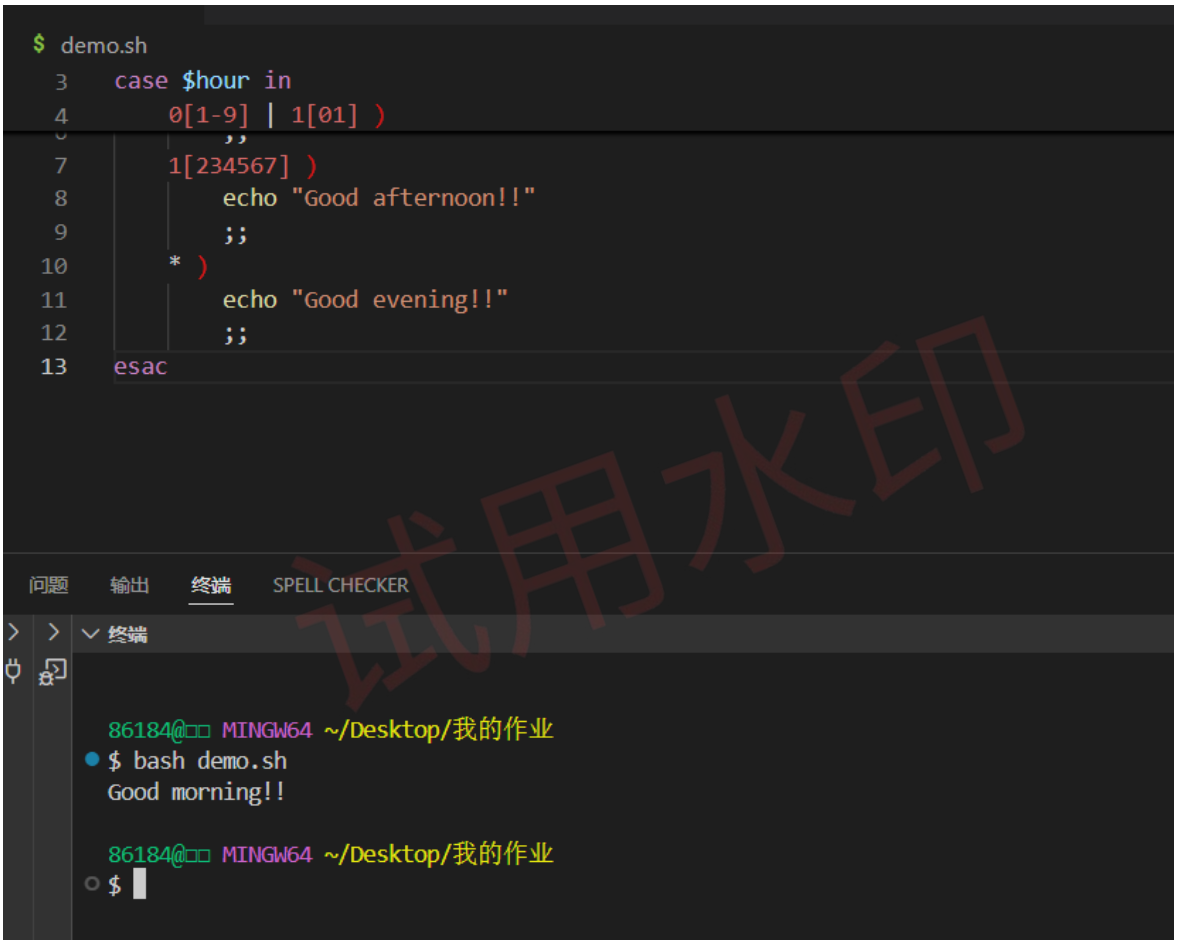
Experiment 2

use a editor to finishe the following shell scripts, and run them in Linux system.

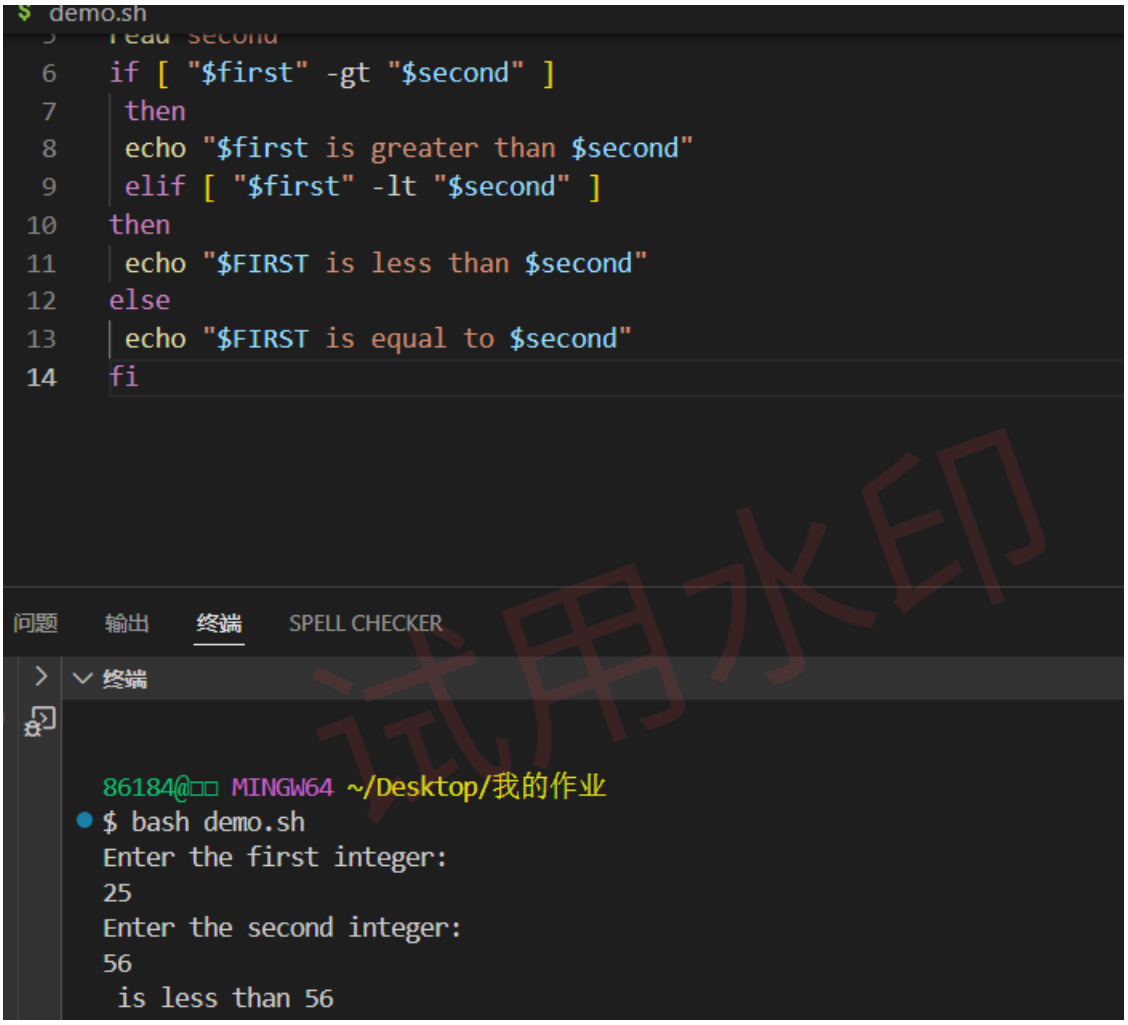
1. Obtain the system time, and check whether it is in the morning, afternoon, or evening.

```
#!/bin/bash
hour = `date +%H`
 case $hour in
 0[1-9] | 1[01] )
 echo "Good morining !!"
 ;;
1[234567] )
 echo "Good afternoon !!"
 ;;
 * )
 echo "Good evening !! "
 ;;
Esac
```



2. Input two number, check which one is greater, and output the result.

```
#!/bin/sh
echo "Enter the first integer:"
read first
echo "Enter the second integer:"
read second
if [ "$first" -gt "$second" ]
 then
 echo "$first is greater than $second"
 elif [ "$first" -lt "$second" ]
then
 echo "$FIRST is less than $second"
else
 echo "$FIRST is equal to $second"
fi
```



3. Find the minimal value in a given list.

```
#!/bin/bash
smallest=10000
```

```
for i in 8 2 18 0 -3 87
do
if test $i -lt $smallest
then
 smallest=$i
fi
done
echo $smallest
```

```
$ demo.sh
2  smallest=10000
3  for i in 8 2 18 0 -3 87
4  do
5  if test $i -lt $smallest
6  then
7   smallest=$i
8  fi
9  done
10 echo $smallest
```

```
问题  输出  终端  SPELL CHECKER
终端

86184@□□ MINGW64 ~/Desktop/我的作业
$ bash demo.sh
-3

86184@□□ MINGW64 ~/Desktop/我的作业
$
```

5. Check whether a given number is a prime, you have to write a function, and call the function.

```
prime( )
{
flag=1
j=2
while [ $j -le `expr $1 / 2` ]
do
if [ `expr $1 % $j` -eq 0 ]
then
flag=0
break
fi
j=`expr $j + 1`
done
if [ $flag -eq 1 ]
then
return 1
else
return 0
fi
}
prime $1
```

```
$ demo.sh
2  {
20 }
21 prime $1
22 if [ $? -eq 1 ]
23  then
24  echo "$1 is a prime!"
25  else
26  echo "$1 is not a prime!"
27 fi
```

```
问题  输出  终端  SPELL CHECKER
终端

86184@□□ MINGW64 ~/Desktop/我的作业
$ bash demo.sh
expr: syntax error: unexpected argument ‘2’
demo.sh: line 5: [: 2: unary operator expected
 is a prime!

86184@□□ MINGW64 ~/Desktop/我的作业
$
```