慧科
只为赋能人才

模块、线程、网络

——— 张洪涛

慧科
只为赋能人才

python多文件

- 当编写大工程时，需要对源码进行分文件，方便管理和代码阅读。
- 如果要调用其他文件中定义好的函数、使用其他文件中定义的类等，需要了解：类，模块，包的概念。

慧科
只为赋能人才

模块

- 在Python可理解为对应于一个文件。在创建了一个脚本文件后，定义了某些函数和变量。你在其他需要这些功能的文件中，导入这模块，就可重用这些函数和变量。

- 包：通常包总是一个目录，可以使用import导入包，或者from + import来导入包中的部分模块。包目录下为首的一个文件便是 __init__.py。

慧科
只为赋能人才

模块的搜索方式

- 当导入一个模块时，解释器先在当前包中查找模块，若找不到，然后在内置的built-in模块中查找，找不到则按sys.path给定的路径找对应的模块文件(模块名.py)

慧科
只为赋能人才

多文件示例

- 调用自己定义的，其他文件中的函数

慧科
只为赋能人才

if __name__ == '__main__'

- if __name__ == '__main__'的意思是：当.py文件被直接运行时，if __name__ == '__main__'之下的代码块将被运行；当.py文件以模块形式被导入时，if __name__ == '__main__'之下的代码块不被运行。

慧科
只为赋能人才

示例

- 验证if __name__ == '__main__'

慧科
只为赋能人才

python线程

- 线程：并发执行，同时实现多任务

- import thread
- thread.start_new_thread (function, args[, kwargs])

示例

```
#!/usr/bin/python
# -*- coding: UTF-8 -*-

import thread
import time

# 为线程定义一个函数
def print_time( threadName, delay):
   count = 0
   while count < 5:
      time.sleep(delay)
      count += 1
      print "%s: %s" % ( threadName, time.ctime(time.time()) )

# 创建两个线程
try:
   thread.start_new_thread( print_time, ("Thread-1", 2, ) )
   thread.start_new_thread( print_time, ("Thread-2", 4, ) )
except:
   print "Error: unable to start thread"

while 1:
   pass
```

慧科
只为赋能人才

网络

- TCP：面向连接
- UDP：面向无连接

- mac地址
- ip地址
- 端口号

- 套接字

慧科
只为赋能人才

UDP编程

- **UDP服务端：**
- import socket
- s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
- server_addr = ('127.0.0.1', 8888)
- s.bind(server_addr)
- data, client_addr = s.recvfrom(BUF_SIZE)
- s.sendto(data, client_addr)

慧科
只为赋能人才

UDP编程

- **UDP客户端：**
- import socket
- server_addr = ('127.0.0.1', 8888)
- client = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
- client.sendto(data, server_addr)
- data, addr = client.recvfrom(BUF_SIZE)

慧科
只为赋能人才

UDP广播

- s.setsockopt(socket.SOL_SOCKET, socket.SO_BROADCAST, 1)
- PORT = 8080
- network = '<broadcast>'
- s.sendto('Client broadcast message!'.encode('utf-8'), (network, PORT))

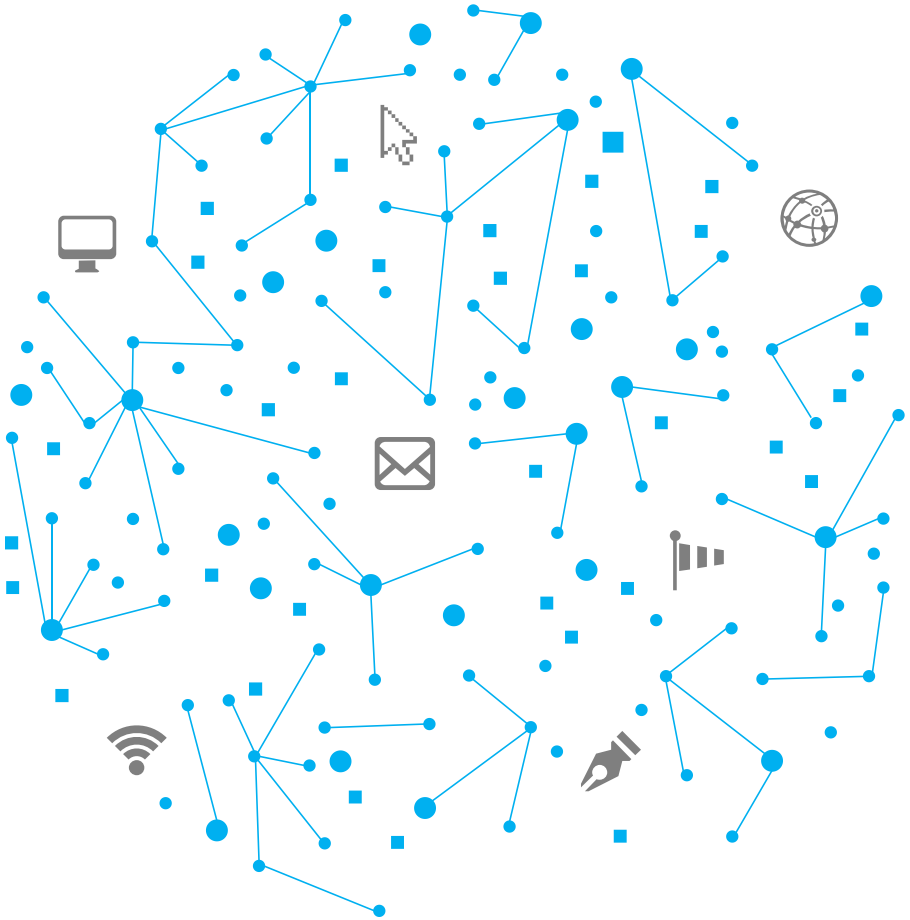
慧科
只为赋能人才

练习

- 使用广播通信，实现小组内4-5台机器人协同
- 1、发消息：同时运动
- 2、发消息：同时停止

慧科

只为赋能人才



THANK YOU

谢谢观看!