

دلال پیام - قسمت اول

پرهام الوانی

۲ آذر ۱۴۰۰

فهرست مطالب

۲	۱ مقدمه
۲	۲ دستورات
۲	۳ پیاده‌سازی سرور
۳	۴ پیاده‌سازی کلاینت

۱ مقدمه

دلال پیام^۱ یک سرور است که پیام‌های دریافتی از کلاينت‌ها را بين آن‌ها تقسيم می‌کند. البته این دلال‌ها کارهای بیشتری هم انجام می‌دهند که خارج از بحث ما می‌باشد. دو عملیات مهم در سرور وجود دارد یکی Publish کردن و دیگری Subscribe کردن است. در عملیات Subscribe کلاينت درخواست گوش دادن رو یک موضوع^۲ خاص را می‌دهد. در Publish کلاينت یک پیام را روی یک موضوع خاص منتشر می‌کند. در نهایت دلال پیام‌ها را به دست **تمام** کسانی که روی آن موضوع گوش می‌دهند می‌رساند.

۲ دستورات

کلاينت به سرور:

Publish

این دستور یک پیام از سمت کلاينت را تحت یک عنوان خاص برای سرور ارسال می‌کند. پیام‌ها رشته‌هایی با طول دلخواه می‌باشند.

Subscribe

این دستور به سرور اعلام می‌کند که این کلاينت متقاضی دریافت پیام‌های عنوان داده شده است.

Ping

این دستور از سمت کلاينت برای اطمینان از ارتباط ارسال می‌گردد.

Pong

این دستور از سمت کلاينت در پاسخ به پیام Ping ارسال می‌گردد.

سرور به کلاينت:

Message

این دستور یک پیام از سمت سرور را برای کلاينت زمانی که متقاضی موضوعی است، ارسال می‌کند. پیام‌ها رشته‌هایی با طول دلخواه می‌باشند.

SubAck

این دستور از سمت سرور در جهت تایید پیام Subscribe کلاينت زمانی که عملیات Subscribe موفقیت آمیز باشد ارسال می‌شود.

PubAck

این دستور از سمت سرور در جهت تایید پیام Publish کلاينت زمانی عملیات Publish موفقیت آمیز باشد ارسال می‌شود.

Ping

این دستور از سمت سرور برای اطمینان از ارتباط ارسال می‌گردد.

Pong

این دستور از سمت سرور در پاسخ به پیام Ping ارسال می‌گردد.

۳ پیاده‌سازی سرور

سرور یک سوکت سرور دارد که روی یک پورت مشخص گوش می‌دهد (در اینجا فرض می‌کنیم پورت موردنظر ۱۳۷۳ است). از آنجایی که قصد داریم سرور چند ارتباط همزمان را هندل کند نیاز به ساخت Thread برای هر کانکشن خواهید داشت تا تابع accept بلافاصله فراخوانی شود. سرور هیچ ارتباطی را نمی‌بیند و همه ارتباط‌ها را به صورت باز نگه می‌دارد. برای سرور هر ارتباط یک کلاينت را نمایندگی می‌کند بنابراین می‌بایست لیست از سوکت‌ها و عنوانین موردنظر آن‌ها داشته باشد.

```
import socket
import threading

HOST = '127.0.0.1' # Standard loopback interface address (localhost)
PORT = 1373        # Port to listen on (non-privileged ports are > 1023)
```

¹Message Broker

²Topic

```
def handler(conn, addr):
    with conn:
        print('Connected by', addr)
        while True:
            data = conn.recv(1024)
            if not data:
                break
            conn.sendall(data)
        print('Disconnected by', addr)

with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
    s.bind((HOST, PORT))
    s.listen()
    while True:
        conn, addr = s.accept()
        threading.Thread(target=handler, args=(conn, addr)).start()
```

۴ پیاده‌سازی کلاینت

کلاینت به صورت یک ابزار درون ترمینالی پیاده‌سازی می‌شود و نیازی به پیاده‌سازی گرافیکی نیست. کلاینت آرگومان‌های زیر را در زمان اجرا دریافت می‌کند. برای اطلاعات بیشتر از آرگومان‌ها در پایتون می‌توانید از اینجا استفاده کنید. کلاینت دو دستور اصلی دارد:

این سند برپایه بسته Persian X۳ گونه 23.1 توسعه پیدا کرده است.