

# Cobra CLI

개발 및 운영을 위한  
Golang 기반 CLI 라이브러리

백경준  
GDG Golang Korea



# Speaker

Golang Korea

Organizer



- 올해 **Golang Korea**에 합류한 따끈따끈한 고린이
- 국내 최대 온라인 핸드메이드 제품 쇼핑몰 아이디어스의 **DevOps** 엔지니어
- 오픈소스 생태계와 커뮤니티에 관심이 많음
- **TMI**: 최근 결혼하여 장인어른과 야구를 즐겨봄



[paikend](#)

[paikend@gmail.com](mailto:paikend@gmail.com)



**idUS**

아이디어스 채용: <https://team.idus.com/join-idus>

Golang Korea

# 목차

- CLI 알아보기
- CLI 라이브러리 소개
- Cobra CLI 소개
- Cobra CLI 맛보기
- 요약
- Q&A

# CLI 알아보기 - GUI vs CLI




Graphical User Interface

Command Line Interface

# CLI 알아보기 - GUI vs CLI

	GUI	CLI
복잡한 작업	복잡한 작업을 수행하기 어렵거나 불가능	복잡한 작업이나 스크립트를 사용하여 자동화 작업을 쉽게 수행 가능
시스템 자원 사용	이미지와 텍스트를 표시하기 위해 많은 시스템 자원을 사용	텍스트만 표시하므로 상대적으로 적은 시스템 자원을 사용
사용자 친화성	시각적 인터페이스로 인해 사용자 친화적이며 사용 방법을 쉽게 이해 가능	명령어를 기억하고 입력해야하므로 초보자에게는 비교적 친화적이지 않을 수 있음
제어	마우스를 주로 사용하여 화면상의 아이콘, 메뉴 등을 선택하고 클릭	키보드를 사용하여 명령어를 입력하고 실행
멀티태스킹	여러 프로그램을 동시에 실행하고, 창을 이용하여 쉽게 전환	여러 작업을 동시에 실행할 수 있지만, 전환하는 것이 비교적 복잡할 수 있음
학습 곡선	대부분의 사용자에게 친숙하므로 쉽게 배울 수 있음	명령어를 학습하고 사용하는데 시간이 필요하고 상대적으로 가파른 학습 곡선을 가지고 있음

# CLI 알아보기 - CLI Application 특징



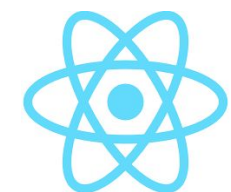
효율적인  
시스템  
자원 사용

자동화  
용이

원격 제어  
용이

소프트웨어 개발, 시스템 관리, 자동화 등  
다양한 분야에서 필수적인 도구로 인식되고 있음

# CLI 라이브러리 소개



Pulumi

=

\$ click\_ 



cobra

Front-end  
개발 도구

Back-end  
개발 도구

IaC  
개발 도구

CLI  
개발 도구

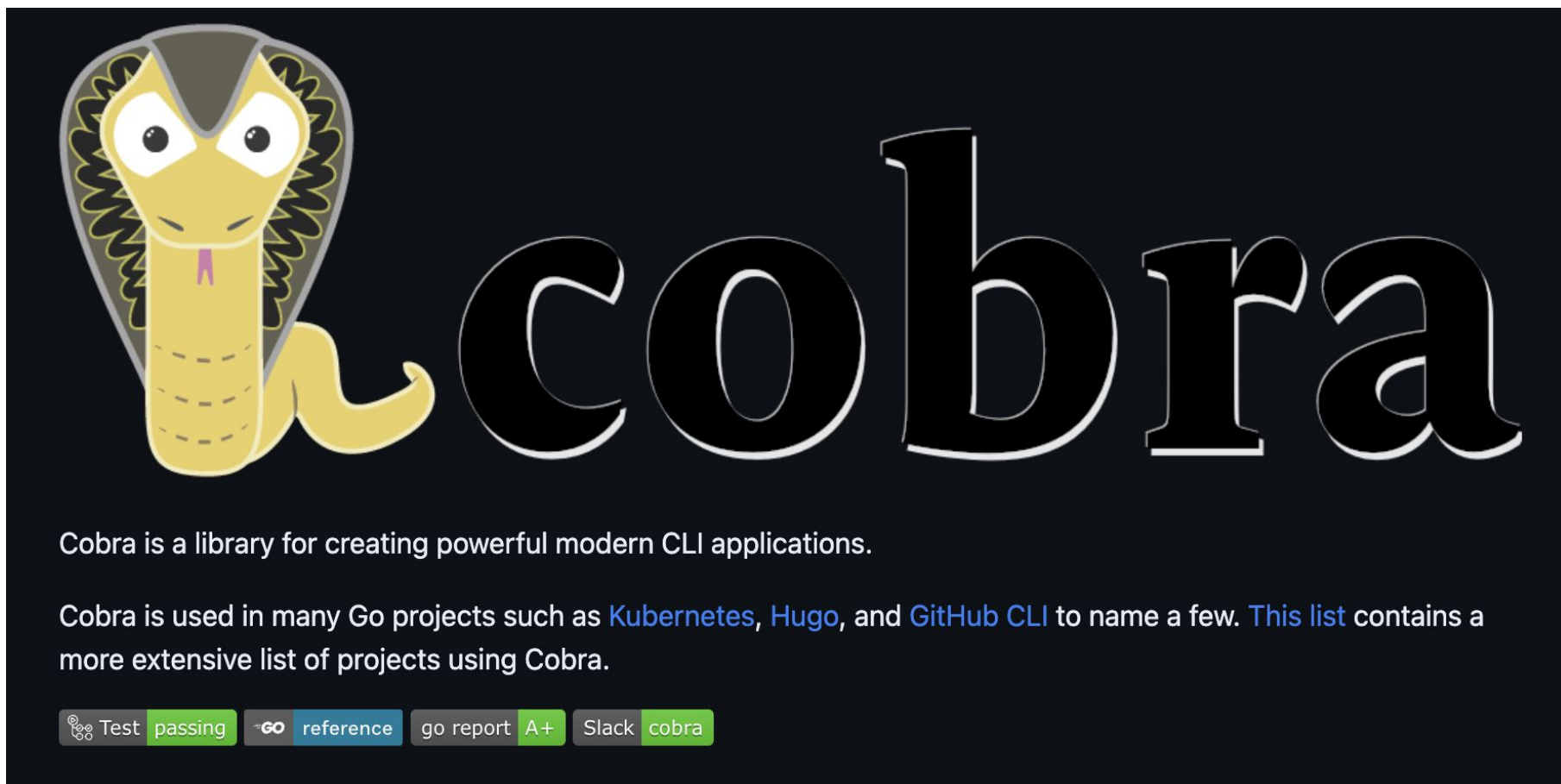
CLI 애플리케이션 개발을 돕는 개발 도구

# CLI 라이브러리 소개 - 유명 CLI 라이브러리 현황

언어	프로젝트	Stars ↓	Forks ↓	최신 릴리즈	릴리즈 날짜
Go	Cobra	32k	2.7k	1.7.0	2023년 4월 4일
JavaScript	Commander.js	24.7k	1.7k	10.0.1	2023년 4월 15일
Python	Click	13.9k	1.3k	8.1.3	2022년 4월 28일
Rust	Clap	11.4k	933	4.3.0	2023년 5월 19일
Ruby	Thor	5k	554	1.2.2	2023년 5월 11일



# Cobra CLI 소개



<https://github.com/spf13/cobra>

# Cobra CLI 소개 - CLI 구성요소

요소	예시	설명
Command	ls	<ul style="list-style-type: none"><li>• 'ls' 명령어는 디렉토리의 내용을 나열하는 기능을 수행함</li></ul>
Argument	/home/user	<ul style="list-style-type: none"><li>• 'ls' 명령어가 실행될 때 나열하고자 하는 디렉토리를 지정하는 인자</li><li>• 예시에서는 '/home/user' 디렉토리를 나열하도록 지정함</li></ul>
Flag	-l	<ul style="list-style-type: none"><li>• 'ls' 명령어의 출력 형태를 변경하는 플래그</li><li>• '-l' 플래그는 'long format'으로 출력하도록 지시함</li></ul>

# Cobra CLI 소개 - Cobra CLI를 사용한 유명 프로젝트

- [Github CLI](#) . . . . . **Git**Hub의 기능을 **CLI**로 사용할 수 있는 도구
- [Kubectl](#) . . . . . **Kubernetes** 클러스터를 제어하기 위한 **CLI** 도구
- [Helm](#) . . . . . **Kubernetes** 애플리케이션을 관리하기 위한 패키지 매니저 **CLI**기능이 내장되어 있음
- [Hugo](#) . . . . . 정적 사이트 개발 도구로 설정, 빌드, 배포 등의 기능이 **CLI**로 구현되어 있음
- [k9s](#) . . . . . 터미널 기반의 대화형 **Kubernetes UI** 도구

# Cobra CLI 맛보기 - Go 프로젝트 초기화

Showing 2 changed files with 5 additions and 0 deletions.

Filter changed files

README.md

go.mod

2 README.md

```
@@ -0,0 +1,2 @@
1 + ## 1. Install
2 + 1. go mod init hello-cli
```

3 go.mod

```
@@ -0,0 +1,3 @@
1 + module hello-cli
2 +
3 + go 1.20
```

```
go mod init ${project_name}
```

# Cobra CLI 맛보기 - Cobra CLI 프로젝트 초기화

Filter changed files

LICENSE

README.md

go.mod

go.sum

0 LICENSE

Empty file.

2 README.md

@@ -1,2 +1,4 @@  
1 ### 1. Install  
2 1. go mod init hello-cli

1 ### 1. Install  
2 1. go mod init hello-cli  
3 + 2. go get -u github.com/spf13/cobra  
4 +

6 go.mod

@@ -1,3 +1,9 @@  
1 module hello-cli  
2  
3 go 1.20

1 module hello-cli  
2  
3 go 1.20  
4 +  
5 + require (  
6 + github.com/inconshreveable/mousetrap v1.1.0 // indirect  
7 + github.com/spf13/cobra v1.7.0 // indirect  
8 + github.com/spf13/pflag v1.0.5 // indirect  
9 + )

10 go.sum

@@ -0,0 +1,10 @@

1 + github.com/cpuguy83/go-md2man/v2 v2.0.2/go.mod h1:tgQtvFLXSQ0S0SIRvRPT7W67SCa46tRH0mNcaadrF8o=  
2 + github.com/inconshreveable/mousetrap v1.1.0 h1:wN+x4NVGpMs07ErUn/mUI3vEoE6Jt13X2s0bqwp9tc8=  
3 + github.com/inconshreveable/mousetrap v1.1.0/go.mod  
4 + github.com/russross/blackfriday/v2 v2.1.0/go.mod  
5 + github.com/spf13/cobra v1.7.0 h1:hyqWnYt1ZQShIdd05k8Bpj3vu05/+x6tJ6dg8EC572I=  
6 + github.com/spf13/cobra v1.7.0/go.mod h1:uLxZILRyS/50Wlh0IKD7W6V5bgeIt+4sICxh6uMRb0=  
7 + github.com/spf13/pflag v1.0.5 h1:iy+VFU0CP1a+8yFto/drg2CJ5u0yRoB7fZw3DKv/JXA=  
8 + github.com/spf13/pflag v1.0.5/go.mod h1:McXfInJRz4CXVZ0BLb0bTZqETkiAhM9Iw0y3An2Bg=  
9 + gopkg.in/check.v1 v0.0.0-20161208181325-20d25e280405/go.mod  
10 + gopkg.in/yaml.v3 v3.0.1/go.mod h1:K4uyk7z7BCEPqu6E+C64Yfvc1Q7kz7rIZviUmN+EgEM=

go get -u github.com/spf13/cobra

# Cobra CLI 맛보기 - Cobra CLI 'greet' 명령어 추가

README.md

cmd

greet.go

cobra-cli add greet

```
1 + /*
2 + Copyright © 2023 NAME HERE <EMAIL ADDRESS>
3 +
4 + */
5 + package cmd
6 +
7 + import (
8 +     "fmt"
9 +
10 +     "github.com/spf13/cobra"
11 + )
12 +
13 + // greetCmd represents the greet command
14 + var greetCmd = &cobra.Command{
15 +     Use:   "greet",
16 +     Short: "A brief description of your command",
17 +     Long: `A longer description that spans multiple lines and likely contains examples
18 + and usage of using your command. For example:
19 +
20 + Cobra is a CLI library for Go that empowers applications.
21 + This application is a tool to generate the needed files
22 + to quickly create a Cobra application.`,
23 +     Run: func(cmd *cobra.Command, args []string) {
24 +         fmt.Println("greet called")
25 +     },
26 + }
27 +
28 + func init() {
29 +     rootCmd.AddCommand(greetCmd)
30 +
31 +     // Here you will define your flags and configuration settings.
32 +
33 +     // Cobra supports Persistent Flags which will work for this command
34 +     // and all subcommands, e.g.:
35 +     // greetCmd.PersistentFlags().String("foo", "", "A help for foo")
36 +
37 +     // Cobra supports local flags which will only run when this command
38 +     // is called directly, e.g.:
39 +     // greetCmd.Flags().BoolP("toggle", "t", false, "Help message for toggle")
40 + }
```

# Cobra CLI 맛보기 - 'Hello world' 출력하기

Filter changed files

README.md

cmd

greet.go

21 cmd/greet.go

```
... @@ -1,6 +1,5 @@
1  /*
2  Copyright © 2023 NAME HERE <EMAIL ADDRESS>
3  -
4  */
5  package cmd
6
7  @@ -13,28 +12,12 @@ import (
13  // greetCmd represents the greet command
14  var greetCmd = &cobra.Command{
15      Use: "greet",
16      - Short: "A brief description of your command",
17      - Long: `A longer description that spans multiple lines and likely contains examples
18      - and usage of using your command. For example:
19      -
20      - Cobra is a CLI library for Go that empowers applications.
21      - This application is a tool to generate the needed files
22      - to quickly create a Cobra application.`,
23      Run: func(cmd *cobra.Command, args []string) {
24      -      fmt.Println("greet called")
25      },
26  }
27
28  func init() {
29      rootCmd.AddCommand(greetCmd)
30
31      - // Here you will define your flags and configuration settings.
32      -
33      - // Cobra supports Persistent Flags which will work for this command
34      - // and all subcommands, e.g.:
35
36
37
38
39      - // greetCmd.Flags().BoolP("toggle", "t", false, "Help message for toggle")
40  }
```

```
1  /*
2  Copyright © 2023 NAME HERE <EMAIL ADDRESS>
3  */
4  package cmd
5
6
7
8
9
10
11
12  // greetCmd represents the greet command
13  var greetCmd = &cobra.Command{
14      Use: "greet",
15      + Short: "Greets a user",
16
17
18
19
20
21
22
23  Run: func(cmd *cobra.Command, args []string) {
24      +      fmt.Printf("Hello World\n")
25      },
26  }
27
28  func init() {
29      rootCmd.AddCommand(greetCmd)
30
31
32
33
34
35
36
37
38
39
40  }
```

./hello-cli greet



# Cobra CLI 맛보기 - 'Hello \${인자값}' 출력하기

```
@@ -4,4 +4,5 @@
3. cobra-cli init
4. cobra-cli add greet
5. update cmd/greet.go
7 - 1. Print Hello World
8 + 2. Print Hello ${name}, default name is World

@@ -9,15 +9,18 @@ import (
    "github.com/spf13/cobra"
)

// greetCmd represents the greet command
var greetCmd = &cobra.Command{
    Use: "greet",
    Short: "Greets a user",
    Run: func(cmd *cobra.Command, args []string) {
17 -     fmt.Printf("Hello World\n")
18     },
19 }
20
21 func init() {
22     rootCmd.AddCommand(greetCmd)
23 }
```

```
9     "github.com/spf13/cobra"
10 )
11
12 + var name string
13 +
14 // greetCmd represents the greet command
15 var greetCmd = &cobra.Command{
16     Use: "greet",
17     Short: "Greets a user",
18     Run: func(cmd *cobra.Command, args []string) {
19 +     fmt.Printf("Hello %s\n", name)
20     },
21 }
22
23 func init() {
24 +     greetCmd.Flags().StringVarP(&name, "name", "n", "World", "Name to greet")
25     rootCmd.AddCommand(greetCmd)
26 }
```

`./hello-cli greet -n paikend`



# Cobra CLI 맛보기 - 'Hello \${인자값}' 반복 출력하기

The screenshot shows a code editor with two files: `README.md` and `cmd/greet.go`. The `README.md` file shows a list of tasks, with task 3 highlighted in green. The `cmd/greet.go` file shows the implementation of the `greet` command, with the `Run` function modified to include a loop that prints the greeting multiple times based on the `count` flag.

```
@@ -6,3 +6,4 @@  
5. update cmd/greet.go  
1. Print Hello World  
2. Print Hello ${name}, default name is World  
+ 3. (Print Hello ${name}, default name is World) * ${count}, default count is 1  
  
@@ -10,17 +10,21 @@ import (  
10 )  
11  
12 var name string  
  
13  
14 // greetCmd represents the greet command  
15 var greetCmd = &cobra.Command{  
16     Use: "greet",  
17     Short: "Greets a user",  
18     Run: func(cmd *cobra.Command, args []string) {  
19 -         fmt.Printf("Hello %s\n", name)  
20  
21     },  
22 }  
23  
24 func init() {  
25     greetCmd.Flags().StringVarP(&name, "name", "n", "World", "Name to greet")  
26  
27     rootCmd.AddCommand(greetCmd)  
  
10 )  
11  
12 var name string  
13 + var count int  
14  
15 // greetCmd represents the greet command  
16 var greetCmd = &cobra.Command{  
17     Use: "greet",  
18     Short: "Greets a user",  
19     Run: func(cmd *cobra.Command, args []string) {  
20 +         for i := 0; i < count; i++ {  
21 +             fmt.Printf("Hello %s\n", name)  
22 +         }  
23     },  
24 }  
25  
26 func init() {  
27     greetCmd.Flags().StringVarP(&name, "name", "n", "World", "Name to greet")  
28 +     greetCmd.Flags().IntVarP(&count, "count", "c", 1, "Number of times to print the greeting")  
29     rootCmd.AddCommand(greetCmd)  
30 + }
```

`./hello-cli greet -n paikend -c 3`

# Cobra CLI 맛보기 - makefile을 통한 명령어 자동화

The screenshot shows a code editor with two files open: README.md and makefile. The README.md file contains a table of contents with line numbers 10, 11, and 12. The makefile file contains a Makefile for hello-cli with line numbers 1 through 21. The editor interface includes a sidebar on the left with a search bar and file list, and a top bar with file icons.

```
10 3. (Print Hello ${name}, default name is World) * ${count}, default count is 1
11 ### 4. Results
12 ![execute_image](./assets/execute_image.png)

1 + # Makefile for hello-cli
2 +
3 + # Variables
4 + APP_NAME := hello-cli
5 + GO_BUILD := go build
6 + GO_RUN := go run
7 +
8 + # Build Targets
9 + .PHONY: build
10 + build:
11 +     $(GO_BUILD) -o $(APP_NAME)
12 + .PHONY: run
13 + run:
14 +     $(GO_RUN) main.go
15 +
16 + .PHONY: clean
17 + clean:
18 +     rm -f $(APP_NAME)
19 +
20 + .PHONY: all
21 + all: clean build run
```

# Cobra CLI 맛보기

## 실행 결과

```
✖ paikend@paikend-mac ~/go/src/hello-cli ▶ ? main ± ▶ ./hello-cli greet
Hello World
paikend@paikend-mac ~/go/src/hello-cli ▶ ? main ± ▶ ./hello-cli greet -n paikend
Hello paikend
paikend@paikend-mac ~/go/src/hello-cli ▶ ? main ± ▶ ./hello-cli greet --name paikend
Hello paikend
paikend@paikend-mac ~/go/src/hello-cli ▶ ? main ± ▶ ./hello-cli greet --name paikend -c 3
Hello paikend
Hello paikend
Hello paikend
```

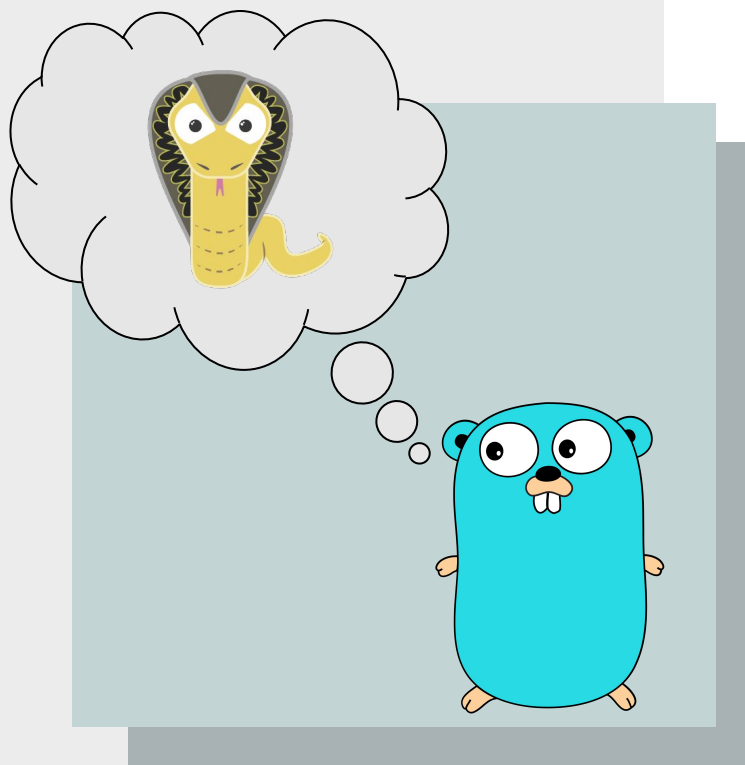
# Cobra CLI 맛보기

소스 코드 공유

<https://github.com/paikend/hello-cli>



# 요약



**CLI** 애플리케이션은 개발 도구, 자동화, 크롤러 등 다양한 분야에서 유용하게 사용할 수 있습니다.

**CLI** 애플리케이션 개발이 필요한 순간이 오면, **Cobra CLI**도 같이 떠올려 봅시다!



# Q & A





# Thank you