# [简单的shell脚本练习实例](https://zhangge.net/4023.html)

此文将整理一些简单实用、有趣易学的shell练习脚本，希望可以帮到一些想学习shell的初学者。张戈也会在闲暇时间不定期更新此文，对已有实例进行改进和注释，增强可读性和易懂性。当然我也会不定期的添加更多简单、有趣且实用的shell练习脚本，帮助一些有需要的入门学习者。

### 一、编写一个脚本使我们在写一个脚本时自动生成”#!/bin/bash”这一行和注释信息。

原文代码为：

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | #!/bin/bash  if  ! grep "^#!" $1 &>/dev/null; then  cat >> $1 << EOF  #!/bin/bash  # Author:  #Date & Time: `date +"%F %T"`  #Description:  EOF  fi  vim +5 $1 |

初学者看到这代码，可能不太会用，其实很简单，看到有$1，就表示需要带参数来执行，所以，这个脚本的执行方法是：

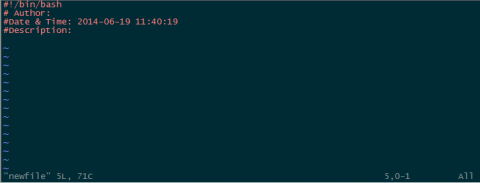
①、将以上代码保存为shell脚本，比如test，

②、使用chmod加上执行权限，chmod +x test

③、执行 ./test  newfile 即可看到效果。

Ps：当然也可以不要第②步，直接使用 sh test newfile 即可。

执行效果如下：

[](https://res.zgboke.com/wp-content/uploads/2014/06/script1.png)

这个脚本对于经常写shell的童鞋就很有用，但是忘记带参数执行就会卡住不动，而且不带路径的话就直接在当前目录下生成新文件，会很乱。所以张戈就将其改进一下，变得更加易用：

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | #!/bin/bash  if [[ -z "$1" ]];then     newfile=~/newscript\_`date +%m%d\_%S`  else     newfile=$1  fi  if  ! grep "^#!" $newfile &>/dev/null; then  cat >> $newfile << EOF  #!/bin/bash  # Author: Inert Your Name here.  #Date & Time: `date +"%F %T"`  #Description: Please Edit here.  EOF  fi  vim +5 $newfile |

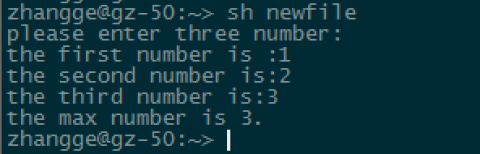
改进说明：如果未带参数执行，将在家目录下生成带时间戳的新文件，避免重复及乱的问题。可将这个脚本改名后直接丢到path路径中，比如/bin/addjb  那么你的系统就多了一个命令 addjb了，是不是很有趣呢！

时间有限，以下脚本暂时就不做测试、解释或改进了，请先自行测试吧！有空再来更新。

### 二、任意三个整数，判断最大数。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | #!/bin/bash  echo  "please enter three number:"    read -p "the first number is :" n1    read -p "the second number is:" n2    read -p "the third number is:" n3  let MAX=$n1  if [ $n2 -ge $n1 ]; then      MAX=$n2  fi  if [ $n3 -ge $MAX ]; then       MAX=$n3  fi  echo "the max number is $MAX." |

执行效果：

[](https://res.zgboke.com/wp-content/uploads/2014/06/script2.png)

注：非常简单的逻辑判断脚本，有兴趣的可以改进下，练练手。

### 三、求100以内偶数的和。

方法①：

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | #!/bin/bash  sum=0  for I in {1..50}; do      sum=$(($sum+2\*$I))  done  echo "the sum is $sum" |

方法②：

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | #!/bin/bash  let SUM=0  for I in $(seq 1 100); do     if [ $[$I%2] == 0 ]; then           let  SUM+=$I     fi  done  echo "the sum is $SUM." |

### 四、利用for语句ping C类网、ping B类网。

①、ping C类网：

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | #!/bin/bash  read -p "C NETWORK:" MYNET  PINGNET=`echo $MYNET | sed 's/\([0-9.]\*\)\ .[0-9]\*/\1/g'`  let I=1  while [ $I -le 254 ];do    ping –c1 –W1 $PINGNET.$I &>/dev/null     [ $? -eq 0 ] && echo "$PINGNET.$I  online." || echo "$PINGNET.$I offline."  let  I++  done |

②、ping B类网：

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | #!/bin/bash  read -p "B network:" MYNET  PINGNET=`echo $MYNET | sed 's/\([0-9]\{1,3\}\.[0-9]\{1,3\}\)\..\*/\1/g'`  for P in {0..255}; do        for I in {1..255}; do    if ping -c1 -W2 $PINGNET.$P.$I &>/dev/null; then      echo  "$PINGNET.$P.$I is online."        else          echo "$PINGNET.$P.$I is offline."    fi      done  done |

### 五、提示输入一个用户名，判断用户是否存在，如果存在，显示一下用户默认的shell。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | #!bin/bash  read –p "please input a username:"  USER  if cut –d：-f1  /etc/passwd | grep "^$USER$" &> /dev/null ;then  MYBASH=`grep  "^$USER:"  /etc/passwd | cut –d :  -f7`  echo "${USER}'s shell is $MYBASH"  else    echo "$USER not exists."    exit  4  fi |

### 六、监控系统登录人数，超过四个，显示已经达到四个，5S检查一下，并退出脚本（exit）

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | #! /bin/bash  read –p "A user:"  MYUSER  cut –d : -f1 /etc/passwd | grep "^$MYUSER" &> /dev/null || exit 6  let COUNT=`who |  grep "^$MYUSER"  | wc –l`  until [ $COUNT –ge 4 ]; do  sleep 5  let COUNT=`who | grep “^$MYUSER” |  wc -l`  done  echo "$MYUSER loged 4 times." |