

Terminal i osnovne Linux komande

Zašto komanda linija?

Iako su svuda oko nas grafička okruženja koja su ugodna za rad i delovala da će učiniti komandnu liniju nepotrebnom. Međutim, mnogi moderni alati su orijentisani ka komandnoj liniji i jedan takav ćete videti na narednom bloku - sistem za kontrolu verzija Git. Štaviše, u mnogim situacijama više mogućnosti se nudi pomoću komadne linije nego postojećih grafičkih okruženja.

Mi ćemo koristiti Git Bash, a kasnije ćete se upoznati sa alatom Git za kontrolu verzija za projekte koje kreirate.

Git Bash je deo Git-a za Windows koji je jednostavan za instalaciju. Besplatan i otvorenog koda.

Instalacija Git-a za Windows koja sadrži Git Bash terminal.

Instalacija Git-a za Windows koja sadrži Git Bash terminal.

Postoji više načina da se instalira Git on windows. Jedan je da se koristi projekat: "Git for Windows". U pregledaču po izboru (npr. Google Chrome) idite na <https://gitforwindows.org/>

S obzirom da je 64bitni Windows koji koristimo (za instalaciju na vašem računaru proverite), biće korišćena 64bitna verzija Git for Windows.

Klinkenmo na download i kada se instaler preuzme kliknemo na "Open". Time se pokreće instaler, a Windows proverava da li smo sigurni da hoćemo da pokrenemo instalaciju. Klinkenmo na "Yes".

Kada se Git setup wizard pokrene, klinkenmo na "Next". Pa opet klinkenmo na "Next" da prihvatimo "license agreement".

Sledeći korak bira lokaciju na kojoj će Git biti instaliran. Nema potrebe da menjamo (a možemo), pa klinkenmo na "Next".

Na stranici "Select Components", možete dodati da I opciju Git Bash icon on the desktop, pa klinkenmo na "Next".

Naredno, biramo koji delovi Git i Git Bash će biti instalirani. Odaberemo srednju opciju: to će instalirati Git Bash, kao i integraciju sa Windows command prompt. Pa opet klinkenmo na "Next".

Na Windows-u, završavanje linije teksta se razlikuje od onog u tekstualnim fajlovima na Linux ili Mac operativnim sistemima. U zavisnosti da li će projekti koje koristimo biti razvijani na različitim platformama. Možemo da odaberemo srednju opciju: to checkout as-is, that is don't change anything, but then commit Unix-style. To podrazumeva da kada se koristi Git za commit, tekstualni fajlovi će biti konvertovani automatski da odgovaraju Unix-stilu. Pa opet klinkenmo na "Next".

Naredna strana, "Enable file system caching" nije potrebno, i možemo odčekirati ovu opciju. Pa opet klinkenmo na "Next" I čekamo.

Kada je process instalacije završen, možemo da na poslednjoj stranici odčekiramo opciju da vidimo README. Pa opet klinkenmo na dugme "Finish".

Na Desktop-u se nalazi ikonica Git Bash. Otvorimo terminal i da bismo se uverili da je Git instaliran možemo otkucati

```
git version
```

pa zatim pritisnuti dugme Enter na tastaturi. Git treba da odgovori koja verzija je instalirana. Da bismo zatvorili Git Bash, možemo kliknuti na dugme za zatvaranje prozora kao i uvek na Windows-u ili korišćenjem komande:

```
exit
```

Komande

U nastavku su komande koje smo mi radili, a detaljnije linux komande možete naći na:

<http://poincare.matf.bg.ac.rs/~maricm/komande.html>

Gde se nalazimo?

U bash-u, "~" predstavlja korisnikov home direktorijum. Sa pwd komandom dobijamo trenutnu poziciju (trenutni položaj u stablu direktorijuma). Komanda ls izlistava sadržaj, a ukoliko navedemo opciju -l daje opširnije informacije).

```
pwd
ls
ls -l
```

Promena direktorijuma (Change Directory)

Komanda `cd` (*Change directory*) omogućava promenu direktorijuma. `cd ..` – Vraćanje u prethodni direktorijum, dok `cd ../../` vraćanje dva koraka unazad. Voditi računa ukoliko naziv direktorijuma ima razmak (blanko karakter)npr. `My\ Documents/`.

```
pwd
ls -l
cd Videos/
pwd
cd ..
pwd
cd My\ Documents/
pwd
cd ../../..
ls -l
cd ~
cd ../../..
cd
cd /c/Windows/System32/
pwd
cd
```

Da li postoji i gde se nalazi?

Gde je (ukoliko postoji) na disku fajl koji se startuje tim imenom.

```
which ls
which man
```

Echo...Echo

Ispis teksta, ispis vrednosti promenljive...

```
echo "Hello World"
echo $PATH
echo "The current path is $PATH"
```

Sadržaj dokumenata

Čitanje ili dopisivanje u fajl.

```
cat test.txt
less test.txt
```

Ukoliko se koristi `less`, kada završimo sa gledanjem dokumenta možemo da izađemo pritiskom na dugme `q` na tastaturi.

Kreiranje, promena imena i brisanje dokumenata (engl. Files)

`touch <fajl>` - Dotakne fajl, ukoliko ne postoji kreira ga. Ukoliko postoji ažurira mu vreme kada je poslednji put promenjen.

`mv <fajl> <novo ime fajla>` – Move – Slično kao kopiranje, samo što se originalni fajl ne}e sačuvati, tj. ovako se vrši preimenovanje.

`mv <lista fajlova> <direktorijum>` – Promena mesta boravka fajlova.

`rm <lista fajlova>` – Remove – Brisanje fajlova.

```
pwd
ls
touch demo.txt
ls
mv demo.txt demo-1.txt
ls
rm demo-1.txt
ls
ls -l
touch test.txt
ls -l
```

Kreiranje i brisanje direktorijuma (engl. Folders)

`mkdir <direktorijum>` – Make directory – Pravljenje direktorijuma.

`rmdir <direktorijum>` – Remove directory – Briše samo prazne direktorijume.

`rm -r <direktorijum>` – Briše ceo <direktorijum> bez daljih pitanja.

Sa `mkdir -p projects/client-a/awesome-web-project` može se kreirati više ugnježenih direktorijuma odjednom.

```
pwd
mkdir projects
cd projects/
pwd
cd ..
rmdir projects
cd projects
ls
mkdir -p projects/client-a/awesome-web-project
cd projects/client-a/awesome-web-project
cd
rmdir projects
rm -rf projects/
ls
```

Clear & Exit

`clear` – Briše ekran.

`exit` – Zatvaranje tekućeg prozora (u našem slučaju Git bash-a).

```
ls -l
clear
exit
```

Redirekcija izlaza pomoću > ili >> (Output to a File)

`echo "Neki novi tekst" >> <fajl>` – Dopisuje poruku na kraj fajla.

```
pwd
```

```
echo "hello there" >> demo.txt
ls
cat demo.txt
echo "hello again" >> demo.txt
cat demo.txt
echo "HI" > demo.txt
cat demo.txt
clear
ls -al > listing.txt
cat listing.txt
```

Pretraga datoteka

```
find X -name "Y"
```

Umesto *X* se stavlja putanja do lokacije na računaru na kojoj želimo da pretražujemo. Ako želimo da pretražimo ceo računar, onda stavimo */*. Korišćenje *.* pretražuje samo u trenutnom direktorijumu. *~/* pretražuje ceo home direktorijum.

Umesto *Y* (pod navodnicima) se navodi kriterijum pretrage.

Može se tražiti i određeni tip datoteka korišćenjem zvezdice (an asterisk, ***). Npr. Za pretragu svih *".jpg"* dokumenata:

```
find . -iname "*.jpg"
```

Pretraga teksta u datotekama

Sintaksa:

```
grep [options] pattern [files]
```

-c : This prints only a count of the lines that match a pattern

-h : Display the matched lines, but do not display the filenames.

-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

-e exp : Specifies expression with this option. Can use multiple times.

-f file : Takes patterns from file, one per line.

-E : Treats pattern as an extended regular expression (ERE)

-w : Match whole word

-o : Print only the matched parts of a matching line, with each such part on a separate output line.

Pretažujemo *PATTERN* "nav" u našem dokumentu primer.html.

```
grep "nav" primer.html
grep --color "nav" primer.html
grep --color -n "nav" primer.html
grep --color -n -i "nav" primer.html
grep --color -n -i "nav" *.html
grep --color -n -i -r "nav" *.html
```

Kreiranje i izvršavanje jednostavnog bash skripta (Simple Bash Script)

```
pwd
which bash
notepad++ example.sh ili notepad example.sh
```

```
#!/usr/bin/bash

echo "Hi, everyone!"
```

```
ls -al
chmod +x example.sh      #Ukoliko je potrebno (nije potrebno ukoliko je example.sh vec izvršiv executable)
./example.sh
```

Podešavanje Bash okruženja (Bash Environment)

```
notepad .bashrc
alias ll= 'ls -al'
source .bashrc
ll # sad moze umesto ls -al
```

Otvaranje terminala iz Visual Studio Code:

- Koristiti precicu **Ctrl+`**
- **View > Terminal**