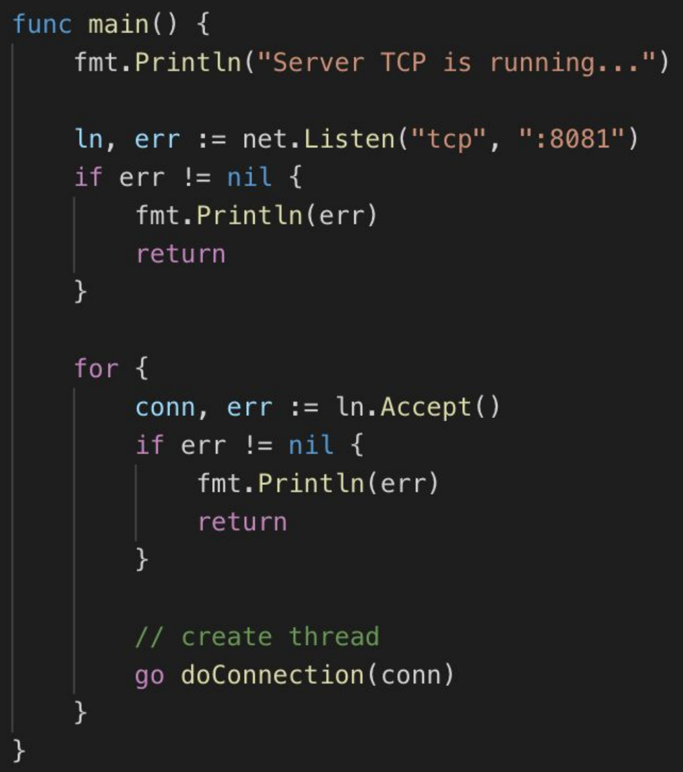
Exercício 01

•••

Caio Gomes, Kácio Henrique e Izabella Melo

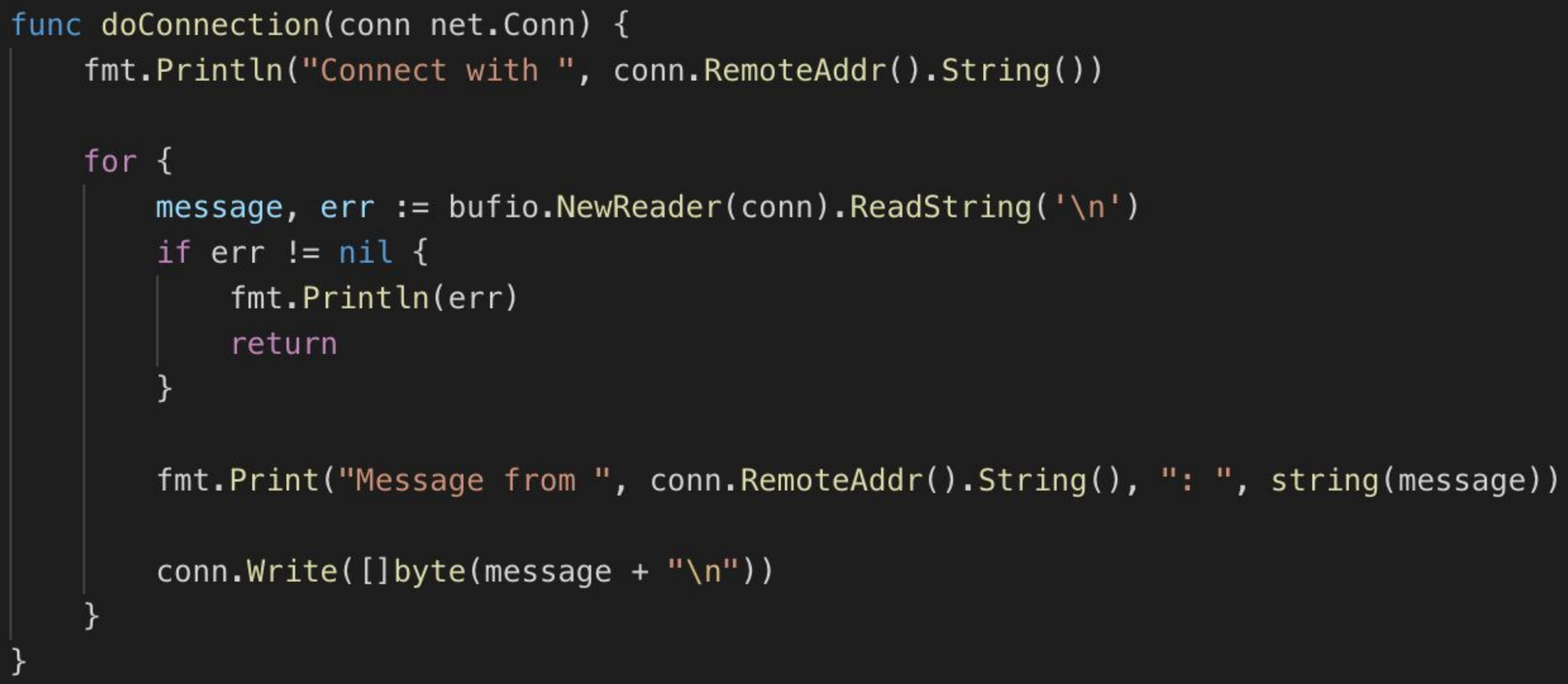
Server TCP

main()



```go
func main() {
    fmt.Println("Server TCP is running...")

    ln, err := net.Listen("tcp", ":8081")
    if err != nil {
        fmt.Println(err)
        return
    }

    for {
        conn, err := ln.Accept()
        if err != nil {
            fmt.Println(err)
            return
        }

        // create thread
        go doConnection(conn)
    }
}
```

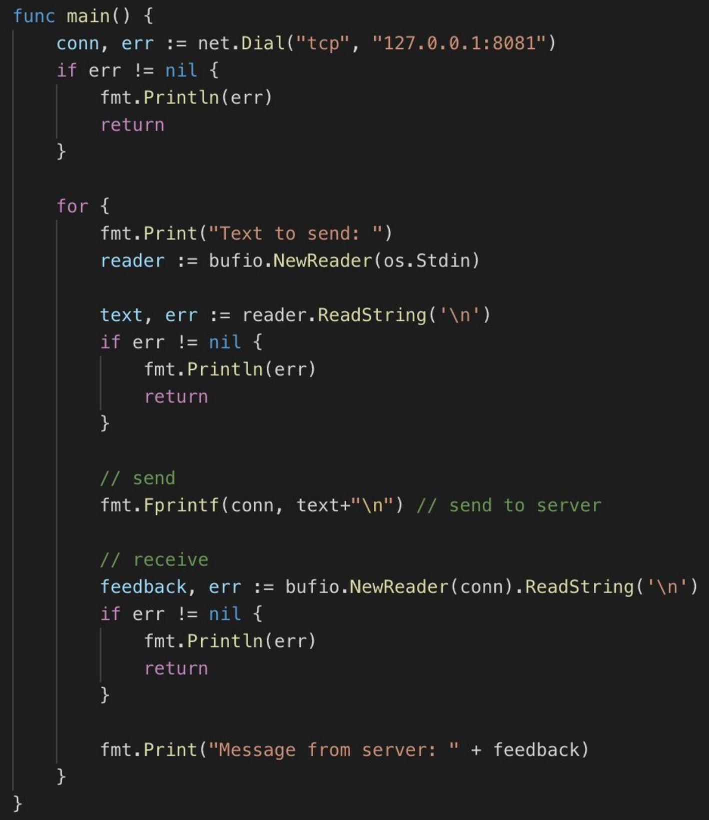
doConnection()



```go
func doConnection(conn net.Conn) {
    fmt.Println("Connect with ", conn.RemoteAddr().String())

    for {
        message, err := bufio.NewReader(conn).ReadString('\n')
        if err != nil {
            fmt.Println(err)
            return
        }

        fmt.Print("Message from ", conn.RemoteAddr().String(), ": ", string(message))

        conn.Write([]byte(message + "\n"))
    }
}
```

Client TCP

main()

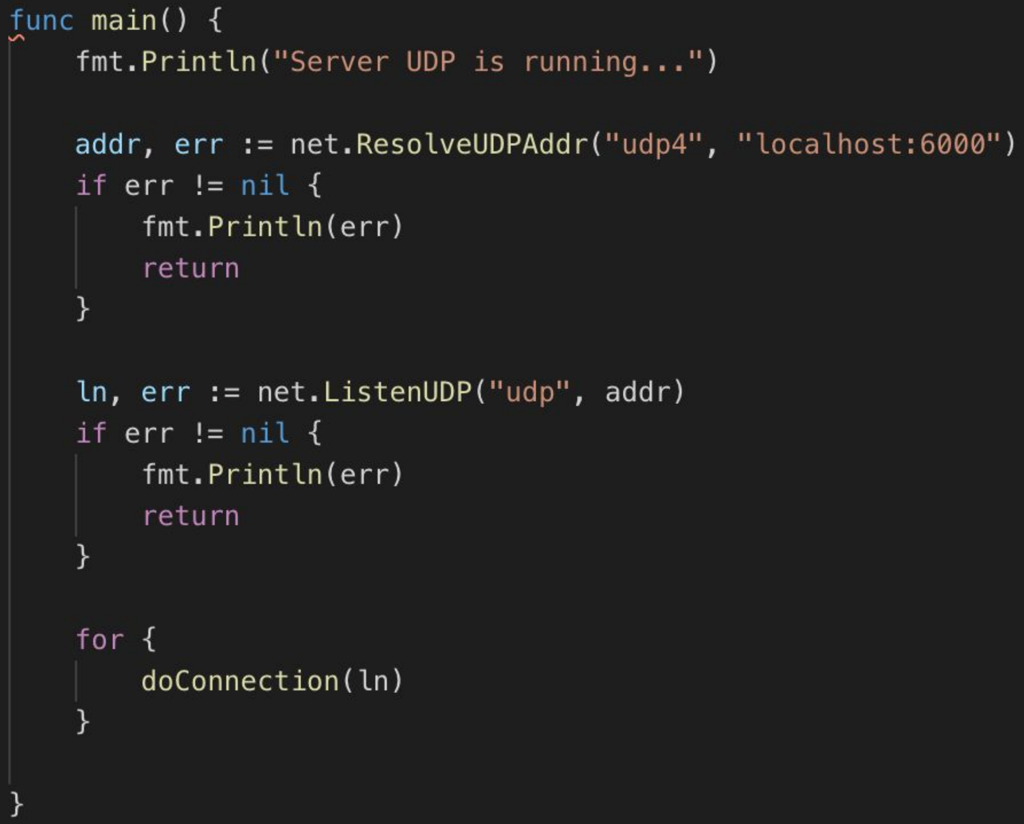


```go
func main() {
    conn, err := net.Dial("tcp", "127.0.0.1:8081")
    if err != nil {
        fmt.Println(err)
        return
    }

    for {
        fmt.Print("Text to send: ")
        reader := bufio.NewReader(os.Stdin)

        text, err := reader.ReadString('\n')
        if err != nil {
            fmt.Println(err)
            return
        }

        // send
        fmt.Fprintf(conn, text+"\n") // send to server

        // receive
        feedback, err := bufio.NewReader(conn).ReadString('\n')
        if err != nil {
            fmt.Println(err)
            return
        }

        fmt.Print("Message from server: " + feedback)
    }
}
```

X

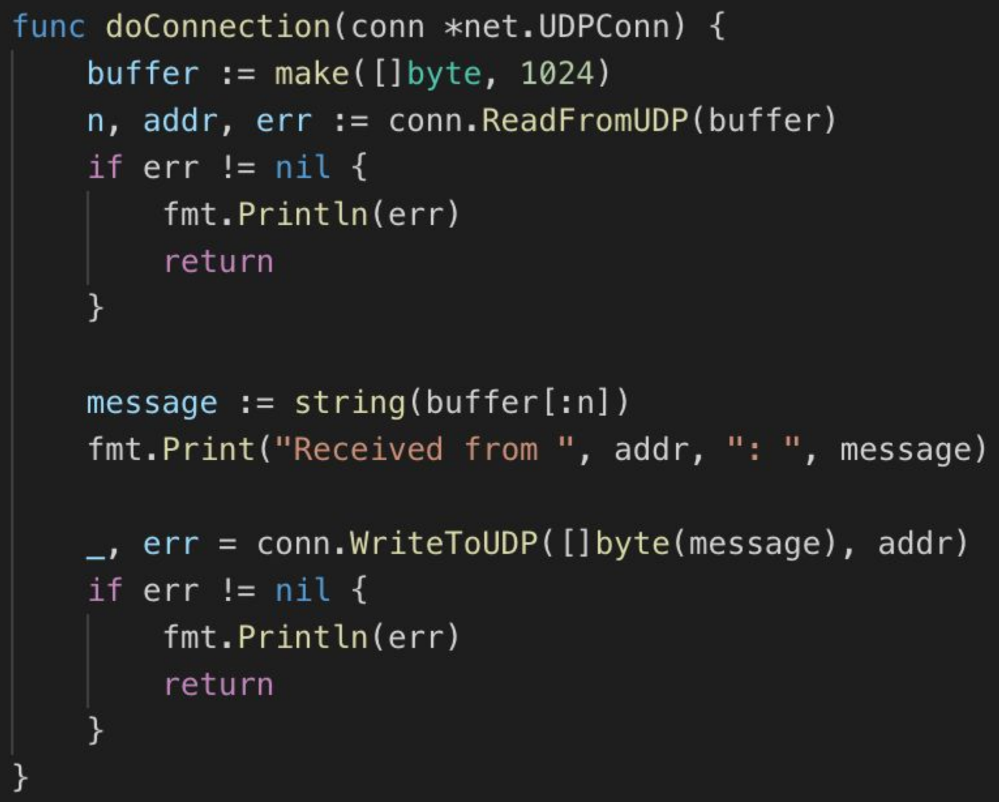
Server UDP

main()



```go
func main() {
    fmt.Println("Server UDP is running...")

    addr, err := net.ResolveUDPAddr("udp4", "localhost:6000")
    if err != nil {
        fmt.Println(err)
        return
    }

    ln, err := net.ListenUDP("udp", addr)
    if err != nil {
        fmt.Println(err)
        return
    }

    for {
        doConnection(ln)
    }

}
```

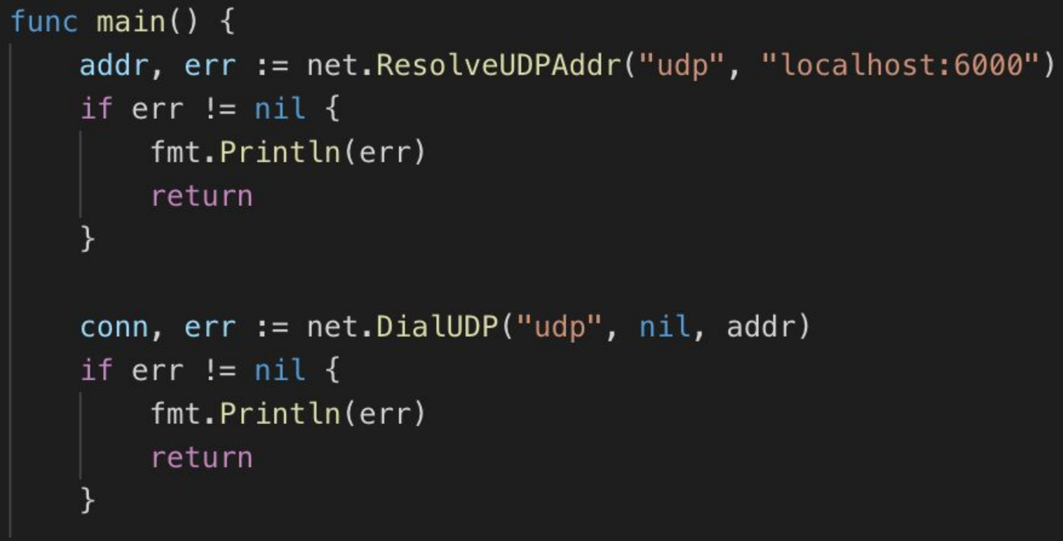
doConnection()



```go
func doConnection(conn *net.UDPConn) {
    buffer := make([]byte, 1024)
    n, addr, err := conn.ReadFromUDP(buffer)
    if err != nil {
        fmt.Println(err)
        return
    }

    message := string(buffer[:n])
    fmt.Print("Received from ", addr, ": ", message)

    _, err = conn.WriteToUDP([]byte(message), addr)
    if err != nil {
        fmt.Println(err)
        return
    }
}
```

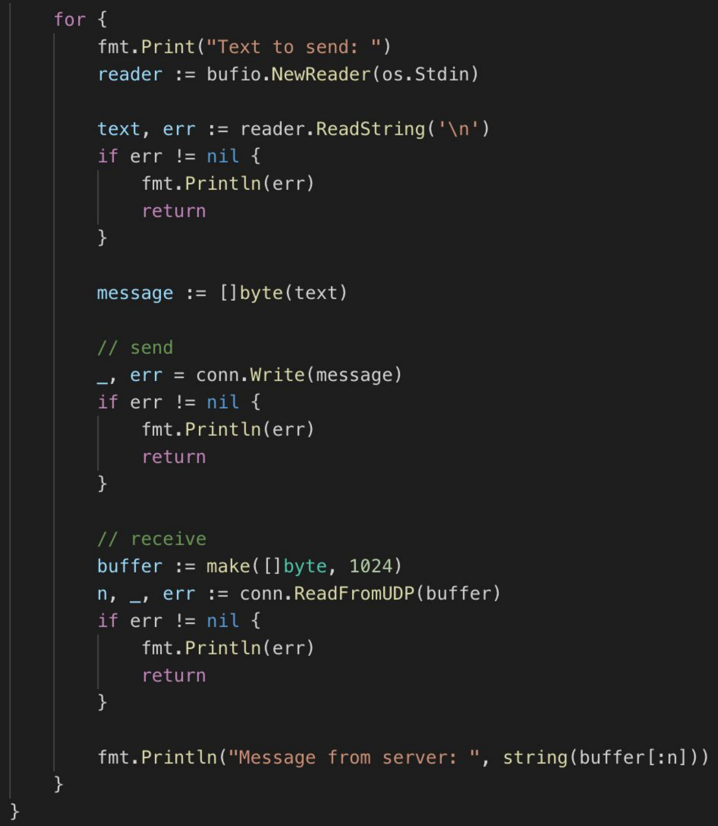
Client UDP

main()



```go
func main() {
    addr, err := net.ResolveUDPAddr("udp", "localhost:6000")
    if err != nil {
        fmt.Println(err)
        return
    }

    conn, err := net.DialUDP("udp", nil, addr)
    if err != nil {
        fmt.Println(err)
        return
    }
```

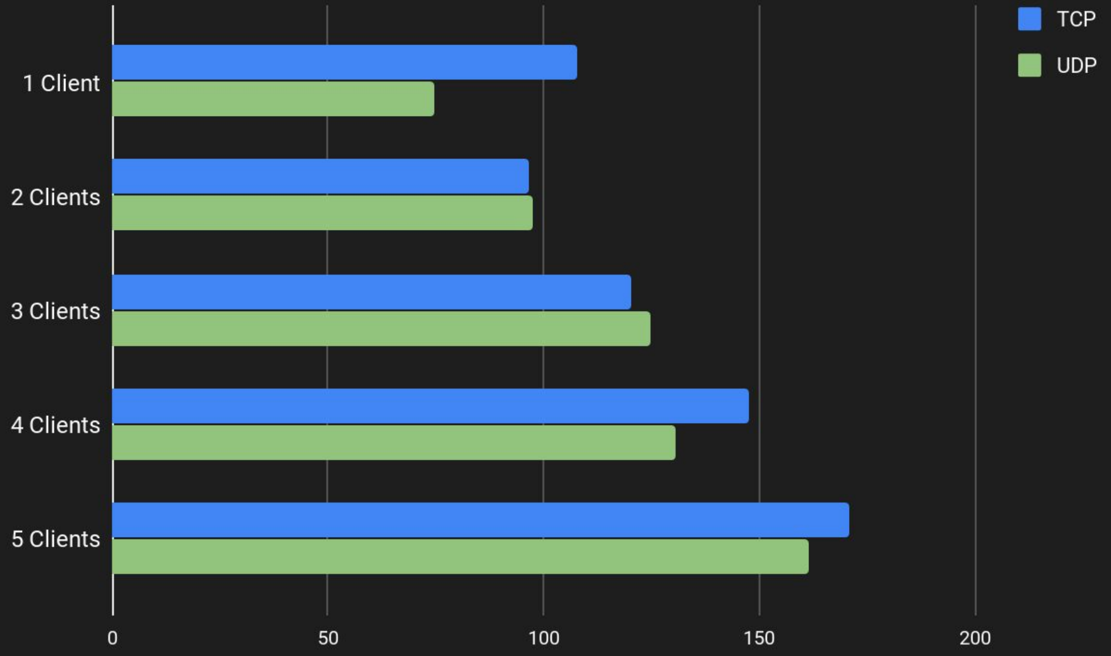
main()



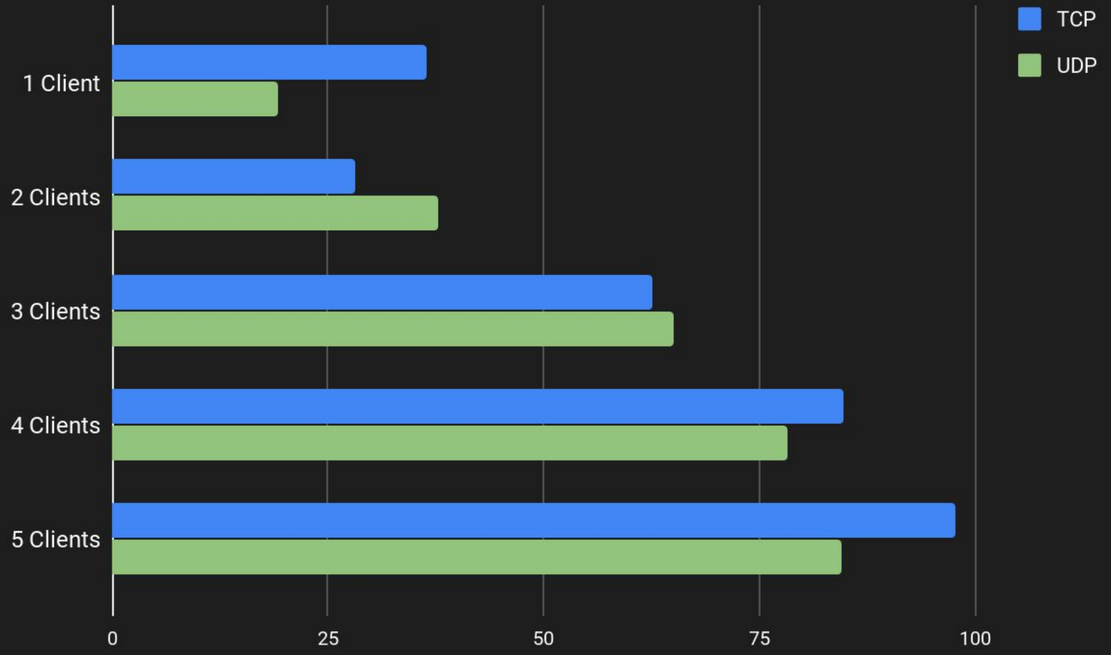
```go
    for {
        fmt.Print("Text to send: ")
        reader := bufio.NewReader(os.Stdin)

        text, err := reader.ReadString('\n')
        if err != nil {
            fmt.Println(err)
            return
        }

        message := []byte(text)

        // send
        _, err = conn.Write(message)
        if err != nil {
            fmt.Println(err)
            return
        }

        // receive
        buffer := make([]byte, 1024)
        n, _, err := conn.ReadFromUDP(buffer)
        if err != nil {
            fmt.Println(err)
            return
        }

        fmt.Println("Message from server: ", string(buffer[:n]))
    }
}
```

Qual o melhor?

Tempo médio de resposta



Desvio padrão



That's all Folks!