Darktea Home

1. 变量

- 变量的赋值用 等号, 但是等号间不能有空格

```
 foo=42  # sets foo to 42
```

- 引用变量的值, 用 $ 符号

```
echo $foo    # prints 42
```

2. 数组

- 给数组赋值

```
  foo[0]="one"
  foo[1]="two"
  echo ${foo[1]}  # prints "two"
```

- 换种方式给数组赋值

```
foo=("a a a" "b b b" "c c c")
echo ${foo[2]}  # prints "c c c"
echo $foo       # prints "a a a"
```

- 整个数组的赋值 (注意空格时需要用引号)

```
 foo=("a 1" "b 2" "c 3")
 bar=(${foo[@]})
 baz=("${foo[@]}")
 echo ${bar[1]}            # oops, print "1"
 echo ${baz[1]}            # prints "b 2"
```

3. 专用变量

```
 echo $0      # 脚本本身的名字

 echo $1      # 传给脚本的第一个参数
 echo $2      # 第2个参数
 echo $9      # 第9个参数
 echo $10     # 第一个参数, 后面跟一个 0
 echo ${10}   # 第10个参数

 echo $#      # 传给脚本的参数总数

 echo $?   # prints 0 代表前一个进程退出时返回0, 执行成功

 echo $?   # 非 0; 代表前一个进程退出时返回非0, 执行失败

 echo  $$  # 当前 shell 的 进程id
```

- $! 最近运行的背景进程的进程 id

```
# sort two files in parallel:
 sort words > sorted-words &        # launch background process
 p1=$!
 sort -n numbers > sorted-numbers & # launch background process
 p2=$!
 wait $p1
 wait $p2
 echo Both files have been sorted.
```

4. 字符串处理

- 替换

```
 foo="I'm a cat."
 echo ${foo/cat/dog}  # prints "I'm a dog."
 echo $foo                  # still prints "I'm a cat."
```

- 替换一次和多次

```
 foo="I'm a cat, and she's cat."
 echo ${foo/cat/dog}   # prints "I'm a dog, and she's a cat."
 echo ${foo//cat/dog}  # prints "I'm a dog, and she's a dog."
```

- 删除

```
 foo="I like meatballs."
 echo ${foo/balls}       # prints I like meat.
```

5. 数组长度

```
ARRAY=(abcdd b c)
echo ${#ARRAY}          # prints 5 错误

echo ${#ARRAY[@]}     # prints 3 正确
```

6. 引号

```
world=Earth
foo='Hello, $world!'
bar="Hello, $world!"
echo $foo            # 单引号, prints Hello, $world!
echo $bar            # 双引号, prints Hello, Earth!
```

References:

- Shell programming with bash: by example, by counter-example

darktea

tiantianding@Gmail.com

github.com/darktea