

Come generare un UUID versione 4 con Go

Un UUID (Universally Unique Identifier) è un identificatore unico universalmente utilizzato in molti contesti, come identificatori per risorse, chiavi di database, e molto altro. Esistono diverse versioni di UUID, e in questo articolo ci concentreremo sulla generazione di un UUID versione 4 utilizzando il linguaggio di programmazione Go.

Un UUID versione 4 è un identificatore generato casualmente, basato sul concetto di randomness piuttosto che su informazioni come timestamp o indirizzi MAC (come avviene per altre versioni di UUID). Un UUID v4 ha una lunghezza fissa di 128 bit, e la probabilità che due UUID generati casualmente siano uguali è estremamente bassa, rendendolo ideale per identificatori unici.

Per generare un UUID versione 4, utilizzeremo un pacchetto esterno chiamato `github.com/google/uuid`. Questo pacchetto, sviluppato da Google, offre un'implementazione completa degli UUID per Go.

```
go get github.com/google/uuid
```

Ora siamo pronti per scrivere il codice necessario per generare un UUID v4. Apri il file `main.go` e inserisci il seguente codice:

```
package main

import (
    "fmt"
    "github.com/google/uuid"
)

func main() {
    // Genera un nuovo UUID versione 4
    uuid := uuid.New()
    fmt.Println("Generated UUID v4:", uuid.String())
}
```

Conclusione

In questo articolo, abbiamo visto come generare un UUID versione 4 utilizzando Go. Questo processo è abbastanza semplice grazie al pacchetto `github.com/google/uuid`, che fornisce un'implementazione robusta e facile da usare per lavorare con UUID in Go. Questa tecnica è particolarmente utile quando hai bisogno di identificatori unici per risorse o entità nel tuo progetto.

Applicazioni Correlate

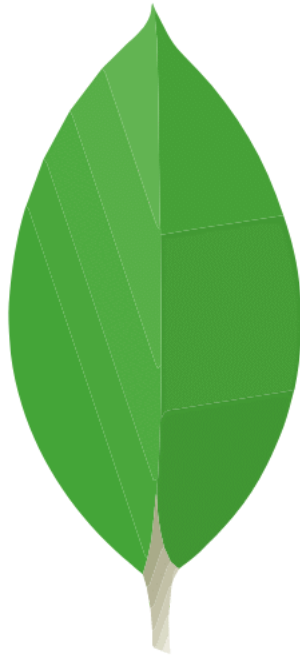


-

Go Placeholder Image

Applicazione in Go per la creazione di immagini segnaposto.

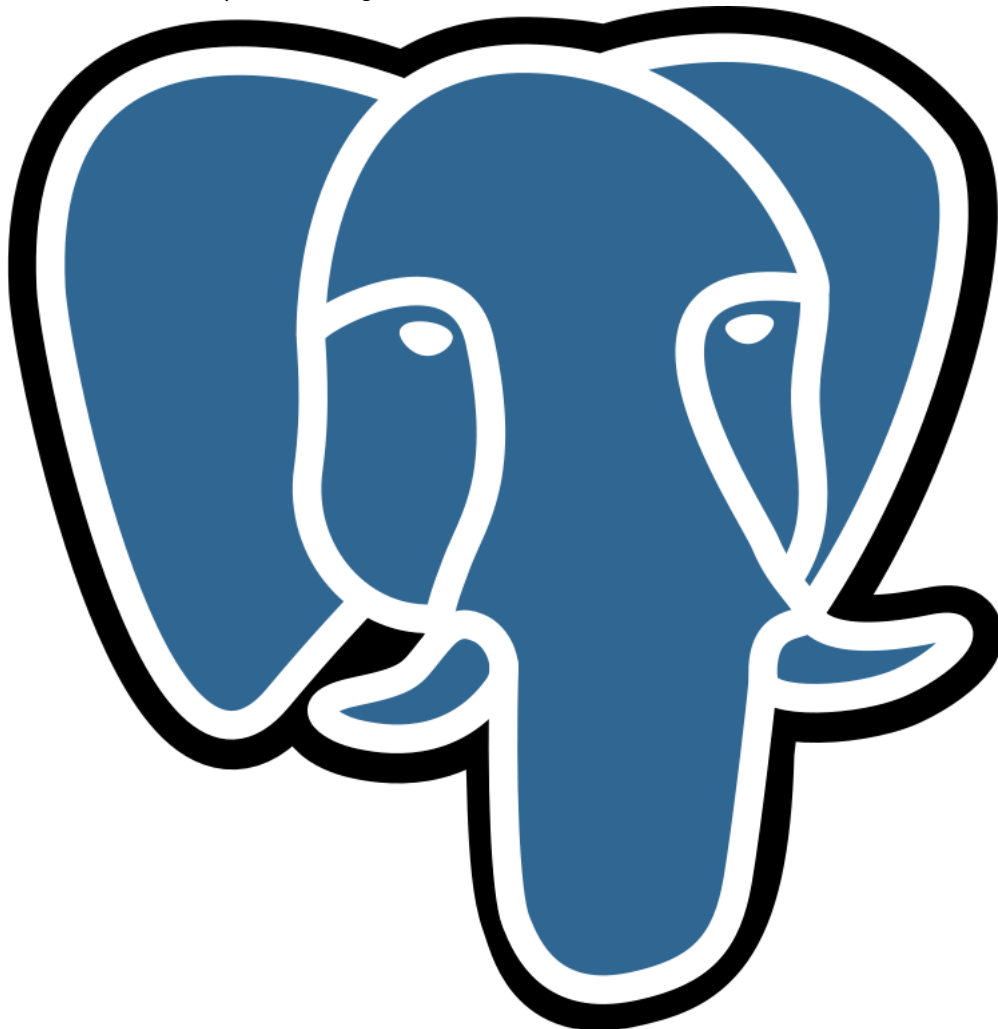
[Docker](#)[Docker Compose](#)[Go](#)[JavaScript](#)



•

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