

GABRIELE ROMANATO

Menu

Come inviare email con Go

Go, noto anche come Golang, è un linguaggio di programmazione moderno che eccelle per la sua semplicità ed efficienza. In questo articolo, esploreremo come inviare email utilizzando il pacchetto `net/smtp`, incluso nella libreria standard di Go.

Prerequisiti

Prima di iniziare, assicurati di avere:

- Un ambiente di sviluppo Go configurato.
- Accesso a un server SMTP, come Gmail o un provider simile.

Implementazione

Qui di seguito trovi un esempio di codice per inviare un'email utilizzando Go:

```
package main

import (
    "fmt"
    "net/smtp"
)

func main() {
    // Configurazione dell'email
    from := "tuoemail@example.com"
    password := "tuapassword"
    to := []string{"destinatario@example.com"}
    smtpHost := "smtp.example.com"
    smtpPort := "587"

    // Corpo del messaggio
    message := []byte("Subject: Test Email\r\n\r\nCiao! Questa è un'email inviata da Go.")

    // Autenticazione
    auth := smtp.PlainAuth("", from, password, smtpHost)

    // Invio dell'email
    err := smtp.SendMail(smtpHost+":"+smtpPort, auth, from, to, message)
    if err != nil {
        fmt.Println("Errore durante l'invio dell'email:", err)
        return
    }

    fmt.Println("Email inviata con successo!")
}
```

Spiegazione del Codice

Il codice sopra esegue i seguenti passaggi:

1. Imposta le variabili per l'email, tra cui mittente, destinatario, server SMTP e porta.

2. Componi il messaggio dell'email, incluso il campo `Subject`.
3. Utilizza il metodo `smtp.PlainAuth` per l'autenticazione.
4. Invia l'email con `smtp.SendMail`.

Assicurati di sostituire i valori di esempio con quelli reali del tuo account email e server SMTP.

Considerazioni sulla Sicurezza

Quando gestisci credenziali sensibili come password, evita di inserirle direttamente nel codice. Utilizza variabili d'ambiente o un sistema di gestione delle credenziali per una maggiore sicurezza.

Conclusione

In questo articolo, abbiamo mostrato come inviare un'email utilizzando Go e il pacchetto `net/smtp`. Questo approccio è semplice ed efficace, ma per applicazioni più complesse, potresti considerare librerie di terze parti come `gmail`.

Applicazioni Correlate

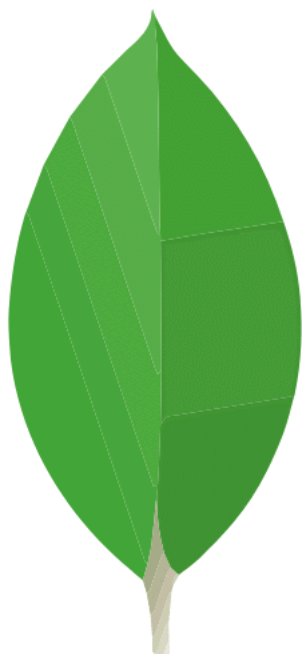


-

Go Placeholder Image

Applicazione in Go per la creazione di immagini segnaposto.

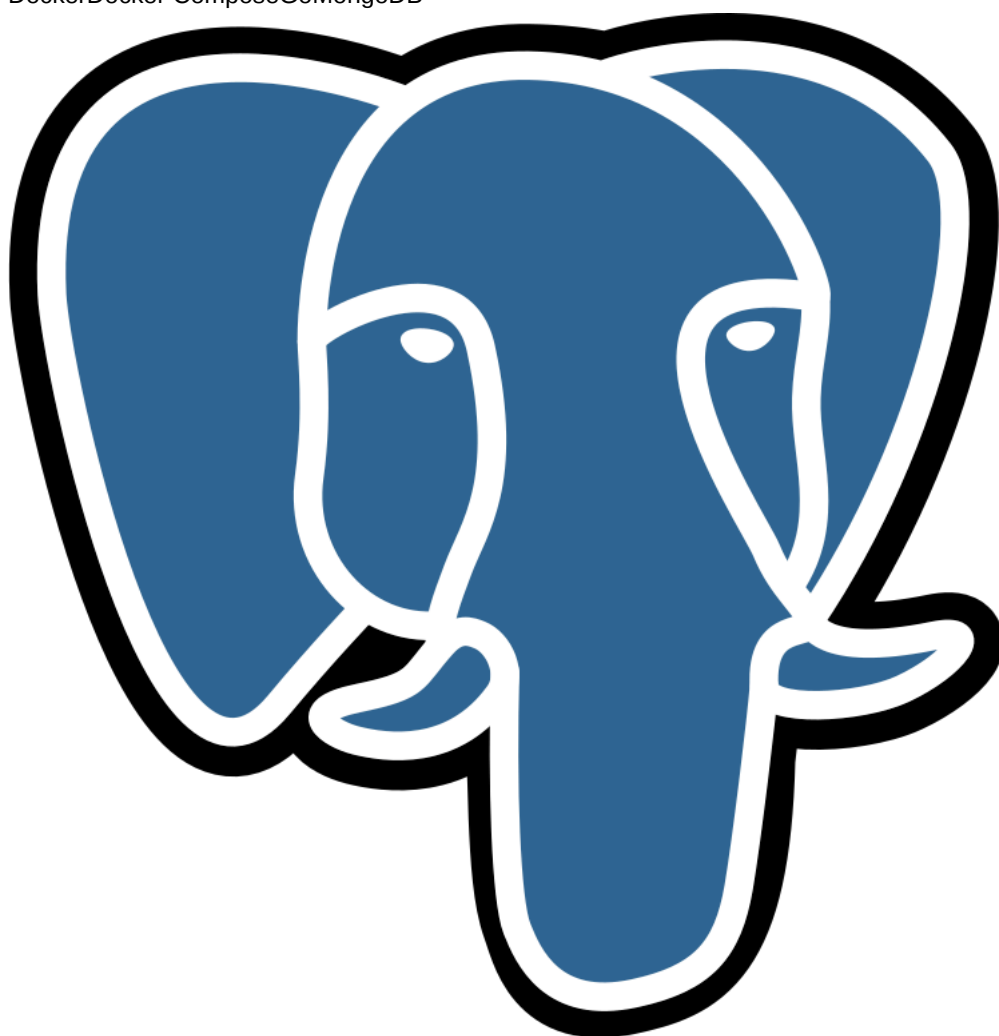
[Docker](#)[Docker Compose](#)[Go](#)[JavaScript](#)



•

Go MongoDB App

Applicazione basata su MongoDB ed implementata in Go con il driver ufficiale.
DockerDocker ComposeGoMongoDB



•

Go PostgreSQL App

Applicazione basata su PostgreSQL e sviluppata in Go.

DockerDocker ComposeGoPostgreSQL