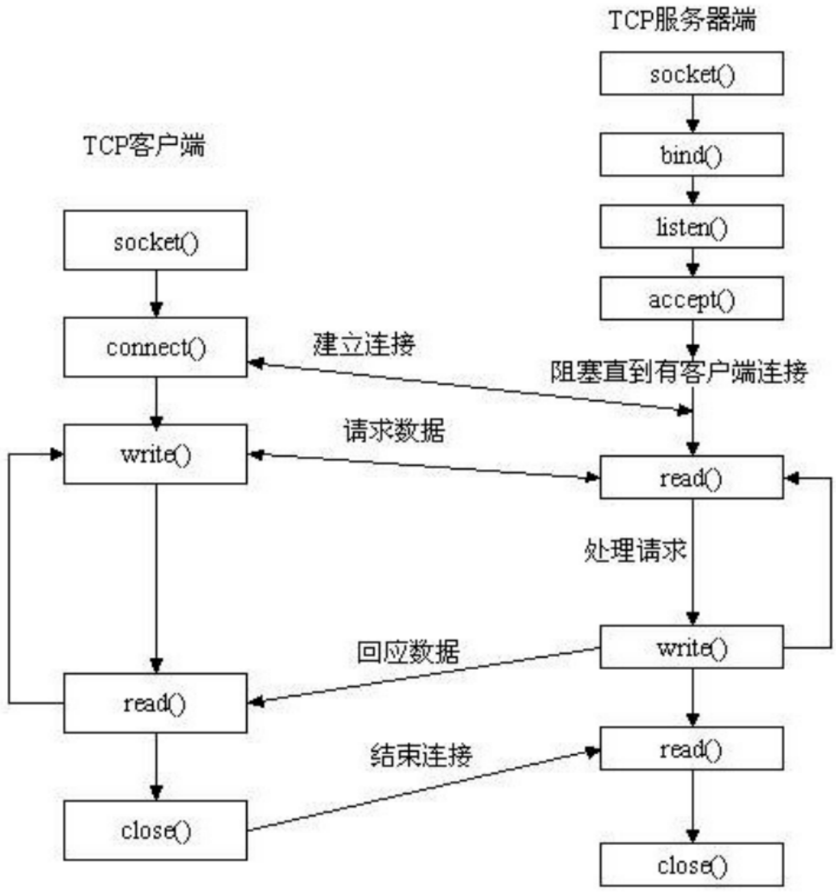
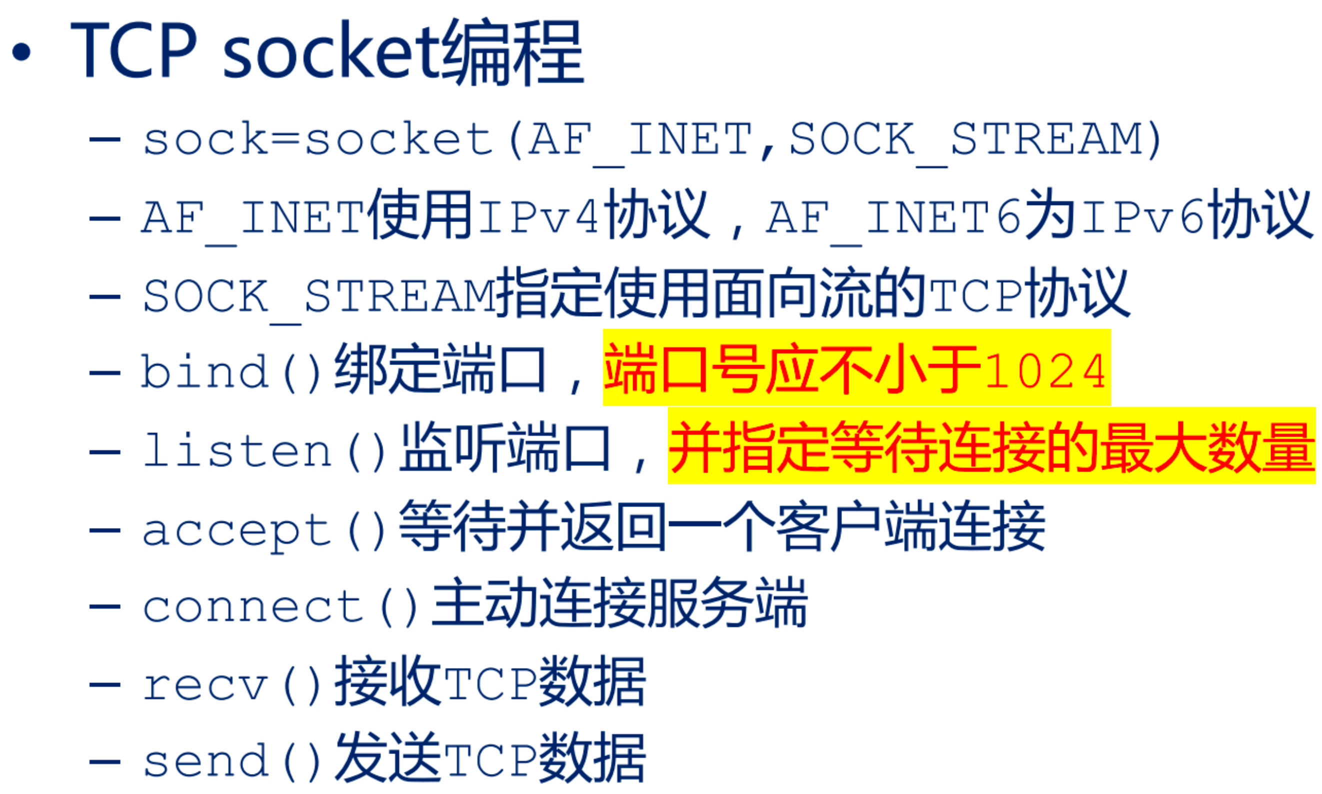
网络编程

只需理解编程逻辑

- TCP socket编程
  - c/s结构

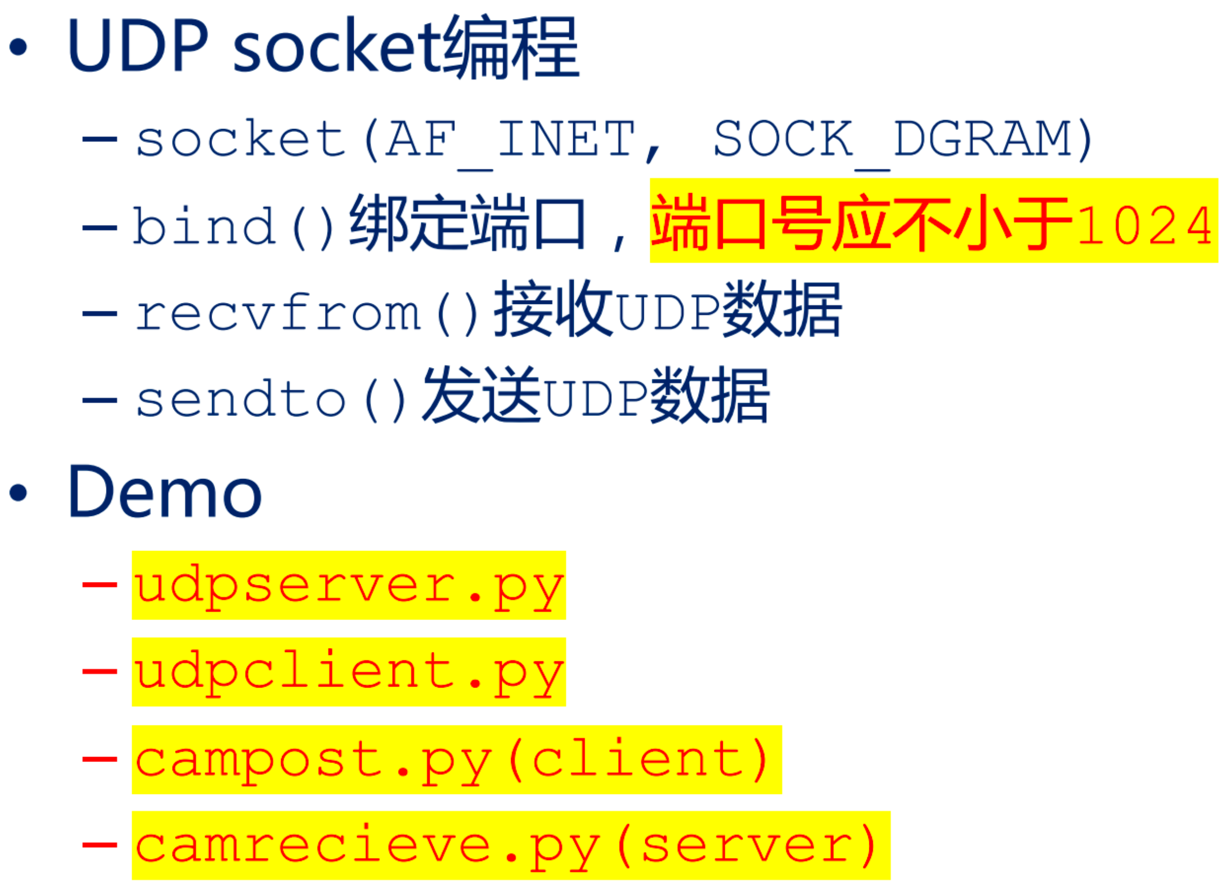




- TCP socket编程
  - sock=socket(AF_INET,SOCK_STREAM)
  - AF_INET使用IPv4协议，AF_INET6为IPv6协议
  - SOCK_STREAM指定使用面向流的TCP协议
  - bind()绑定端口，端口号应不小于1024
  - listen()监听端口，并指定等待连接的最大数量
  - accept()等待并返回一个客户端连接
  - connect()主动连接服务端
  - recv()接收TCP数据
  - send()发送TCP数据

```
import sys
import os
from socket import *
import time
import random

HOST='127.0.0.1'

RSIZE=1024

class TCPServer:#非多线程，只能响应一个client

  def __init__(self,port,maxconnections=5):
    self._port=port
    self._maxconnections=maxconnections
    self._server=socket(AF_INET, SOCK_STREAM)

  def start(self):
    self._server.setsockopt(SOL_SOCKET,SO_REUSEADDR,1)#端口释放后马上可以被重新使用
    self._server.bind((HOST,self._port))
    self._server.listen(self._maxconnections)
    print("SERVER is listening on %s" % self._port)
    while True:#如果有多个client同是发起连接，只响应一个，其他会等待
      conn,addr=self._server.accept()#block
      print(f"client's connection: {conn}, its address:{addr}")
      while True:
        try:
          data=conn.recv(RSIZE)
          if not data:
            break
          print("CLIENT: %s" % data.decode('utf-8'))
          if data.decode('utf-8')=='bye':
            conn.send("再见!".encode('utf-8'))
            break
          else:
            conn.send('收到!'.encode('utf-8'))
        except Exception as e:
          print("SERVER ERROR: %s" % e)
          break
      conn.close()
    self._server.close()

def main():
  ser=TCPServer(int(sys.argv[1]))
  ser.start()

if __name__=='__main__':
  main()
```

```
import os
import sys
import time
import random
from socket import *

class TCPClient:

  def __init__(self,server_ip,server_port):
    self._server_ip=server_ip
    self._server_port=server_port
    self._client=socket(AF_INET,SOCK_STREAM)

  def start(self):
    self._client.connect((self._server_ip,self._server_port))
    while True:
      msg=input("CLIENT： ")
      self._client.send(msg.encode('utf-8'))
      data=self._client.recv(1024)
      if not data:
        continue
      if(data.decode('utf-8')=='再见!'):
        print("结束连接")
        break
      else:
        print("SERVER: %s" % data.decode('utf-8'))
    self._client.close()

def main():
  client=TCPClient(sys.argv[1],int(sys.argv[2]))
  client.start()

if __name__=='__main__':
  main()
```



- UDP socket编程
  - socket(AF_INET, SOCK_DGRAM)
  - bind()绑定端口，端口号应不小于1024
  - recvfrom()接收UDP数据
  - sendto()发送UDP数据
- Demo
  - udpserver.py
  - udpclient.py
  - campost.py(client)
  - camrecieve.py(server)