

1. 编写脚本，通过 3 个 read 命令读取用户输入的三个任意数字，脚本对输入的三个数字求和输出。

```
#vim test.sh
#!/bin/bash
read -p "请输入数字" num1
read -p "请输入数字" num2
read -p "请输入数字" num3
echo ${num1+num2+num3}
```

2. 判断当前系统启动的进程数量，如果进程数量超过 100 个，则发送邮件给 root 报警。

(ps 命令可以查看所有进程，wc 可以统计数量)

```
#vim test.sh
#!/bin/bash
num=$(ps aux)
[ $num -gt 100 ] && echo too many procs | mail -s Warning root
```

3. 编写脚本，通过 2 个 read 命令，2 次读取用户输入的密码，脚本判断 2 次密码是否一致，一致则返回 OK，否则返回 Error。

```
#vim test.sh
#!/bin/bash
read -p "请输入密码" pass1
read -p "请再输入密码" pass2
if [ $pass1 == $pass2 ];then
    echo OK
else
    echo Error
fi
```

4. 编写脚本，测试当前用户对/etc/passwd 文件是否具有读、写、执行的权限，脚本执行结果类似下面的效果。

当前用户对/etc/passwd 有读权限
当前用户对/etc/passwd 有写权限
当前用户对/etc/passwd 无执行权限

```
#vim test.sh
#!/bin/bash
file=/etc/passwd
[ -r /etc/passwd ] && echo "当前用户对$file 有读权限" || echo "当前用户对$file 没有读权限"
[ -w /etc/passwd ] && echo "当前用户对$file 有写权限" || echo "当前用户对$file 没有写权限"
[ -x /etc/passwd ] && echo "当前用户对$file 有执行权限" || echo "当前用户对$file 没有执行权限"
```

5. 编写脚本，判断系统中是否启动了 sshd、httpd、crond、vsftpd 服务。

提示：ps 命令把所有进程列出后，使用 grep 过滤，过滤到表示启动，未过滤到表示未启动
执行结果如下：

sshd 已经启动
httpd 未启动
crond 已经启动
vsftpd 未启动

```
# vim test.sh
ps aux | grep -v grep | grep sshd
[ $? -eq 0 ] && echo "sshd 已经启动" || echo "sshd 未启动"
ps aux | grep -v grep | grep httpd
[ $? -eq 0 ] && echo "httpd 已经启动" || echo "httpd 未启动"
ps aux | grep -v grep | grep crond
[ $? -eq 0 ] && echo "crond 已经启动" || echo "crond 未启动"
ps aux | grep -v grep | grep vsftpd
[ $? -eq 0 ] && echo "vsftpd 已经启动" || echo "vsftpd 未启动"
```

6. 编写脚本，通过 read 读取用户输入的一个圆半径，脚本输出圆的面积，面积公式如下：

圆面积= πr^2 ，提示 r^2 就是 r 乘 r， π 是 3.14

```
# vim test.sh
#!/bin/bash
read -p "请输入半径:" r
s=$(echo "scale=2;3.14*$r*$r" | bc)
echo "圆面积为:$s"
```

7. 编写脚本，通过 read 读取用户任意输入的用户名，脚本判断系统中是否有该账户

```
# vim test.sh
#!/bin/bash
read -p "请输入用户名:" user
id $user &>/dev/null
[ $? -eq 0 ] && echo "该用户已存在" || echo "该用户不存在"
```

8. 编写脚本，通过 find 找出系统中大于 1G 的文件有数个，如果大于 1G 的文件数量少于 10 个，脚本仅输出

文件个数即可，如果大于 1G 的文件大于等于 10 个则显示具体的文件名。

```
#vim test.sh
num=$(find / -size +100M 2>/dev/null | wc -l)
if [ $num -lt 10 ];then
    echo "大于 1G 的文件有$num 个"
else
    find / -size +100M 2>/dev/null
fi
```