

命令行环境和 Python 实验报告

姓名：刘昱莹 学号：22010022040

2024 年 9 月 12 日

目录

1 练习内容	2
1.1 命令行环境	2
1.1.1 暂停和后台执行进程	2
1.1.2 终端多路复用	2
1.1.3 别名	2
1.1.4 SSH 命令的使用	2
1.1.5 scp 命令的使用	2
1.2 Python 入门基础	2
1.2.1 基本数据类型	2
1.2.2 基本语法	2
1.3 Python 视觉应用	2
1.3.1 PIL: Python 图像处理类库	2
1.3.2 Matplotlib	2
1.3.3 Numpy	2
2 结果	2
2.1 命令行环境	2
2.1.1 暂停和后台执行进程	2
2.1.2 终端多路复用	3
2.1.3 别名	3
2.1.4 SSH 命令的使用	4
2.1.5 scp 命令的使用	4
2.2 Python 入门基础	5
2.2.1 基本数据类型	5
2.2.2 基本语法	8
2.3 Python 视觉应用	10
2.3.1 PIL: Python 图像处理类库	10
2.3.2 Matplotlib	10
2.3.3 Numpy	11
3 解题感悟	11
4 Github 链接	11

1 练习内容

1.1 命令行环境

1.1.1 暂停和后台执行进程

学习并练习了 nohup、bg、kill -STOP、kill -SIGHUP 指令的使用方法。

1.1.2 终端多路复用

使用 tmux 命令新建了 session1 对话，并分别新建了两个窗口。

1.1.3 别名

练习了创建别名、获取别名定义以及禁用别名的练习。

1.1.4 SSH 命令的使用

使用 SSH 连接 Windows 系统和虚拟机。

1.1.5 scp 命令的使用

使用 scp 命令进行文件的复制。

1.2 Python 入门基础

1.2.1 基本数据类型

学习了 Python3 中常见的数据类型：Number（数字）String（字符串）bool（布尔类型）List（列表）Tuple（元组）Set（集合）Dictionary（字典）。

1.2.2 基本语法

学习了 Python 中条件语句、循环语句、输入输出等基本语法。

1.3 Python 视觉应用

1.3.1 PIL: Python 图像处理类库

1.3.2 Matplotlib

1.3.3 Numpy

2 结果

2.1 命令行环境

2.1.1 暂停和后台执行进程

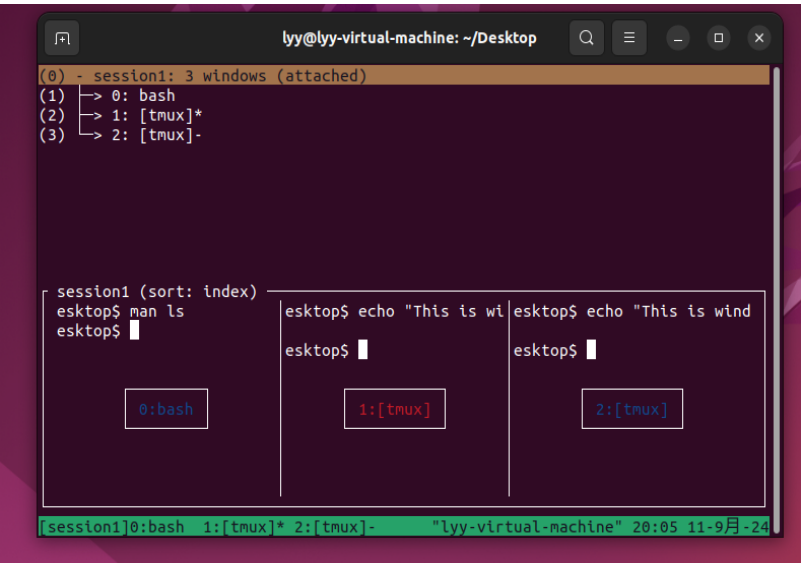
```
lly@lly-virtual-machine:~/Desktop$ sleep 1000
^Z
[1]+  Stopped                  sleep 1000
lly@lly-virtual-machine:~/Desktop$ nohup sleep 2000 &
[2] 2541
lly@lly-virtual-machine:~/Desktop$ nohup: ignoring input and appending output to 'nohup.out'
jobs
[1]+  Stopped                  sleep 1000
[2]-  Running                  nohup sleep 2000 &
lly@lly-virtual-machine:~/Desktop$ bg %1
[1]+ sleep 1000 &
lly@lly-virtual-machine:~/Desktop$ jobs
[1]-  Running                  sleep 1000 &
[2]+  Running                  nohup sleep 2000 &
lly@lly-virtual-machine:~/Desktop$ kill -STOP %1
lly@lly-virtual-machine:~/Desktop$ jobs
[1]+  Stopped                  sleep 1000
[2]-  Running                  nohup sleep 2000 &
lly@lly-virtual-machine:~/Desktop$ kill -SIGHUP %1
[1]+  Stopped                  sleep 1000
lly@lly-virtual-machine:~/Desktop$ jobs
[1]+  Hangup                   sleep 1000
[2]-  Running                  nohup sleep 2000 &
lly@lly-virtual-machine:~/Desktop$ kill -SIGHUP %2
lly@lly-virtual-machine:~/Desktop$ jobs
[2]+  Running                  nohup sleep 2000 &
lly@lly-virtual-machine:~/Desktop$ kill %2
lly@lly-virtual-machine:~/Desktop$ jobs
[2]+  Terminated              nohup sleep 2000
```

图 1: 进程命令练习

2.1.2 终端多路复用

```
lly@lly-virtual-machine:~/Desktop$ tmux new -s session1
[detached (from session session1)]
lly@lly-virtual-machine:~/Desktop$ tmux ls
session1: 3 windows (created Wed Sep 11 20:01:08 2024)
lly@lly-virtual-machine:~/Desktop$
```

(a)



(b)

图 2: tmux 命令

2.1.3 别名

```

lly@lly-virtual-machine:~/Desktop$ alias ll="ls -lh"
lly@lly-virtual-machine:~/Desktop$ ll
total 60K
drwxrwxr-x 2 lly lly 4.0K 8月 30 11:28 bar
-rwxrwxr-x 1 lly lly 212 9月 2 16:44 buggy.sh
-rw-rw-r-- 1 lly lly 758 6月 22 15:29 chacha_code.py
-rw-rw-r-- 1 lly lly 768 6月 22 14:46 code1.py
-rw-rw-r-- 1 lly lly 2.3K 6月 22 15:25 code.py
-rwxrwxr-x 1 lly lly 230 9月 2 16:41 debug.sh
drwxrwxr-x 2 lly lly 4.0K 8月 30 11:27 foo
-rw-rw-r-- 1 lly lly 18 9月 11 17:20 hello.txt
drwxrwxr-x 3 lly lly 4.0K 9月 2 19:40 html_root
drwxrwxr-x 3 lly lly 4.0K 8月 23 11:31 learnGit
drwxrwxr-x 3 lly lly 4.0K 8月 30 09:09 Learn-Git-and-LaTeX
-rw-rw-r-- 1 lly lly 283 8月 30 10:55 marco.sh
-rw-rw-r-- 1 lly lly 0 9月 6 10:10 nohup.out
-rw-rw-r-- 1 lly lly 634 9月 2 16:49 out.log
-rw-rw-r-- 1 lly lly 1.7K 6月 22 15:57 rsa_code.py
-rwxrwxr-x 1 lly lly 356 9月 6 10:08 test.sh
lly@lly-virtual-machine:~/Desktop$ alias ll
alias ll='ls -lh'
lly@lly-virtual-machine:~/Desktop$ unalias ll
lly@lly-virtual-machine:~/Desktop$ ll
ll: command not found
lly@lly-virtual-machine:~/Desktop$

```

图 3: 新建 ll 为 ls -lh 命令的别名

2.1.4 SSH 命令的使用

```

C:\Users\hx>ssh lly@192.168.141.130
lly@192.168.141.130's password:
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.8.0-40-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

31 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

7 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

lly@lly-virtual-machine:~$

```

图 4: SSH 命令连接 Windows 和 Linux

2.1.5 scp 命令的使用

```

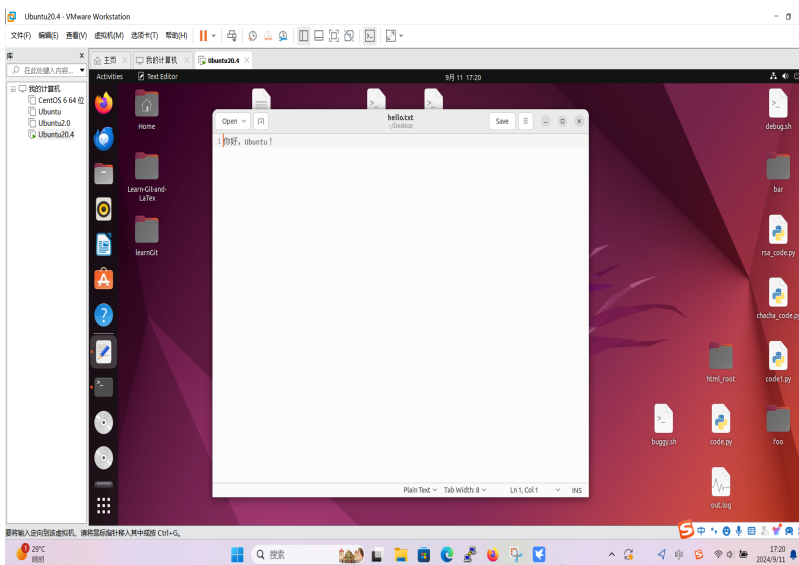
lly@lly-virtual-machine:~/Desktop$ scp ~/Desktop/test.sh hx@192.168.3.8:"C:\User
s\hx\Desktop"
hx@192.168.3.8's password:
test.sh                                100% 356 199.2KB/s 00:00
lly@lly-virtual-machine:~/Desktop$

```

图 5: Linux 向 Windows 进行文件传输的结果

```
C:\Users\hx> scp "C:\Users\hx\Desktop\hello.txt" lyy@192.168.141.130:~/Desktop
lyy@192.168.141.130's password:
hello.txt
100% 18 7.4kB/s 00:00
C:\Users\hx>
```

(a)



(b)

图 6: Windows 向 Linux 进行文件传输的结果

2.2 Python 入门基础

2.2.1 基本数据类型

String (字符串)

```
str.split(" ")
[3] ✓ 0.0s
... ['Hello', 'World!']

str.split(" ")[-1::-1]
[4] ✓ 0.0s
... ['World!', 'Hello']

''.join(str.split(" ")[-1::-1])
[5] ✓ 0.0s
... 'World! Hello'
```

图 7: string 逆向读取

bool (布尔类型)

```
[4] type(b)
✓ 0.0s
... bool

[5] 5 > 6
✓ 0.0s
... False
```

图 8: bool

List (列表)

```
[13] list[1:]
✓ 0.0s
... [786, 2.23, 'runoob', 70.2]

[14] list[1:3]=[12, '1112']
✓ 0.0s

[15] list
✓ 0.0s
... ['abcd', 12, '1112', 'runoob', 70.2]

[16] list[0:-1:2]
✓ 0.0s
... ['abcd', '1112']

[17] list[-1::-1]
✓ 0.0s
... [70.2, 'runoob', '1112', 12, 'abcd']
```

图 9: list

bytes (不可变的二进制序列)

```
[18] x = b"hello"
✓ 0.0s

[19] x
✓ 0.0s
... b'hello'

[20] x[1:3]
✓ 0.0s
... b'el'

[21] x[0] == ord("h")
✓ 0.0s
... True

[22] x[0] == "h"
✓ 0.0s
... False
```

图 10: bytes

Tuple (元组)

```
[6] tup1=(20,)
✓ 0.0s

[7] tup2=(20)
✓ 0.0s

[10] type(tup1)
✓ 0.0s
... tuple

[11] type(tup2)
✓ 0.0s
... int

[12] type(())
✓ 0.0s
... tuple
```

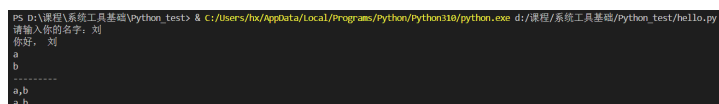
图 11: tuple

2.2.2 基本语法

1、输入输出的使用

Listing 1: input() 和 print()

```
1 name=str(input("请输入你的名字："))
2 print("你好，", name)
3
4 x="a"
5 y="b"
6 # 换行输出
7 print( x )
8 print( y )
9
10 print('-----')
11 # 不换行输出
12 print( x, end=",", )
13 print(y)
14 print(x,y)
```



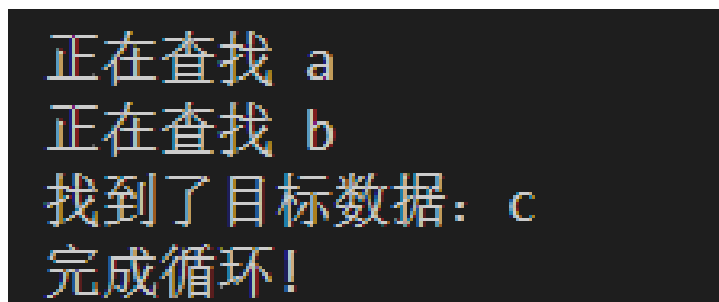
```
PS D:\课程\系统工具基础\python_test> & c:/Users/hz/AppData/Local/Programs/Python/Python310/python.exe d:/课程/系统工具基础/python_test/hello.py
请输入你的名字：刘
你好， 刘
a
b
-----
a,b
a b
```

图 12: 输入输出

2、循环语句中有关 break 和 else 的练习

Listing 2: for()

```
1 sites = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k']
2 for site in sites:
3     if site == "c":
4         print("找到了目标数据：" + site)
5         break
6     print("正在查找 " + site)
7 else:
8     print("没有找到目标数据!")
9 print("完成循环!")
```



```
正在查找 a
正在查找 b
找到了目标数据: c
完成循环!
```

图 13: for 循环

3、str.format() 函数的使用

```
print('站点列表 {0}, {1}, 和 {other}'.format('Google', 'Runoob', other='Taobao'))
[2] ✓ 0.0s
... 站点列表 Google, Runoob, 和 Taobao。

print('站点列表 {1}, {0}, 和 {other}'.format('Google', 'Runoob', other='Taobao'))
[3] ✓ 0.0s
... 站点列表 Runoob, Google, 和 Taobao。
```

图 14: str.format()

4、有关函数参数的练习

Listing 3: 参数的调用

```
1 def printinfo( name, age ):
2     "打印任何传入的字符串"
3     age=20
4     print ("名字: ", name)
5     print ("年龄: ", age)
6     return
7
8 #调用 printinfo 函数
9 printinfo( age=50, name="runoob" )
```

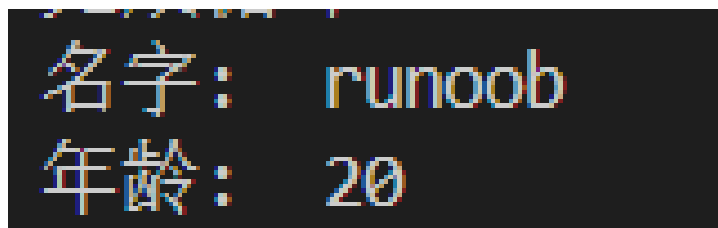


图 15: 函数调用输出结果

2.3 Python 视觉应用

2.3.1 PIL: Python 图像处理类库

Listing 4: PIL

```

1  from PIL import Image
2  import os
3
4  im=Image.open('picture.png')
5
6  #灰度化
7  gray=im.convert('L')
8  gray.show()
9
10 #裁剪
11 box=(100,100,400,400)
12 region=im.crop(box)
13 region.show()
14
15 #复制粘贴
16 region=region.transpose(Image.ROTATE_180)
17 region.show()
18 im.paste(region,box)
19 im.show()
20
21 #调整尺寸和旋转
22 out=im.resize((200,200))
23 out=out.rotate(45)
24 out.show()
25
26 im.thumbnail((100,100))
27 im.show()

```

2.3.2 Matplotlib

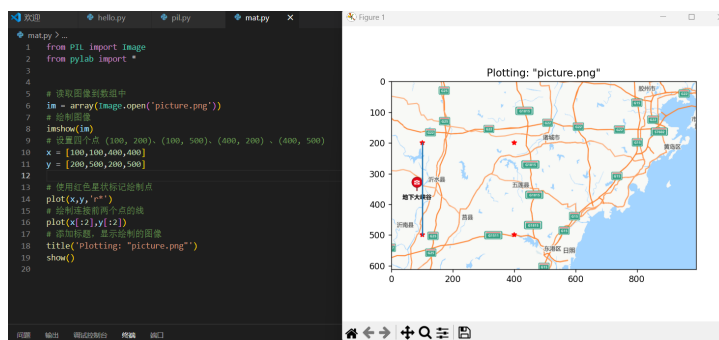


图 16: 绘制结果

2.3.3 Numpy

Listing 5: 图像反转

```
1 from PIL import Image
2 from pylab import *
3 from numpy import array
4
5 im_gray = Image.open('picture.png').convert('L') # 转换为灰度
6 im2 = 255 - array(im_gray) # 图像反转
7 imshow(im2, cmap='gray')
8 title('Reverse Image')
9 show()
```

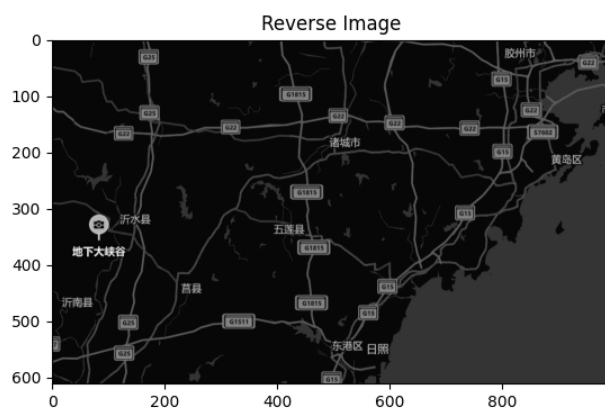


图 17: 图像反转

3 解题感悟

通过本次实验学会了一些有关命令行的指令，在学习 SSH 指令时，我对不同服务器之间的关系和连接方式有了较为初步的认识。通过自主学习，我了解了 Python 的基本数据类型和基本语句以及其中部分库函数的使用方法，我还了解到了 Python 中一些处理图像的基本工具。在之前的学习中某些内容我已有所涉及，但通过本次的系统学习，我对这些技能的印象更加深刻，便于我在以后的学习中灵活运用以处理一些相关问题。

4 Github 链接

<https://github.com/yuying019828/Python>