



Vyšší odborná škola obalové techniky
a střední škola, Štětí

Digitální učební materiály

Operační systémy - Linux

Ivan Pomykacz



esf evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Licence



Digitální učební materiály, jejímž autorem je Ivan Pomykacz, podléhají licenci [Creative Commons: Uvedte autora - Nevyužívejte dílo komerčně - Zachovejte licenci 3.0 Unported](https://creativecommons.org/licenses/by-nc-sa/3.0/).

Vytvořeno na základě tohoto díla: <http://dumy.odbornaskola.cz/pomykacz>

Práva nad rámec této licence jsou popsána zde: <http://dumy.odbornaskola.cz/pomykacz>.

Disclaimer

Tento PDF dokument byl strojově vygenerován z HTML stránek

<http://dumy.odbornaskola.cz/pomykacz/>.

Je tedy možné, že sazba textu může obsahovat chyby. Jde převážně o vizuální a typografické chyby, které mohou narušit plynulou čitelnost textu. V některých případech může být ohrožena i funkčnost některých komponent (jako vnitřní odkazy).

Vzhledem k tomu, že vypracované materiály nebyly nikdy určeny pro výstupní formát PDF, autor se zříkává jakékoli odpovědnosti za nalezené chyby. Nesnažte se proto v této souvislosti autora kontaktovat.

Operační systémy

Linux

Obsah

- Skriptování

Skriptování

Název školy	Vyšší odborná škola obalové techniky a Střední škola, Štětí, příspěvková organizace		
Adresa školky	Kostelní 134, 411 08 Štětí		
IČ	46773509		
Název operačního programu	OP Vzdělávání pro konkurenceschopnost		
Registrační číslo	CZ.1.07/1.5.00/34.1006		
Označení vzdělávacího materiálu	VY_32_INOVACE_21_PSS_417		
Název tématické oblasti (sady)	Operační systémy		
Název materiálu	Skriptování		
Anotace	Předkládá základy skriptování v shellu BASH. Ukazuje na jednoduchém příkladu způsob psaní skriptů, používání proměnných a výstupu z jiných programů.		
Autor	Ivan Pomykacz	Jazyk	český
Očekávaný výstup	Vytvoří spustitelný skript. Používá proměnné a výstupy jiných programů.		
Klíčová slova	skript, bash, proměnná		
Druh výukového zdroje	Výklad	Věková skupina žáků	17+
Typ interakce	aktivita	Ročník	3.
Speciální vzdělávací potřeby	žádné		
Zhotoveno, (datum/období)	13.5.2014	Celková velikost	

Obsah

- První skript
 - Spuštění skriptu
 - Použití interpretu
 - Práva na spuštění
 - Proměnná PATH
- Proměnné
 - Výstup programu
- Parametry při spuštění

Na Linuxu založené systémy jsou známy mimo jiné i tím, jak jsou svázány se skriptovacími jazyky (jako bash, perl nebo python). Např. pomocí bashe jsou napsány init skripty, pomocí kterých se startuje operační systém.

První skript

Vytvořme soubor `skript.sh`. Na první řádek se standardně píše speciální uvozovací řetězec, který říká, jaký interpret se má použít pro vykonání skriptu.

```
#!/bin/bash
```

```
echo "Skript běží ..."  
sleep 2s  
echo "Skript skončil."
```

Příkaz `echo` vypíše text v uvozovkách (i když ty nejsou vždy nutné). Program `sleep` pozastaví běh skriptu na dvě vřetiny.

Spuštění skriptu

Existuje několik možností jak skript spustit. Vytvořený soubor nemá práva na spuštění, a i kdyby je měl, nepůjde spustit jenom tak, že napíšeme do shellu název skriptu (jako u ostatních programů).

Použití interpretu

Tento způsob spočívá v tom, že prostě spustíme interpret bashe, který se postará o vykonání skriptu.

```
tux@wheezy:~$ bash skript.sh  
Skript běží ...  
Skript skončil.
```

Práva na spuštění

Skriptu jednoduše přidáme právo x. Následně skript spustíme tím, že k němu zadáme cestu (relativní nebo absolutní).

```
tux@wheezy:~$ chmod +x skript.sh  
tux@wheezy:~$ ls -l skript.sh  
-rwxr-xr-x 1 tux tux 68 kvě 25 21:02 skript.sh  
tux@wheezy:~$ ./skript.sh  
Skript běží ...  
Skript skončil.
```

Proměnná PATH

Nejpoužitelnější je však umístění skriptu na místo, které je definováno v proměnné `PATH`. Případně toto místo do proměnné `PATH` přidáme.

Vytvořme adresář `~/bin` a přesuňme do něj vytvořený skript, který již má nastavená práva na spuštění.

```
tux@wheezy:~$ mkdir bin  
tux@wheezy:~$ mv skript.sh bin/
```

Výhoda tohoto řešení je, že Debian počítá dopředu s tím, že si uživatel vytvoří u sebe v adresáři podadresář `~/bin`. Při dalším přihlášení bude tento adresář automaticky přidán do proměnné `PATH`.

Jakmile se tedy odhlásíte a znovu přihlásíte, můžete spustit skript pouhým zapsáním jeho názvu.

```
tux@wheezy:~$ skript.sh
Skript běží ...
Skript skončil.
```

Proměnné

BASH je skriptovací jazyk, který umožňuje deklarovat proměnné, testovat výrazy, cyklení kódu i deklarace funkcí. Proměnné se deklarují zapsáním názvu a symbolem rovná se. Hodnota se nemusí zadávat, proměnná bude prostě prázdná.

Deklarujme proměnnou `CAS` s hodnotou "5s". Proměnnou pak dosadíme do prvního příkazu `echo` a do příkazu `sleep` (na místo původních 2s).

```
#!/bin/bash

CAS=5s

echo Skript poběží po dobu $CAS ...
sleep $CAS
echo Skript skončil.
```

Výstup programu

Do proměnné lze uložit výstup jiného programu a ten pak použít dál. Takový příkaz se musí uzavřít do speciální strof.

```
#!/bin/bash

CAS=5s
DATUM=`date +%d.%m.%Y`

echo Skript byl spuštěn dne $DATUM
echo Skript poběží po dobu $CAS ...
sleep $CAS
echo Skript skončil.
```

V tomto případě ovšem proměnnou `DATUM` vůbec nepotřebujeme. Výraz ``date +%d.%m.%Y`` bychom mohli napsat přímo do příkazu `echo`.

```
tux@wheezy:~$ skript.sh
Skript byl spuštěn dne 25.02.2014
Skript poběží po dobu 5s ...
Skript skončil.
```


Parametry při spuštění

Při spuštění lze skriptu předat parametry, se kterými lze následně ve skriptu dál operovat.

Jsou to automaticky dostupné proměnné `$0`, `$1`, `$2`, .. `$9`. Přičemž proměnná `$0` obsahuje název spuštěného skriptu.