

# ლინუქსში ნავიგაციის ბრძანებები

## სარჩევი

1. ძირითადი კონცეფციები
  - 1.1 საქალალდის სტრუქტურა
  - 1.2 გზა (Path)
  - 1.3 მიმდინარე დირექტორია
2. ძირითადი ნავიგაციის ბრძანებები
  - 2.1 pwd - სად ვარ?
  - 2.2 ls - რა არის აქ?
  - 2.3 cd - საქალალდეში შესვლა
  - 2.4 tree - ვიზუალური წარმოდგენა
3. სპეციალური სიმბოლოები
4. ხშირი შეცდომები და გადაწყვეტები
5. სასარგებლო რჩევები
6. შეჯამება

## შესავალი

წარმოიდგინე, რომ ლინუქსი არის უზარმაზარი შენობა, სადაც ათასობით ოთახი (საქალალდე) და დოკუმენტი (ფაილი) არის განთავსებული. ნავიგაციის ბრძანებები არის შენი რუკა და კომპასი, რომელიც გეხმარება ამ შენობაში გადაადგილებაში.

ლინუქსში ყველაფერი ფაილებითა და საქალალდეებით არის ორგანიზებული. გრაფიკული ინტერფეისის (როგორც Windows-ში) გარეშე, შენ უნდა იცოდე როგორ "იარო" ამ სისტემაში ბრძანებების საშუალებით. ეს პირველი და ყველაზე მნიშვნელოვანი უნარია, რომელიც ლინუქსში უნდა ისწავლო.

# 1. ძირითადი კონფიგურაციები

## 1.1 საქალალდის სტრუქტურა (Directory Structure)

ლინუქსში ყველაფერი იწყება "ფესვიდან" - / (slash). ეს არის ყველაზე მთავარი საქალალდე, საიდანაც სხვა ყველა საქალალდე იშლება, როგორც ხის ტოტები.

მაგალითი:

```
/ (ფესვი)
└── home (სახლი - აქ ინახება მომხმარებლების ფაილები)
    ├── giorgi
    └── nino
── etc (კონფიგურაციები)
── var (ცვლადი მონაცემები)
└── tmp (დროული ფაილები)
```

## 1.2 გზა (Path)

გზა არის "მისამართი", რომელიც გიჩვენებს სად არის ფაილი ან საქალალდე. არსებობს ორი ტიპი:

აბსოლუტური გზა: იწყება / -დან და გიჩვენებს სრულ მისამართს

- მაგალითი: /home/giorgi/documents/essay.txt

ფარდობითი გზა: იწყება იქიდან, სად ახლა იმყოფები

- მაგალითი: თუ ახლა home/giorgi -ში ხარ, მაშინ documents/essay.txt

## 1.3 მიმდინარე დირექტორია (Current Directory)

ეს არის საქალალდე, სადაც ამ წუთში იმყოფები. როგორც GPS-ზე შენი ლოკაცია.

# 2. ძირითადი ნავიგაციის ბრძანებები

## 2.1 pwd - სად ვარ?

PWD = Print Working Directory (დაბეჭდე სამუშაო დირექტორია)

რას აკეთებს: გიჩვენებს სად იმყოფები ახლა.

როგორ გამოიყენო:

```
pwd
```

მაგალითი:

```
$ pwd  
/home/giorgi/documents
```

ეს ნიშნავს, რომ შენ ახლა `/home/giorgi/documents` საქაღალდეში ხარ.

რატომ არის საჭირო: ხშირად დაიბნევი და არ იცი სად ხარ. `pwd` არის შენი "სად ვარ მე?" ლილაკი.

## 2.2 ls - რა არის აქ?

`ls` = List (სია)

რას აკეთებს: გიჩვენებს ყველა ფაილს და საქაღალდეს მიმდინარე დირექტორიაში.

ძირითადი გამოყენება:

```
ls
```

მაგალითი:

```
$ ls  
documents pictures music videos
```

სასარგებლო ვარიანტები:

`ls -l` (დეტალური ინფორმაცია):

```
$ ls -l  
total 32  
drwxr-xr-x 2 giorgi giorgi 4096 იან 15 10:30 documents  
drwxr-xr-x 5 giorgi giorgi 4096 იან 14 09:15 pictures
```

აქ ხედავ: უფლებებს, მფლობელს, ზომას, თარიღს.

`ls -a` (ყველაფერი, ფარული ფაილების ჩათვლით):

```
$ ls -a  
. . . hidden_file documents pictures
```

ფაილები, რომლებიც იწყებიან `.` -ით, ფარულია.

`ls -lh` (ზომა ადამიანისთვის გასაგები ფორმატით):

```
$ ls -lh  
-rw-r--r-- 1 giorgi giorgi 2.5M იან 15 10:30 video.mp4
```

2.5M უკეთესია ვიდეო `2621440` ბაიტი!

კომბინირებული: `ls -lah` (ყველა ფუნქცია ერთად)

## 2.3 cd - საქაღალდეში შესვლა

CD = Change Directory (შეცვალე დირექტორია)

რას აკეთებს: გადაგიყვანს ერთი საქაღალდიდან მეორეში.

კონკრეტულ საქაღალდეში შესვლა:

```
cd documents
```

აბსოლუტური გზით:

```
cd /home/giorgi/pictures
```

უკან დაბრუნება (ერთი დონით):

```
cd ..
```

`..` ნიშნავს "მშობელ საქაღალდეს" (ერთი დონით ზემოთ).

მაგალითი:

```
$ pwd  
/home/giorgi/documents/work  
$ cd ..  
$ pwd  
/home/giorgi/documents
```

მთავარ საქაღალდეში დაბრუნება:

```
cd ~
```

ან უბრალოდ:

```
cd
```

~ (tilde) არის შენი home დირექტორიის სიმბოლო ( /home/giorgi ).

წინა საქაღალდეში დაბრუნება:

```
cd -
```

ეს გადაგიყვანს იმ საქაღალდეში, სადაც უკანასკნელად იყავი.

მაგალითი:

```
$ pwd  
/home/giorgi/documents  
$ cd /var/log  
$ pwd  
/var/log  
$ cd -  
$ pwd  
/home/giorgi/documents
```

პრაქტიკული მაგალითები:

სიტუაცია 1: შენ home-ში ხარ და გინდა documents-ში შესვლა:

```
$ pwd  
/home/giorgi  
$ ls  
documents downloads pictures  
$ cd documents  
$ pwd  
/home/giorgi/documents
```

სიტუაცია 2: ლრმად ჩახვედი და უკან გინდა დაბრუნება:

```
$ pwd  
/home/giorgi/documents/work/projects/2024/january  
$ cd ~  
$ pwd  
/home/giorgi
```

## 2.4 tree - ვიზუალური წარმოდგენა

რას აკეთებს: აჩვენებს საქალალდის სტრუქტურას ხის სახით (ძალიან სასარგებლოა!).

გამოყენება:

```
tree
```

მაგალითი:

```
$ tree  
.  
├── documents  
│   ├── essay.txt  
│   └── work  
│       └── report.pdf  
└── downloads  
└── pictures  
    ├── photo1.jpg  
    └── photo2.jpg  
  
4 directories, 4 files
```

შენიშვნა: შესაძლოა დაგჭირდეს დაინსტალირება:

```
sudo apt install tree
```

### 3. სპეციალური სიმბოლოები

სიმბოლო	მნიშვნელობა	მაგალითი
/	ფესვის დირექტორია	cd /
~	შენი home დირექტორია	cd ~ ან cd
.	მიმდინარე დირექტორია	cd .
..	მშობელი დირექტორია (ერთით ზემოთ)	cd ..
-	წინა დირექტორია	cd -

### 4. ხშირი შეცდომები და გადაწყვეტები

#### შეცდომა 1: "No such file or directory"

```
$ cd documens  
bash: cd: documens: No such file or directory
```

პრობლემა: არასწორად დაწერე საქაღალდის სახელი ( documens ნაცვლად documents ).

გადაწყვეტა:

- გამოიყენე 1s რათა ზუსტად დაინახო სახელები
- გამოიყენე Tab კლავიში ავტომატური შევსებისთვის

#### შეცდომა 2: დაბნეული ხარ სად იმყოფები

გადაწყვეტა: ყოველთვის გამოიყენე pwd რომ შეამოწმო სად ხარ.

#### შეცდომა 3: ვერ ხედავ ფარულ ფაილებს

გადაწყვეტა: გამოიყენე 1s -a ან 1s -la .

## 5. სასარგებლო რჩევები

გამოიყენე `Tab` კლავიში: თუ იწყებ საქაღალდის სახელის აკრეფას და დააჭირ `Tab`-ს, ლინუქსი ავტომატურად შეავსებს სახელს.

მაგალითი:

```
cd doc[Tab]  
# გადაიქცევა:  
cd documents/
```

ისწავლე სიმბოლოების გამოყენება: `~`, `..`, `-` ძალიან დაგზოგავს დროს.

გამოიყენე ისტორია: დააჭირე ზედა ისარს კლავიატურაზე წინა ბრძანებების სანახავად.

ყოველდღიური პრაქტიკა: დაუთმე 10-15 წუთი ნავიგაციას რეგულარულად.

## 6. შეჯამება

ბრძანება	რას აკეთებს	მაგალითი
<code>pwd</code>	აჩვენებს სად ხარ	<code>pwd</code>
<code>ls</code>	აჩვენებს რა არის აქ	<code>ls -lah</code>
<code>cd directory</code>	გადახვედი საქაღალდები	<code>cd documents</code>
<code>cd ..</code>	ერთით უკან	<code>cd ..</code>
<code>cd ~</code> <code>cd</code>	home-ში დაბრუნება	<code>cd ~</code>
<code>cd -</code>	წინა საქაღალდები	<code>cd -</code>

ნავიგაციის ბრძანებები არის საფუძველი, რომელზეც აშენდება შენი ყველა სხვა ლინუქსის უნარი. როცა ამ ბრძანებებს დაეუფლებ პრაქტიკით, ყველაფერი სხვა გაცილებით მარტივი გახდება.