

polecenia powłoki

Powłoka systemowa (ang. *shell*) to program będący pośrednikiem pomiędzy systemem operacyjnym a użytkownikiem. Można powiedzieć, że pełni funkcję interpretera kolejno wpisywanych poleceń. W systemach uniksowych najpopularniejszą powłoką jest *bash* (ang. *Bourne-Again SHell*). Podstawowe polecenia powłoki (nie tylko basha):

<code>passwd</code>	zmiana hasła użytkownika
<code>pwd</code>	aktualny katalog wraz ze ścieżką dostępu od korzenia
<code>clear</code>	czyszczenie ekranu terminala
<code>exit</code>	zamknięcie powłoki
<code>ls</code>	zawartość (pliki) podanego katalogu
<code>cd</code>	zmiana katalogu na podany
<code>man</code>	pomoc (dokumentacja) dla podanego polecenia
<code>touch</code>	uaktualnienie czasu ostatniej modyfikacji pliku (tworzenie go, jeśli nie istniał)
<code>mkdir</code>	tworzenie katalogu o podanej nazwie
<code>rm</code>	usuwanie plików
<code>rmdir</code>	usuwanie (pustych) katalogów
<code>cp</code>	kopiowanie plików
<code>mv</code>	przenoszenie (zmiana nazwy) plików
<code>chmod</code>	zmiana praw własności plików
<code>echo</code>	wyświetlanie podanej linii tekstu
<code>cat</code>	łączenie plików i wypisywanie ich na standardowe wyjście
<code>tac</code>	to samo co <code>cat</code> , ale linie w odwrotnej kolejności
<code>wc</code>	liczenie linijek, słów i znaków danego pliku
<code>head</code>	wyświetlanie początkowych linijek pliku
<code>tail</code>	wyświetlanie końcowych linijek pliku
<code>sort</code>	sortowanie linii
<code>uniq</code>	omijanie powtarzających się linii

Przykłady użycia:

<code>passwd</code>	patrz wyżej
<code>pwd</code>	patrz wyżej
<code>clear</code>	patrz wyżej
<code>exit</code>	patrz wyżej
<hr/>	
<code>ls</code>	zawartość aktualnego katalogu
<code>ls Polska</code>	zawartość katalogu "Polska"
<code>ls -l</code>	zawartość aktualnego katalogu z dodatkowymi informacjami o plikach (np. atrybuty, przynależność)
<code>ls -a</code>	zawartość aktualnego katalogu wraz z plikami ukrytymi (ich nazwy zaczynają się od kropki)
<code>ls -la Polska</code>	zawartość katalogu Polska wraz z plikami ukrytymi i z dodatkowymi informacjami o plikach
<code>ls -R</code>	zawartość aktualnego katalogu i jego podkatalogów

<code>cd</code>	zmiana katalogu na katalog domowy użytkownika
<code>cd Polska</code>	zmiana katalogu na "Polska"
<code>cd ..</code>	zmiana katalogu na nadrzędny względem aktualnego
<code>cd .</code>	zmiana katalogu na aktualny katalog (nic się nie dzieje)
<code>cd /</code>	zmiana katalogu na katalog główny
<code>cd ~</code>	zmiana katalogu na katalog domowy użytkownika (to samo, co polecenie <code>cd</code> bez argumentów)
<code>cd ../Polska/Warszawa</code>	przykład zmiany katalogu (ścieżka względna): katalog Warszawa w katalogu Polska, który znajduje się w katalogu nadrzędnym względem aktualnego katalogu
<code>cd /home</code>	przykład zmiany katalogu (ścieżka bezwzględna): katalog "home" w katalogu głównym
<hr/>	
<code>man cd</code>	pomoc dla polecenia <code>cd</code>
<code>man man</code>	pomoc dla polecenia <code>man</code>
<hr/>	
<code>touch a b</code>	tworzenie (aktualizacja daty modyfikacji) plików "a" i "b"
<code>touch a\ b</code>	tworzenie (aktualizacja daty modyfikacji) pliku "a b"
<code>touch "a b"</code>	tworzenie (aktualizacja daty modyfikacji) pliku "a b"
<hr/>	
<code>mkdir Niemcy</code>	utworzenie katalogu "Niemcy"
<code>mkdir Niemcy/Berlin</code>	utworzenie katalogu "Berlin" w istniejącym katalogu "Niemcy"
<code>mkdir -p Francja/Paryz</code>	utworzenie katalogu "Paryz" w NIEKONIECZNIE istniejącym katalogu "Francja" (może zostać utworzony)
<hr/>	
<code>rm a b</code>	usuwanie plików "a" i "b"
<code>rm *</code>	usuwanie wszystkich plików z aktualnego katalogu poza plikami ukrytymi
<code>rm * .*</code>	usuwanie wszystkich plików (także ukrytych) z aktualnego katalogu
<code>rm *test*</code>	usuwanie wszystkich nieukrytych plików, które posiadają słowo "test" w nazwie
<code>rm m[aiu]d</code>	usuwanie plików "mad", "mid", "mud", o ile te istnieją
<code>rm *.txt</code>	usuwanie nieukrytych plików o nazwach kończących się na ".txt"
<code>rm -r Niemcy</code>	usuwanie katalogu "Niemcy" z całą zawartością
<code>rm -r Polska/Krakow</code>	usuwanie podkatalogu "Krakow" z katalogu "Polska"
<hr/>	
<code>rmdir Francja</code>	usuwanie katalogu "Francja", o ile ten jest pusty
<hr/>	
<code>cp x y</code>	kopiowanie zawartości pliku "x" do pliku "y"
<code>cp x y Polska</code>	kopiowanie plików "x" i "y" do katalogu "Polska"
<code>cp c?t Polska</code>	kopiowanie plików trzyliterowych o pierwszej literze "c", drugiej dowolnej i trzeciej "t" do katalogu "Polska"
<code>cp * Francja</code>	kopiowanie wszystkich nieukrytych plików (ale nie nieukrytych katalogów!) do katalogu "Francja"
<code>cp -r Polska USA</code>	jeżeli katalog USA istnieje, to kopiowanie katalogu "Polska" do katalogu "USA"; w.p.p. utworzenie katalogu "USA" o takiej samej zawartości jak katalog "Polska"

<code>mv x y</code>	zmiana nazwy pliku z "x" na "y"
<code>mv x y Polska</code>	przeniesienie plików "x" i "y" do katalogu "Polska"
<code>mv a[~cr]t Polska</code>	przeniesienie plików trzyliterowych o pierwszej literze "a", drugiej innej niż "c", "r" i trzeciej "t" do katalogu "Polska"
<code>mv fr* Francja</code>	przeniesienie wszystkich plików i katalogów o nazwach zaczynających się od "fr" do katalogu "Francja"
<code>mv Polska USA</code>	jeżeli katalog USA istnieje, to przeniesienie katalogu "Polska" do katalogu "USA"; w.p.p. zmiana nazwy katalogu "Polska" na "USA"

<code>chmod 764 p1</code>	zmiana praw własności do pliku "p1" na: właściciel - prawo do czytania(4), zapisu(2) i wykonania(1), grupa - tylko prawo do czytania(4) i zapisu(2), inni - tylko prawo do czytania(4)
<code>chmod 764 Kat -R</code>	zmiana praw własności do katalogu "Kat" i całej jego zawartości na podane wyżej
<code>chmod a+r p2</code>	nadanie praw wszystkim do czytania pliku "p2"
<code>chmod +r p2</code>	również nadanie praw wszystkim do czytania pliku "p2"
<code>chmod u+w p2</code>	nadanie praw użytkownikowi do zapisu pliku "p2"
<code>chmod g-w p2</code>	usunięcie praw grupie do zapisu pliku "p2"
<code>chmod o-x p2</code>	usunięcie praw innym do wykonania pliku "p2"
<code>chmod u=rw p2</code>	zmiana praw użytkownika do pliku "p2" na: czytanie, zapis, brak możliwości wykonania
<code>chmod go=x p2</code>	zmiana praw grupy i innych do pliku "p2" na: brak możliwości czytania, brak możliwości zapisu, wykonanie

<code>echo "Hej"</code>	wyświetlanie Hej na standardowym wyjściu
<code>echo "Hej\nHej"</code>	wyświetlanie Hej\nHej na standardowym wyjściu
<code>echo -e "Hej\nHej"</code>	wyświetlanie 2 linijek Hej na standardowym wyjściu

<code>cat</code>	wczytywanie ze standardowego wejścia kolejnych linijek tekstu i wypisywanie ich na standardowe wyjście (Ctrl-D kończy proces)
<code>cat input</code>	wypisywanie pliku "input" na standardowe wyjście
<code>cat < input</code>	również wypisywanie pliku "input" na standardowe wyjście
<code>cat > output</code>	wczytywanie ze standardowego wejścia kolejnych linijek tekstu i zapisywanie ich do pliku "output"
<code>cat >> output</code>	wczytywanie ze standardowego wejścia kolejnych linijek tekstu i dopisywanie ich na koniec pliku "output"
<code>cat in1 in2 > output</code>	stworzenie (nadpisanie) pliku "output" poprzez połączenie zawartości plików (kolejno) "in1" i "in2"
<code>cat -n input</code>	wypisywanie pliku "input" wraz z numerami linii
<code>cat -E input</code>	wypisywanie pliku "input" wraz z końcami linii (symbol \$)

<code>tac</code>	to samo co polecenie <code>cat</code> , ale linijki na standardowym wyjściu wypisywane są w odwrotnej kolejności
<code>tac input</code>	wypisywanie pliku "input" na standardowe wyjście od ostatniej linijki do pierwszej

<code>wc p1</code>	zliczanie (kolejno) linijek, słów i znaków pliku "p1"
<code>wc -l p1</code>	zliczanie linijek pliku "p1"
<code>wc -w p1</code>	zliczanie słów pliku "p1"
<code>wc -c p1</code>	zliczanie znaków pliku "p1"
<hr/>	
<code>head text</code>	wypisywanie pierwszych 10 linijek pliku "text"
<code>head -5 text</code>	wypisywanie pierwszych 5 linijek pliku "text"
<hr/>	
<code>tail text</code>	wypisywanie ostatnich 10 linijek pliku "text"
<code>tail -3 text</code>	wypisywanie ostatnich 3 linijek pliku "text"
<hr/>	
<code>sort</code>	wczytywanie ze standardowego wejścia kolejnych linijek tekstu i (po Ctrl-D) wypisywanie ich na standardowe wyjście w postaci posortowanej leksykograficznie
<code>sort < plik1</code>	wypisywanie posortowanych linijek pliku "plik1"
<code>sort -r < plik1</code>	wypisywanie odwrotnie posortowanych linijek pliku "plik1"
<hr/>	
<code>uniq</code>	wczytywanie ze standardowego wejścia kolejnych linijek tekstu i wypisywanie ich na standardowe wyjście, o ile kolejna wpisana linia nie jest taka sama jak jej poprzedniczka (Ctrl-D kończy proces)
<code>uniq < plik2</code>	wypisywanie linijek pliku "plik2" z ominięciem powtarzających się po sobie identycznych linijek

Jak widać, w przypadku poleceń czytających ze standardowego wejścia możliwe jest *przekierowanie* do nich zawartości danego pliku przy pomocy symbolu <. Jeżeli chcemy, aby dane polecenie efekt swoich działań zamiast na standardowe wyjście przekierowało do wybranego przez nas pliku, używamy symboli > lub >> (patrz: polecenie `cat`). Można też zrobić tak: > p1. Jeśli plik "p1" nie istnieje, to zostanie utworzony; w przeciwnym przypadku jego zawartość zostanie wyzerowana.

Jeżeli naszym celem jest potraktowanie wyjścia jednego polecenia jako wejścia do kolejnego, to możemy skorzystać z *potoków* (symbol: |). Przykładowo, jeśli chcemy posortować odwrotnie leksykograficznie pierwszych 25 linijek pliku "file1", to możemy wykorzystać następującego *jednolinijkowca* (ang. *oneliner*):

```
head -25 file1 | sort -r
```

Oczywiście, nie ma przeszkód, aby wynik polecenia wykorzystującego potoki zapisać do pliku, np. "file2":

```
head -25 file1 | sort -r >file2
```

Poprzez sprytne wykorzystanie potoków możemy poradzić sobie z tak naturalnymi problemami, jak zliczenie niepowtarzających się linii pliku "p2":

```
sort <p2 | uniq | wc -l  
cat p2 | sort | uniq | wc -l
```