1. create
4. send (send Packet)
5. recieve (recievePacket)
6. close

: Client

send Recieve Socket : Datagram Socket

2. create

3. create

recieve Packet : Datagram Packet

send Packet : Datagram Packet

```
                                    3. create
                                    4. recieve (recievePacket)
┌────────────────────────┐                              ┌──────────────────────────────────────┐
│  : Intermediate Host   │─────────────────────►        │ recieve Socket : Datagram Socket     │
└────────────────────────┘                              └──────────────────────────────────────┘



                                    2. create
                                   ─────────►            ┌──────────────────────────────────────┐
        1. create                                        │ recievePacket : Datagram Packet      │
        6. send                                          └──────────────────────────────────────┘
          (sendPacket)
                                    5. create
        7. recieve(                 8. create
          recievePacket)           ─────────►            ┌──────────────────────────────────────┐
                                                         │ send Packet : Datagram Packet        │
        9. send (sendPacket)                             └──────────────────────────────────────┘



┌──────────────────────────────────────────────┐
│ send Recieve Socket : Datagram  Socket        │
└──────────────────────────────────────────────┘
```

5. create
6. send (sendPacket)
7. close →

:Server

send Socket : Datagram Socket

2. create
3. recieve (recievePacket) →

recieve Socket : Datagram Socket

1. create →

recieve Packet : Datagram Packet

4. create →

send Packet : Datagram Packet