SHELL PARAMETER EXPANSION

Wyrażenie \${zmienna} podstawia zamiast nazwy zmiennej jej wartość. (W przypadku zapisu \$zmienna, zmienna musi być jednoznaczna.) Domyślnie, jeżeli zmienna nie jest ustawiona bądź jej wartością jest NULL, to powłoka bash rozwija wyrażenie \${zmienna} do postaci pustego ciągu znaków.

Bash daje jednak możliwość rozszerzenia działania powyższego wyrażenia poprzez użycie odpowiedniego modyfikatora.

Modyfikator

Użyj wartości domyślnej

\${zmienna:-wartość domyślna}

Interpretacja powyższego wyrażenia przez powłokę przedstawia się następująco:

"jeżeli zmienna ma wartość NULL lub nie została ustawiona, użyj zamiast niej wartości_domyślnej, w przeciwnym wypadku użyj zmiennej". Modyfikator ten nie zmienia wartości zmiennej.

Modyfikator

Przypisz wartość domyślną

\${zmienna:=wartość_domyślna}

Interpretacja powyższego wyrażenia przez powłokę przedstawia się następująco:

"jeżeli *zmienna* ma wartość NULL lub nie została ustawiona, użyj zamiast niej *wartości_domyślnej*, w przeciwnym wypadku użyj *zmiennej*". Modyfikator ten zmienia wartość *zmiennei*.

Modyfikator :?

Wyświetl komunikat o wystąpieniu błędu

\${zmienna:?komunikat}

Interpretacja powyższego wyrażenia przez powłokę przedstawia się następująco:

"jeżeli zmienna ma wartość NULL lub nie została ustawiona, zostanie wyświetlony na ekranie komunikat o błędzie i skrypt zakończy działanie z kodem zakończenia równym 1".

(Powłoki interaktywne po użyciu modyfikatora :? nie kończą działania.)

Jeżeli *komunikat* zostanie pominięty, powłoka wyświetli domyślny komunikat o błędzie.

Modyfikator :+

\${zmienna:+wartość}

Interpretacja powyższego wyrażenia przez powłokę przedstawia się następująco:

"Jeżeli *zmienna* ma przypisaną *wartość*, użyj *wartości*. W przeciwnym razie nie rób nic".

Modyfikator : ..

\${zmienna:pozycja}

Interpretacja powyższego wyrażenia przez powłokę przedstawia się następująco:

"Rozpoczynając od *pozycji*, w *zmiennej* wydobądź wszystkie następujące po wskazanej pozycji znaki". Pozycja jest liczona od 0.

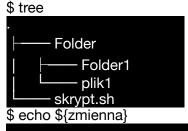
Modyfikator : ..:..

\${zmienna:pozycja:zakres}

Interpretacja powyższego wyrażenia przez powłokę przedstawia się następująco:

"Rozpoczynając od *pozycji*, w *zmiennej* wydobądź *zakres* znaków. Jeżeli *zakres* jest liczbą ujemną policz znaki od końca. Zarówno *zakres*, jak i *pozycja* mogą być wyrażeniami arytmetycznymi".

Przykłady użycia i działania:



\$ echo \${zmienna}	
	\$ echo \$
\$ Is \${zmienna:-Folder}	bash: zr
Folder1 plik1	r null or
\$ Is \${zmienna}	\$ echo \$
Folder sktypt.sh	bash: zr
\$ ls \${zmienna:=Folder}	\$ zmien
Folder1 plik1	\$ echo \$
\$ Is \${zmienna}	akota
Folder1 plik1	\$ echo \$
\$ ls \${zmienna:+Folder}	kota
Folder1 plik1	\$ echo \$
\$ ls \${zmienna1:+Folder}	kot
Folder skrypt.sh	

\$ echo \${zmienna2:?} bash: zmienna2: paramete r null or not set

\$ echo \${zmienna2:?kom} bash: zmienna2: kom \$ zmienna3=Alamakota

\$ echo \${zmienna3:4}

\$ echo \${zmienna3:5:4}

\$ echo \${zmienna3:5:-1}

Bibliografia

Robbins Arnolds (2016): Bash. Leksykon kieszonkowy. Przewodnik dla użytkowników i administratorów systemów (s. 25–27), przeł. Patryk Wierzchoń. Gliwice: Helion.

Sobell Mark G. (2013): Linux. Programowanie w powłoce. Praktyczny przewodnik. Wydanie III (s. 472–473), przeł. Robert Górczyński. Gliwice: Helion.