working dir

```
clarusway@DESKTOP-UN6T2ES:~$ find . -name clarusway.txt
./clarusway.txt
clarusway@DESKTOP-UN6T2ES:~$
```

find . -name clarusway.txt komutunda . arayacağı yeri (bulduğumuz yer) -name de dosya ismini yazmamızı sağlar.

Linux case sensitive olduğu için o isimdeki bütün dosyaları getirmek için -iname yazmamız gerekir. i; ignore anlamındadır.

```
clarusway@DESKTOP-UN6T2ES:~$ find /home -iname clarusway.txt
/home/clarusway/Clarusway.txt
/home/clarusway/clarusway.txt
clarusway@DESKTOP-UN6T2ES:~$
```

tip bazında da arama yapabiliriz, mesela movies isimli directory leri aramak için:

```
clarusway@DESKTOP-UN6T2ES:~$ find /home -type d -name movies
/home/clarusway/movies
clarusway@DESKTOP-UN6T2ES:~$
```

isimde yer alan ifadeye göre arama yapmak için -name den sonra örneğin; "*.txt" şeklinde arama yapabiliriz:

bulduğum directoryde, dosya (f) tipinde, isminde .txt olan bütün dosyaları bulmak için:

```
clarusway@DESKTOP-UN6T2ES:~$ find . -type f -name "*.txt"
./Clarusway.txt
./clarusway.txt
./count.txt
./linux.txt
./quotes.txt
./winter.txt
clarusway@DESKTOP-UN6T2ES:~$
```

grep komutu "global regular expression print" in kısaltmasıdır. dosya içinde metin araması yaptırır.

grep

The grep, which stands for "global regular expression print," is used to search text.

grep [options] pattern [files]

Options	Description
-c	This prints only the number of lines that match a pattern
-h	Do not display the filenames headers.
-i	Ignores, case for matching
-l	Displays list of a filenames only.
-n	Display the matched lines and their line numbers.
-v	This prints out all the lines that do not matches the pattern

aşağıdaki örnekte cat komutu ile quotes.txt dosyasının içeriğini görüntülüyoruz, grep "Start" quotes.txt ile de metin içinde Start kelimesinin geçtiği satırı bize gösteriyor:

```
clarusway@DESKTOP-UN6T2ES:~$ cat quotes.txt
1. Cherish your visions and your dreams as they are the children of your soul, the blueprints of your ultimate achievements.
2. Start by doing what's necessary; then do what's possible; and suddenly you are doing the impossible.
3. The difficult we do immediately. The impossible takes a little longer.
4. We are what we repeatedly do. Excellence, then, is not an act, but a habit.
clarusway@DESKTOP-UN6T2ES:~$ grep "Start" quotes.txt
2. Start by doing what's necessary; then do what's possible; and suddenly you are doing the impossible.
clarusway@DESKTOP-UN6T2ES:~$
```

Birden fazla olsaydı bütün satırları gösterirdi.

ileride kullanacağımız toollarda grep i çok kullanacağız.

grep -n hangi satırda olduğunu da gösterirken, grep -c metinde kaç kere geçtiğini belirtir:

grep -n

Returns the result of lines matching the search string.

```
clarusway@DESKTOP-UN6T2ES:~$ grep -n "Start" quotes.txt
2: Start by doing what's necessary; then do what's possible; and suddenly you are doing the impossible.
```


grep -c

Returns the number of lines in which the results matched the search string.

```
clarusway@DESKTOP-UN6T2ES:~$ grep -c "Start" quotes.txt
1
clarusway@DESKTOP-UN6T2ES:~$
```

grep -v ise kelimenin geçmediği satırları gösterir:

grep -v

Returns the result of lines not matching the search string.

```
clarusway@DESKTOP-UN6T2ES:~$ cat quotes.txt
1. Cherish your visions and your dreams as they are the children of your soul, the blueprints of your ultimate achievements.
2. Start by doing what's necessary; then do what's possible; and suddenly you are doing the impossible.
3. The difficult we do immediately. The impossible takes a little longer.
4. We are what we repeatedly do. Excellence, then, is not an act, but a habit.
clarusway@DESKTOP-UN6T2ES:~$ grep -v "Start" quotes.txt
1. Cherish your visions and your dreams as they are the children of your soul, the blueprints of your ultimate achievements.
3. The difficult we do immediately. The impossible takes a little longer.
4. We are what we repeatedly do. Excellence, then, is not an act, but a habit.
clarusway@DESKTOP-UN6T2ES:~$
```

grep çok derin komutlardan birisi. grep --help ile inceleyebiliriz.

Hands-on a geçiyoruz.

Hands-on readme de olmayan şekilde grep komutunun kullanımı:

```
cat linux.txt | grep "Linux"
```

```
grep -w "ker" linux.txt
```

üstteki komut "ker" tek başına bir kelimeyse getir demektir.

```
history | grep "find"
```

üstteki komut da history içinde find ile yazdığımız komutları arar.