

Shell 传递参数

主讲人：杨老师

我们可以在执行 Shell 脚本时，向脚本传递参数，脚本内获取参数的格式为：\$n。n 代表一个数字，1 为执行脚本的第一个参数，2 为执行脚本的第二个参数，以此类推.....

实例：以下实例我们向脚本传递三个参数，并分别输出，其中 \$0 为执行的文件名：

```
#!/bin/bash
# author:dajiangtai
# url:http://www.dajiangtai.com/
echo "Shell 传递参数实例！";
echo "脚本名称：$0";
echo "第一个参数为：$1";
echo "第二个参数为：$2";
echo "第三个参数为：$3";
```

为脚本设置可执行权限，并执行脚本，输出结果如下所示：

```
$ chmod +x testParameter.sh
$ ./testParameter.sh 1 2 3
```

Shell 参数传递

另外，还有几个特殊字符用来处理参数：

参数处理	说明
<code>\$#</code>	传递到脚本的参数个数
<code>\$*</code>	以一个单字符串显示所有向脚本传递的参数。 如" <code>\$*</code> "用「 <code>"</code> 」括起来的情况、以" <code>\$1 \$2 ... \$n</code> "的形式输出所有参数。
<code>\$\$</code>	脚本运行的当前进程ID号
<code>\$_</code>	后台运行的最后一个进程的ID号
<code>\$@</code>	与 <code>\$*</code> 相同，但是使用时加引号，并在引号中返回每个参数。 如" <code>\$@</code> "用「 <code>"</code> 」括起来的情况、以" <code>\$1" "\$2" ... "\$n</code> " 的形式输出所有参数。
<code>\$-</code>	显示Shell使用的当前选项，与 <code>set</code> 命令功能相同。
<code>\$_</code>	显示最后命令的退出状态。0表示没有错误，其他任何值表明有错误。

\$* 与 \$@ 区别：

相同点：都是引用所有参数。

不同点：只有在双引号中体现出来。

假设在脚本运行时写了三个参数 1、2、3，，则 " *" 等价于 "1 2 3"（传递了一个参数），而 "@" 等价于 "1" "2" "3"（传递了三个参数）。

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

```
echo "-- \${*} 演示 ---"
```

```
for i in "${*}"; do
```

```
    echo $i
```

```
done
```

```
echo "-- \${@} 演示 ---"
```

```
for i in "${@}"; do
```

```
    echo $i
```

```
done
```



THANKS

黄丽老师 : 3354223855

小夏老师 : 972628726