

Podstawy Linux i Bash 1

Wojciech Barczyński

Spis treści

Spis treści	2
1. Podstawy bash	3
2. Praca z plikami	10
3. Praca z serwisami web z terminala.....	12

1. Podstawy basha

Linux (Centos lub Ubuntu) jest doskonałym środowiskiem do testowania, który możemy dowolnie dostosować do naszych indywidualnych potrzeb.

1. Uruchom Terminal:

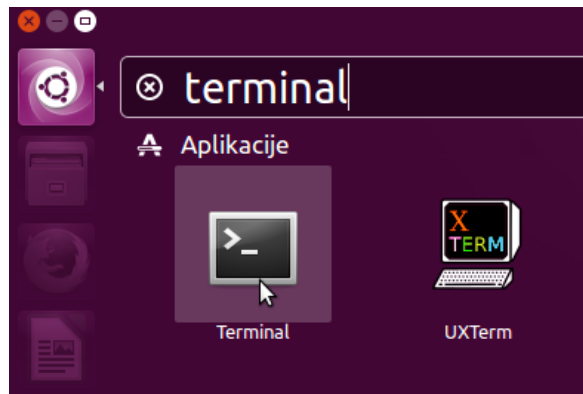


Figure 1 Ubuntu

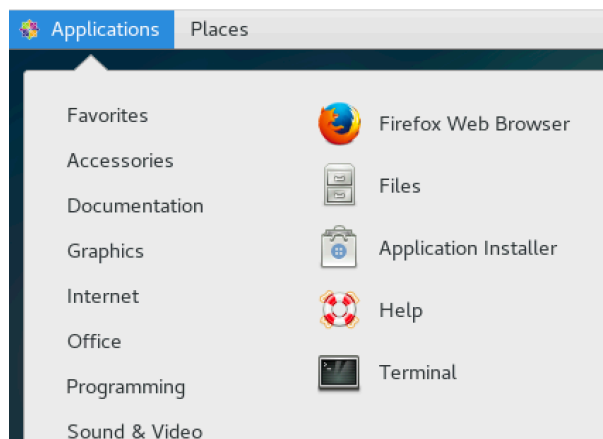


Figure 2 CentOS

2. Otworzenie folderu w interfejsie graficznym:

```
$ nautilus .  
$ xdg-open .
```

Czyli:

```
nautilus /ścieżka/do/folderu  
xdg-open /ścieżka/do/folderu
```

3. Przejrzyj zawartość katalogu \$HOME, z pomocą ls :

```
$ ls  
$ ls $HOME  
$ ls ~  
$ echo $HOME
```

4. Survival 2: gdzie jestem, zanotuj gdzie jesteś:

```
$ pwd
```

5. Survival 3: Powrót do domu:

```
# przejście do głównego katalogu / to C: w Windowsie
$ cd /
$ pwd
$ cd ~
```

```
$ cd /tmp
# po zmiennej w większości automatyzacji
$ cd $HOME
$ cd ../../..
```

6. Przygotowanie do pracy z katalogami:

```
$ sudo apt-get update
$ sudo apt install tree
```

Sprawdź: <https://xkcd.com/149/>

7. Praca z katalogami:

```
$ mkdir polska
$ cd polska
$ mkdir miasta
$ cd miasta
$ mkdir katowice
$ ls
$ mkdir warszawa
$ mkdir sosnowice
$ ls
$ cd ..
$ pwd
$ cd ..
$ tree
```

Powinieneś zobaczyć:

```
polska
├── miasta/
│   ├── katowice/
│   ├── warszawa/
│   ├── wroclaw/
│   └── sosnowiec/
```

Zauważ:

```
$ pwd
$ cd miasta/katowice # do katowic
$ pwd

# wracamy
$ cd ../..
$ pwd
```

Zwróć uwagę na ścieżki:

```
$ ls miasta
$ ls ..
$ ls /home/tester

$ pwd
```

Dodaj do katalogu polska, katalog wsie z katalogami *nowawies* i *zytno*:

```
polska
|- miasta/
|   |- katowice/
|   |- warszawa/
|   |- wroclaw/
|   \- sosnowiec/
|
\-- wsie/
    |- nowawies
    \- zytno
```

8. Szybki start z **vi** i **vim**, aby przeżyć¹:

```
# zainstalujemy vima
$ sudo apt-get install vim
```

pamiętaj:

1. ESC ESC
2. :q
3. ENTER

Uuuu.. jednak coś wpisałem i nie chce zapisać:

1. ESC ESC
2. :q !
3. ENTER

¹ <https://xkcd.com/378/>

Jednak chce zapisać:

1. ESC ESC
2. :wq
3. ENTER

Tryb dodawania:

1. ESC ESC
2. i
3. Teraz piszemy :D

9. Szybki start z emacs / nano :

```
$ nano
```

Mamy jeszcze gedit:

```
$ gedit p.txt
```

10. Praca z plikami:

```
$ touch myfile.txt # pusty plik
```

```
$ echo "Ubuntu 16.10" > myfile.txt  
$ echo "Ubuntu 17.04" >> myfile.txt  
$ cat myfile.txt  
$ echo "Mint" >> myfile.txt  
$ echo "Debian" >> myfile.txt  
$ echo "Linux" >> myfile.txt  
$ cat myfile.txt
```

11. Zmodyfikuj zawartość pliku z pomocą atom-a:

```
$ whereis atom  
$ atom myfile.txt
```

12. Wróć do terminalu i wyszukaj słowa w pliku:

```
$ grep Linux myfile.txt  
$ grep Ubuntu myfile.txt
```

Zadanie:

1. Wyszukaj swojego użytkownika w /etc/passwd
2. Wyszukaj ERROR, dodaj -i do komendy *grep*

13. Analiza logu:

```
$ grep -i Error /var/log/*log  
# alternatywa wc -l  
$ grep -c -i Error /var/log/*log
```

14. Skopiuj plik:

```
$ cp myfile.txt myfile2.txt
$ ls
# wyróżnienie regex
$ grep Linux *.txt
```

15. Przenieś plik (lub zmień jego nazwę):

```
$ mkdir mydirectory
$ mv myfile2.txt mydirectory
$ ls mydirectory
$ tree .
# i spowrotem
$ mv mydirectory/myfile2 .
```

16. Przeniesienie /zmiana nazwy katalogu, zmieńmy **miasta** na **cities**:

```
$ cd ~
$ cd polska
$ ls
$ mv miasta cities
$ ls
```

Skopiuj:

```
$ cp -r cities miasta
```

17. Skasuj plik

```
# wróćmy do katalogu z myfile2
$ pwd
$ rm myfile2.txt
```

18. Skasuj cały folder:

```
$ pwd
$ ls polska
$ rmdir polska/miasta/warszawa
$ rm -r polska
```

19. Utwórz następujące katalogi i pliki (pamiętaj o komendzie *tree*):

```
biologia
|- drzewa
|   |- liściaste.txt
|   \- iglaste.txt
|
\-- zwierzeta
     |- drapieżniki
     |   \- koty.txt
     \- roślinozerne
         \- rogate.txt
```

Po utworzeniu.

1. Przenieś *koty.txt* i *rogate.txt* bezpośrednio do katalogu *zwierzeta*.
2. Następnie przekopiuj *iglaste.txt* do katalogu domowego.
3. Skasuj wszystkie pliki i katalogi.

20. Pliki ukryte:

```
$ ls -a
$ ls -la
$ ls -la | grep bash
```

21. Wyświetl zmienne środowiskowe:

```
$ printenv
$ printenv | grep HOME=
$ printenv | grep LANG=
```

HOME	
LANG	

22. Zmienne środowiskowe i *echo*:

```
$ cd ~
$ mkdir tmp
$ cd tmp
# porównaj:
$ echo "$HOME"
$ echo '$HOME'
# rekomendacja, zawsze z { i }
$ echo "${HOME}"
```

23. Zmienne środowiskowe i *echo* i plik:

```
$ cd ~
$ mkdir tmp
$ cd tmp
# porównaj:
$ echo "$HOME" > home.txt
$ cat home.txt
```


24. Zmienne środowiskowe i *\$HOME/.bashrc*:

```
$ atom ~/.bashrc
# przejdź na koniec pliku i napisz, zapisz i zamknij:

echo "Witam ${USER}!"
export MOJ_TELEFON=9999

# przy uruchomieniu basha, powinniśmy zobaczyć powitanie
$ bash
$ echo $MOJ_TELEFON
```

25. Ad hoc definicja zmiennych środowiskowych:

```
$ export SHOP_USER=natalia
$ echo "$SHOP_USER"
```

2. Praca z plikami

Poznamy narzędzia najczęściej używane do przetwarzania i przygotowania danych testowych oraz logów.

1. Otrzymałeś plik *koszyk1.txt* o treści:

```
mleko,10,1z1  
chleb,1,4z1
```

Używając `cut`:

```
cut -d',' -f1 koszyk1.txt  
cat koszyk1.txt | cut -d',' -f1
```

wypisz na ekran:

```
1z1  
4z1
```

Dodatkowe 1: wyświetl nazwy użytkowników z `/etc/passwd`.

Dodatkowe 2: praca z `/var/log`.

2. Przeglądasz logi i znajdujesz zrzut z obiektu reprezentującego produkt do pliku *koszyk2.txt*:

```
product=mleko  
cena=1  
product=chleb  
cena=4
```

Korzystając z funkcji `cut` oraz `paste`, utwórz plik:

```
mleko 1  
chleb 4
```

Podpowiedź:

```
cat produkty2.txt | paste - -
```

3. Zastępowanie łańcuchów znaków, masz plik *koszyk3.txt* z treścią:

```
product_category: mleczne
```

Chciałbyś zastąpić *mleczne*, flagom produktu *M0*:

```
product_category: M0
```

Podpowiedź:

```
sed 's/mleczne/M0/g'
```

Warto również znać *tr*:

```
cat produkty3.txt | tr ':' '='
```

4. Zamień w poniższym zdaniu **kota** na **psa**, wykorzystując **sed**:

```
echo "Ala ma kota" | sed ...
```

5. Co oznaczają **head**, **tail**, **less**?

```
less /var/log/syslog  
head -n 1 /var/log/syslog  
tail -n 2 /var/log/syslog
```

6. [Dodatkowe] Liczenie w konsoli:

```
magazyn  
|- słodyczne  
|   \- produkty.txt  
|       wedel,2  
|       goplana,3  
|  
|- mleczne  
|   \- produkty.txt  
|       mlekovita,5  
|       zimnemleko,4
```

1. Wypisz za pomocą jednej komendy:

```
2  
3  
5  
4
```

2. Zsumuj powyższe wartości w bashu, znajdź komendę w google-u (stackoverflow).

3. Praca z serwisami web z terminala

Wprowadzenie do innych narzędzi, na następnych zajęciach omówimy dokładniej serwisy REST oraz protokół http:

1. Wykonaj prosty test z pomocą curl:

```
$ curl google.com

# dlaczego -L ?
$ curl -L google.com

$ curl -I -L -s google.com
$ curl -L -I -s google.com | grep HTTP
```

2. Możemy też skonfigurować curl-a aby nam tylko zwracał kod http:

```
# curl pokaże nam tylko kod http
$ curl -s -o /dev/null -w "%{http_code}" google.com

# co się zmieni kiedy dodamy -L
$ curl -s -o /dev/null -w "%{http_code}" google.com -L
```

3. Wyszukaj informacji o Twoim ulubionym zwierzęciu:

```
$ curl -L -A 'Mozilla/5.0 (MSIE; Windows 10)' \
-o chomik.html \
https://www.google.com/search?q=chomik
```

Zanotuj:

Co oznacza -A?	
Co oznacza q?	

Podpowiedź: zobacz w przeglądarce, developer tools. Wykładowca pokaże jak.

4. [Dodatkowe] Otwórz w przeglądarce przykładowy API endpoint:

<http://services.odata.org/>
<http://services.odata.org/V3/OData/OData.svc/Products>

5. [Dodatkowe] Wykorzystaj curl (zainstaluj *libxml2-utils*):

```
$ curl \
-s http://services.odata.org/V3/OData/OData.svc/Products -L \
| xmllint --format -
```

Alternatywy: <https://httpie.org/> (`apt-get install httpie`) albo www.getpostman.com.