

FUKUOKA

GO

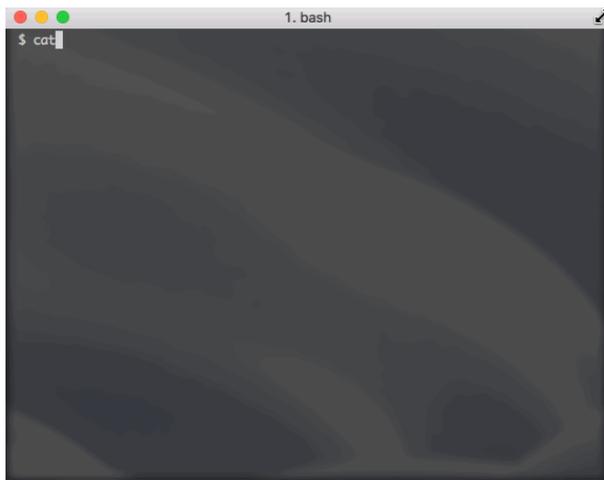
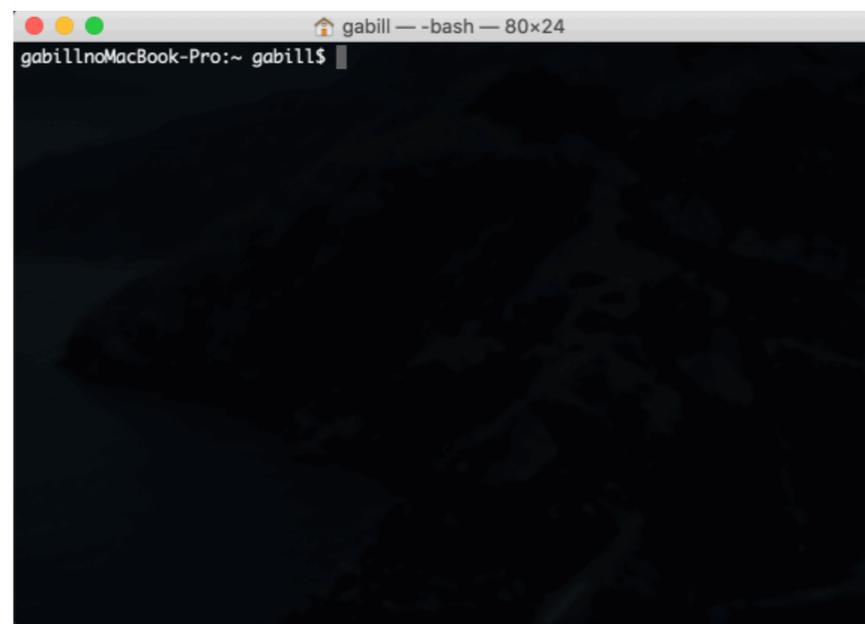
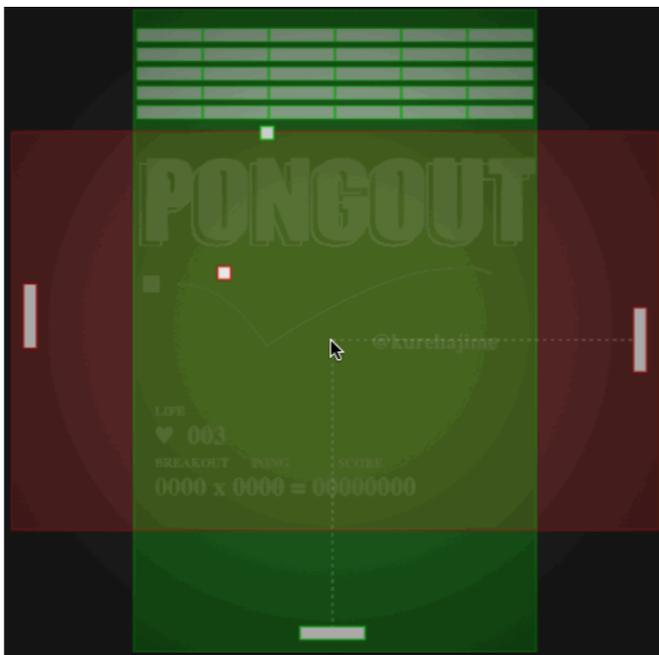
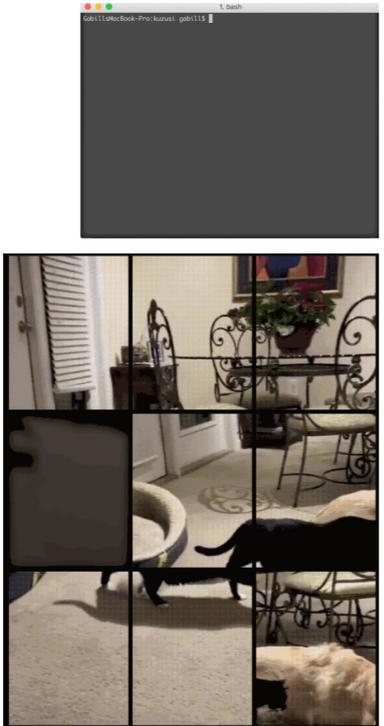
SHI

MA

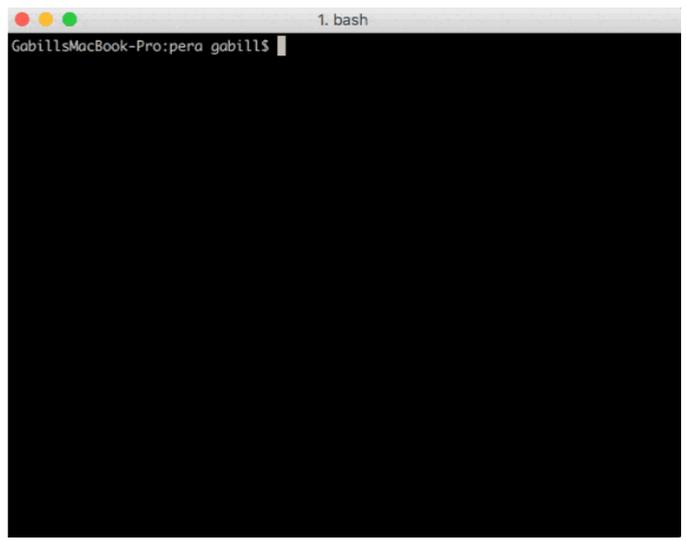
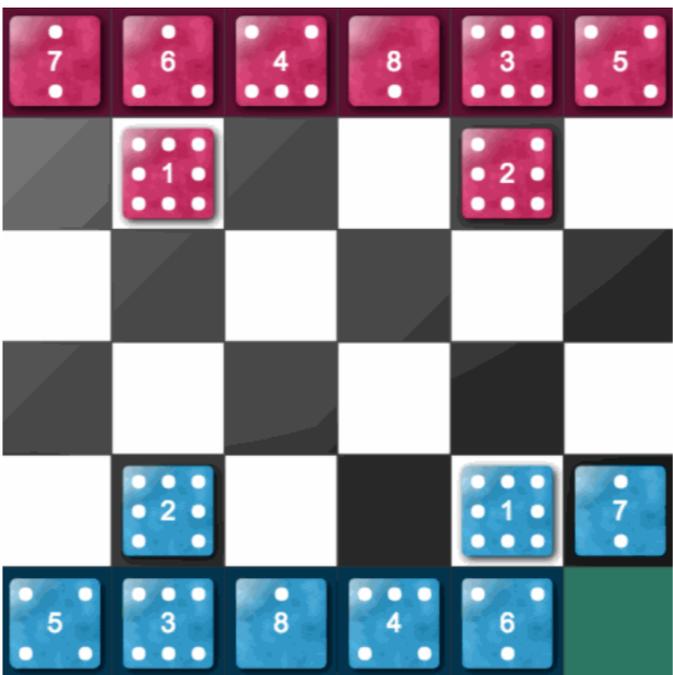
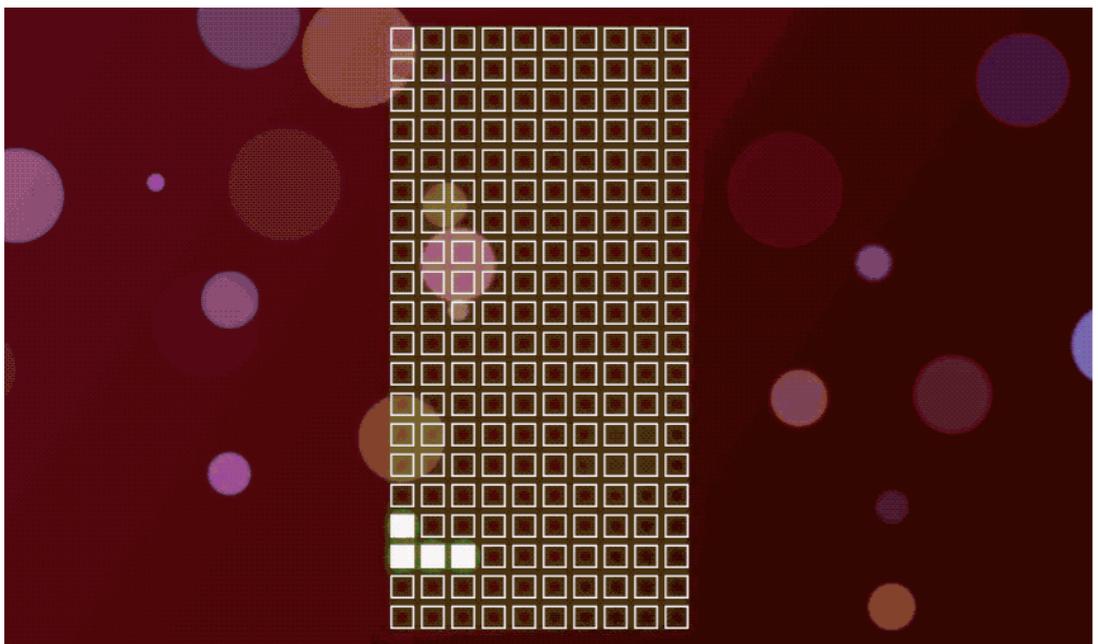
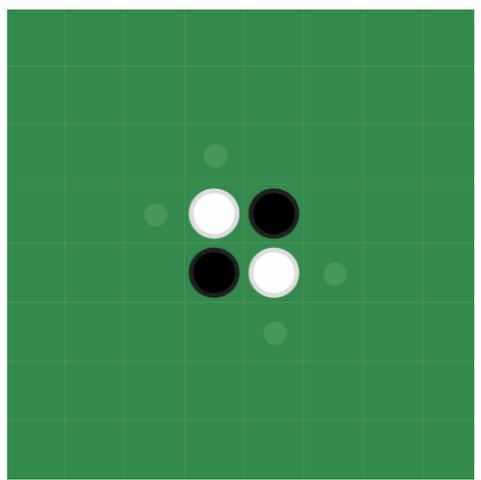
CUIのツールを1行で  
Web GUI化する  
ライブラリを作ってみた

@ kurehajime

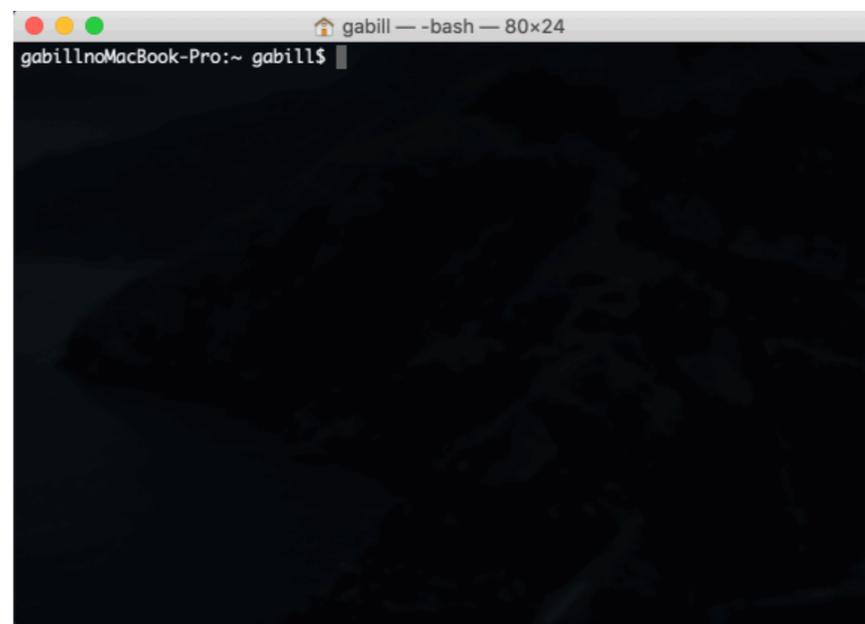
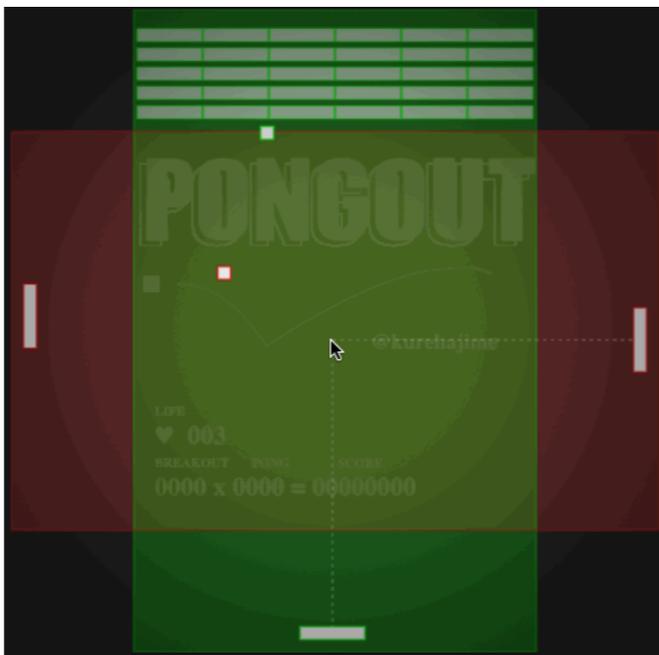
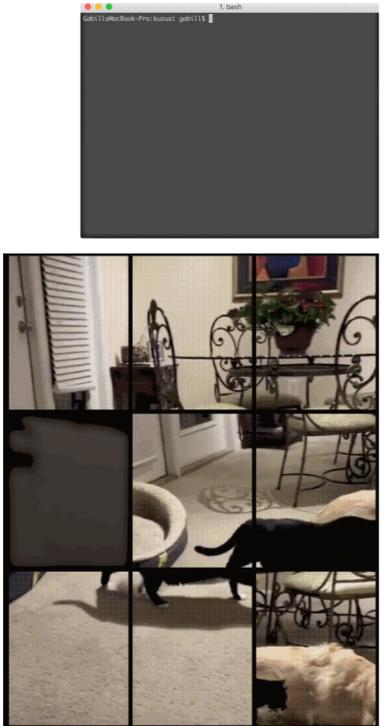
# 自己紹介



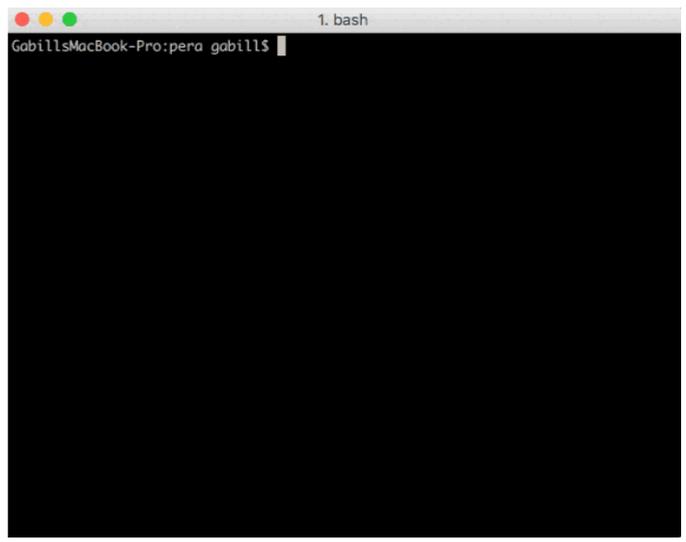
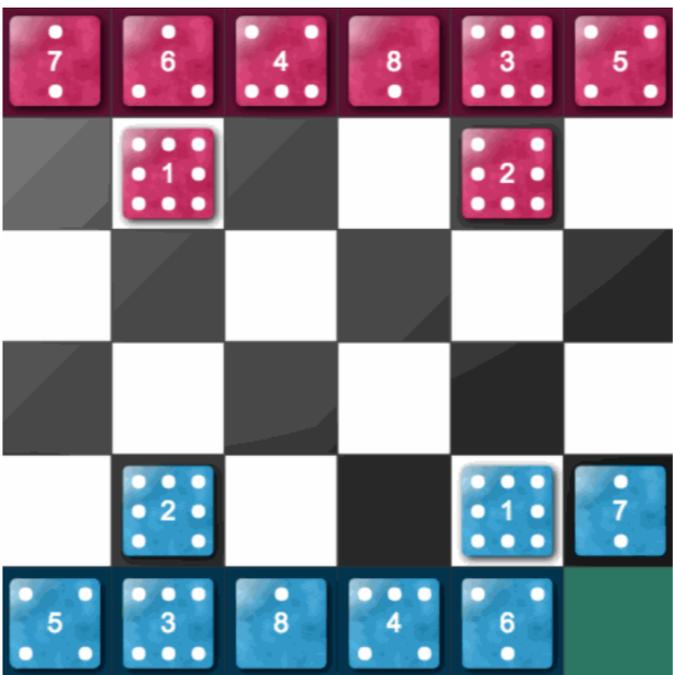
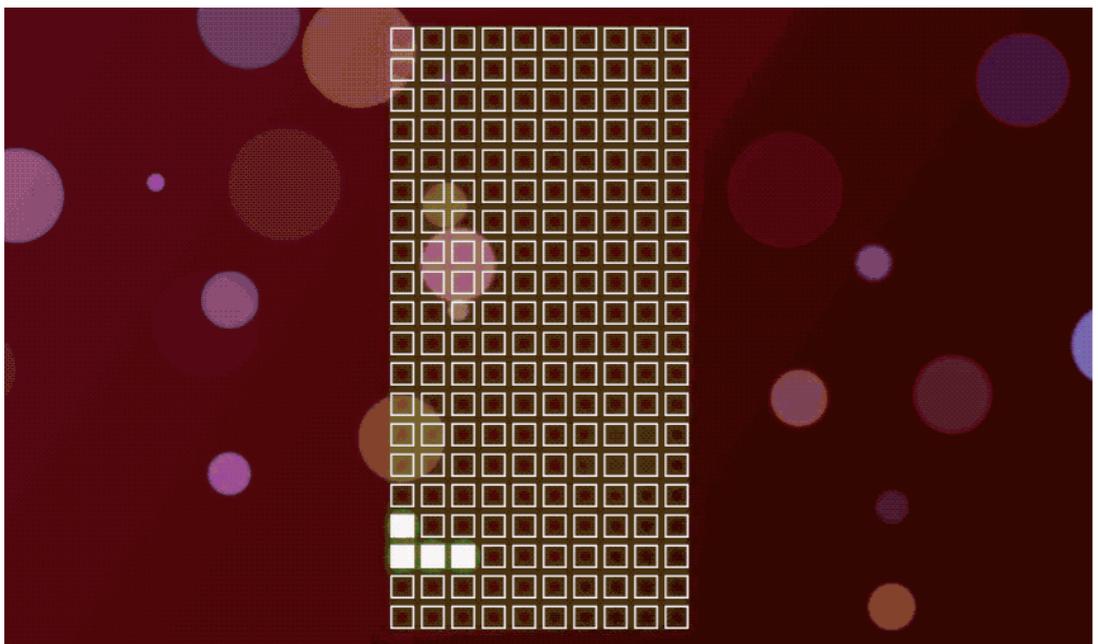
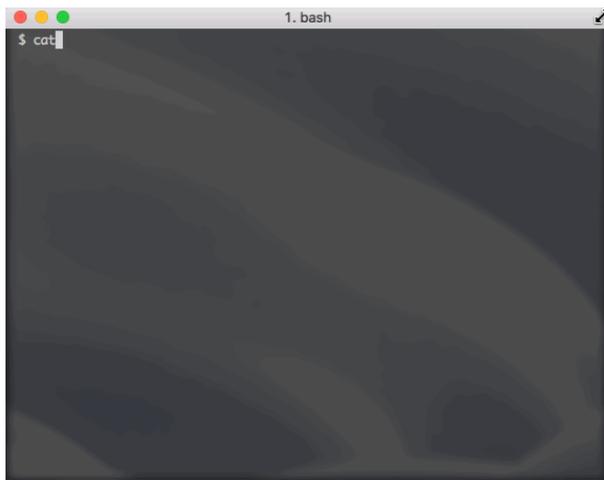
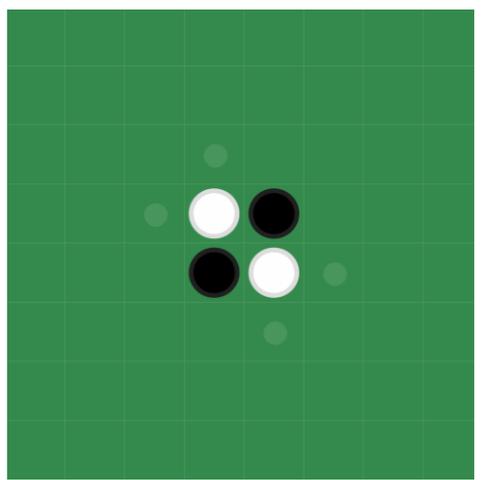
**ハンドルネームは@kurehajimeです。  
鹿児島にいます。  
大学時代は福岡にいました。**



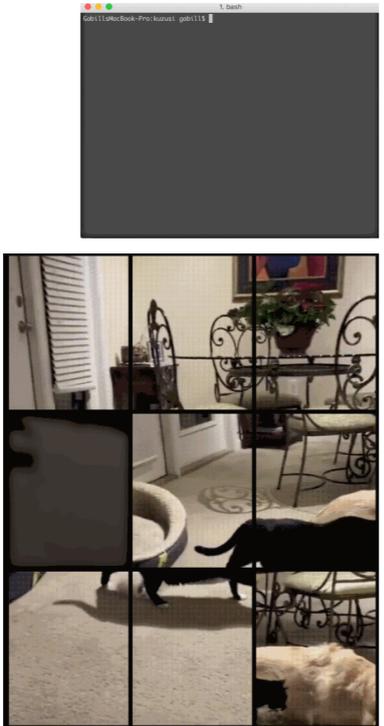
# 自己紹介



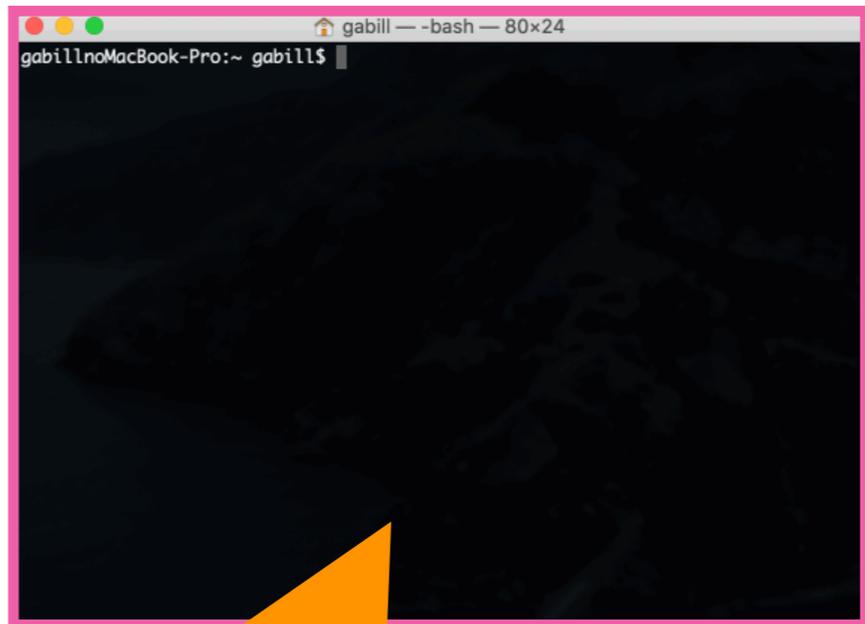
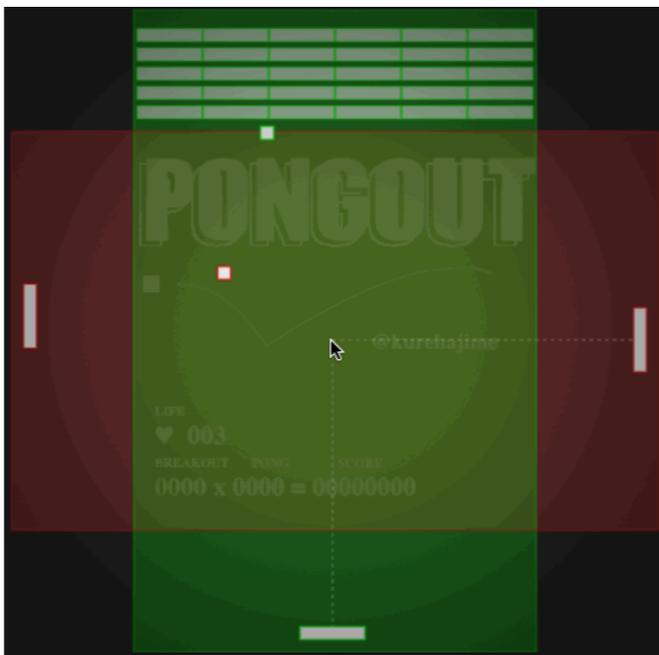
**Slerとして働いています。  
趣味でもいろいろ作ってます。  
Go言語関連で言えば…**



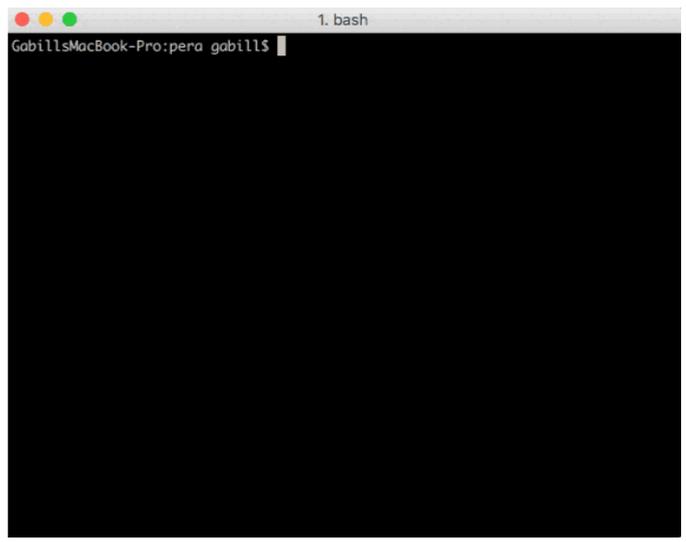
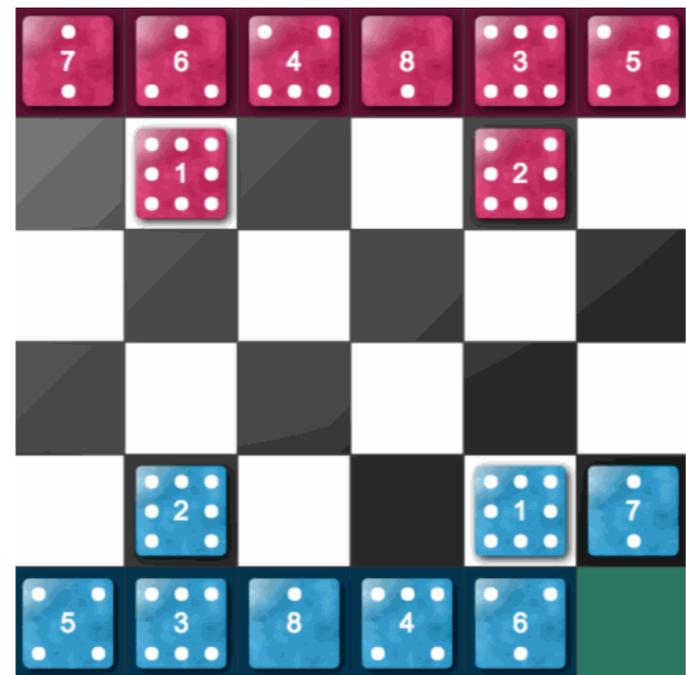
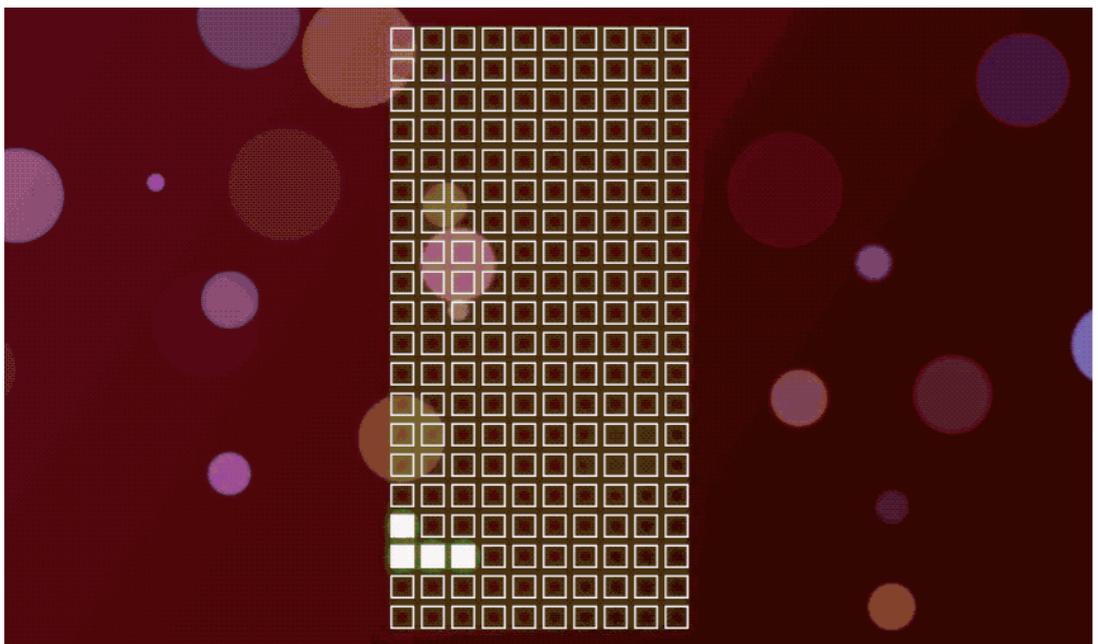
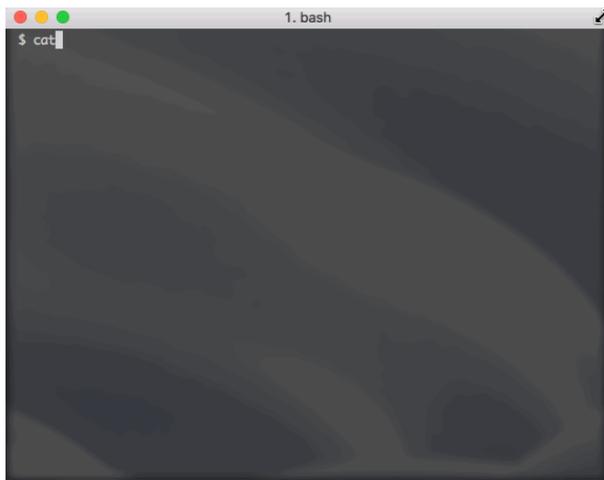
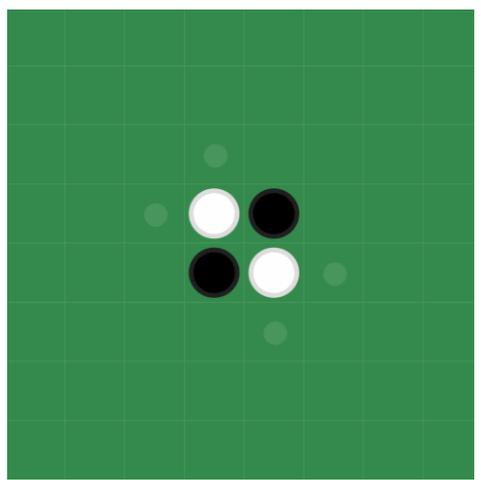
# 自己紹介



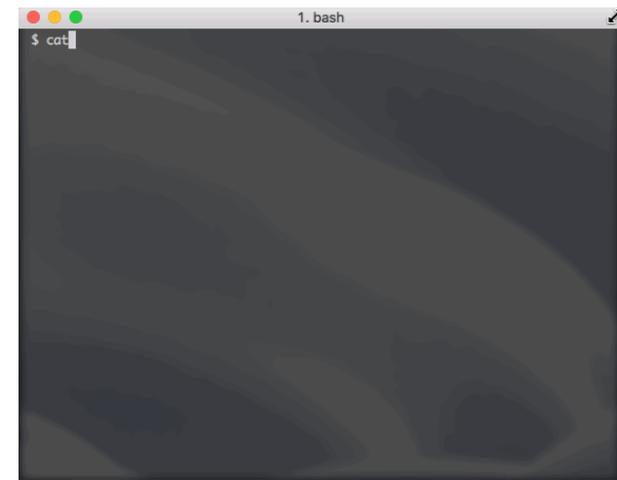
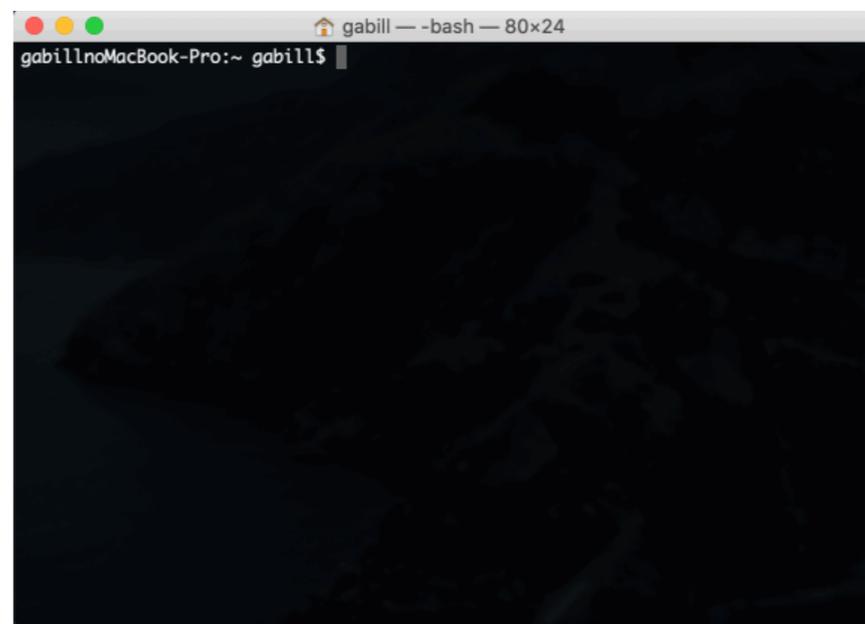
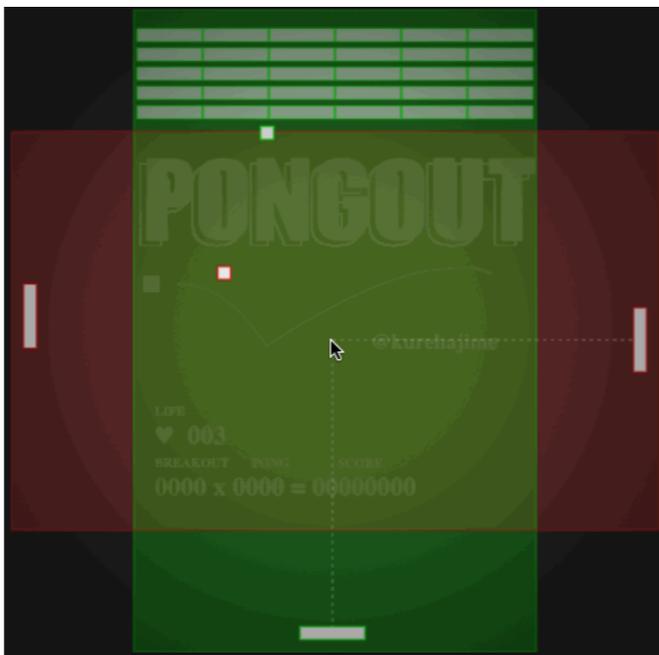
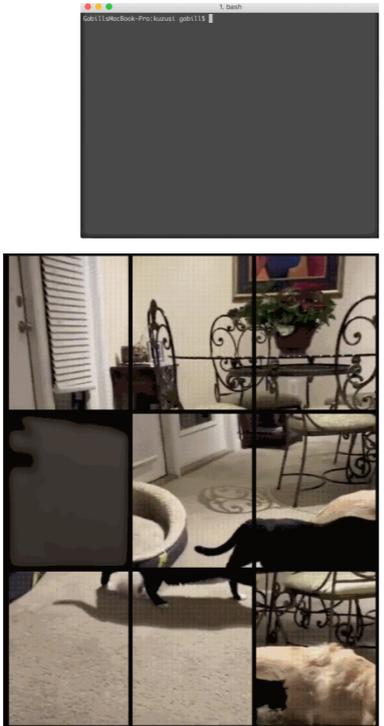
<https://github.com/kurehajime/pong-command>



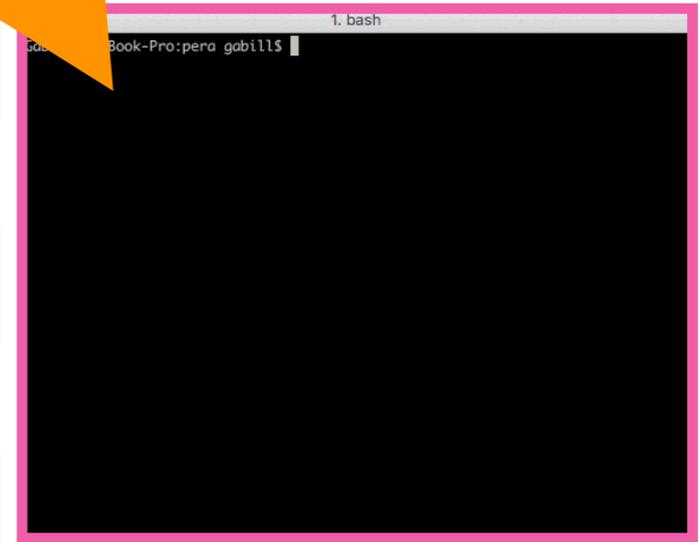
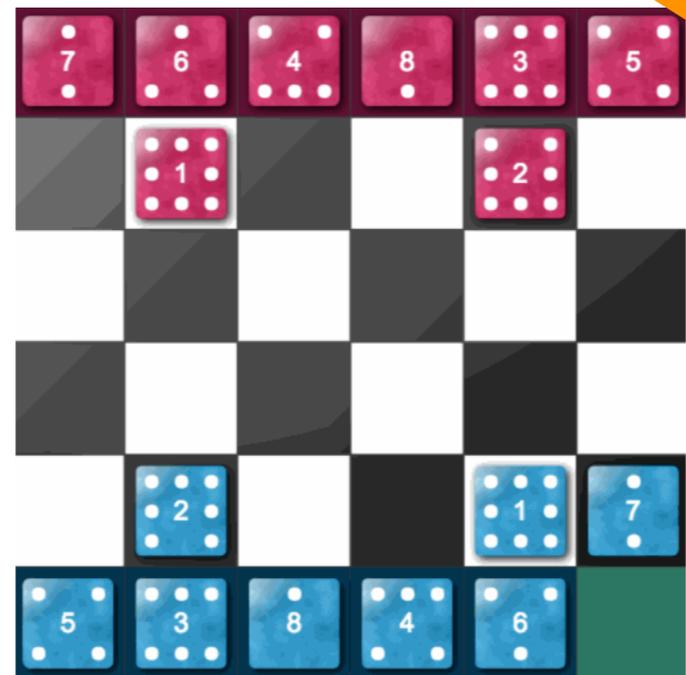
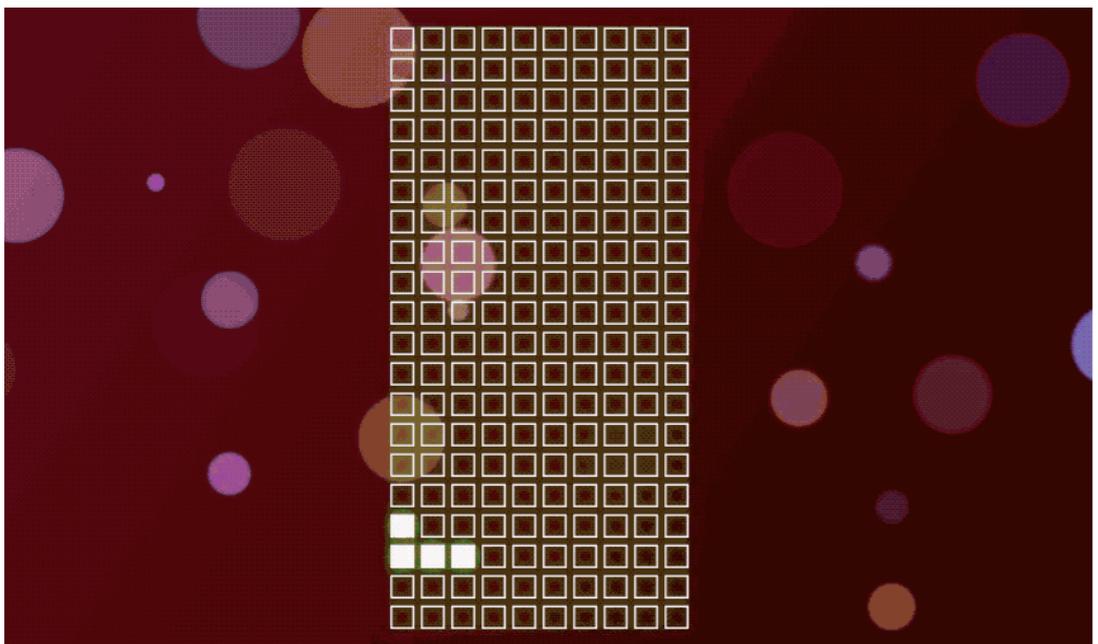
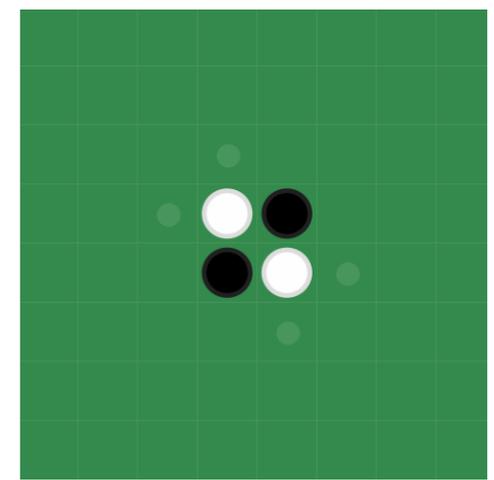
## Pingコマンドのパロディ IPアドレスを跳ね返して遊べる 『pong』コマンド



# 自己紹介



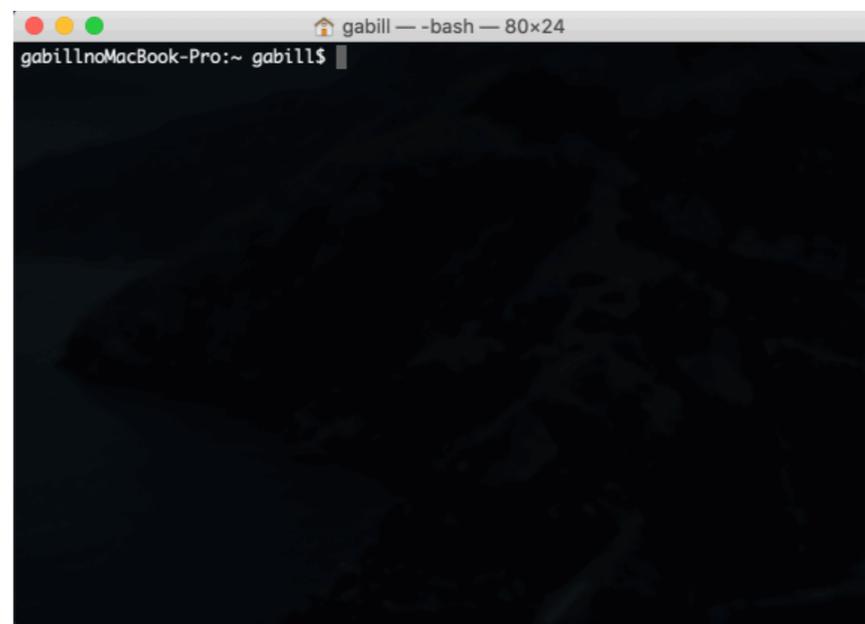
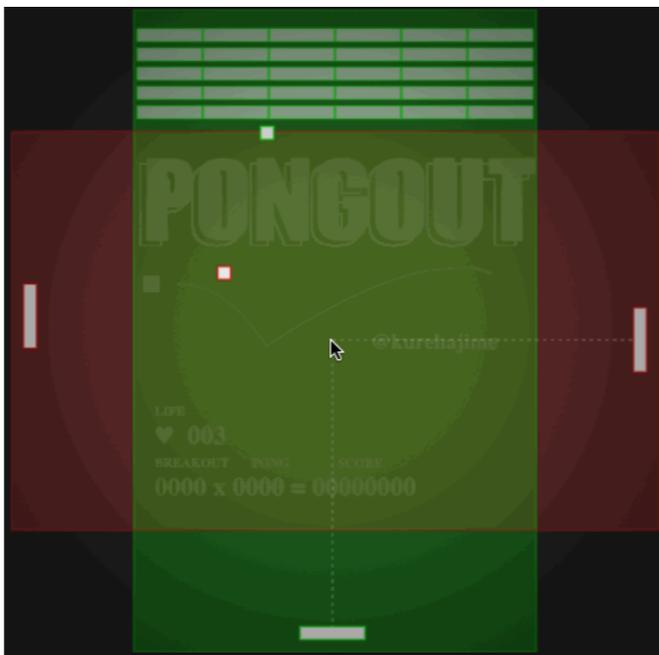
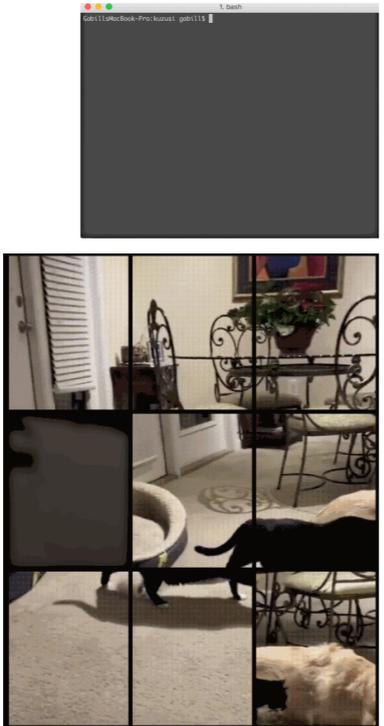
markdownを使って  
コンソールでプレゼンできる『pera』  
(今回は使いません)



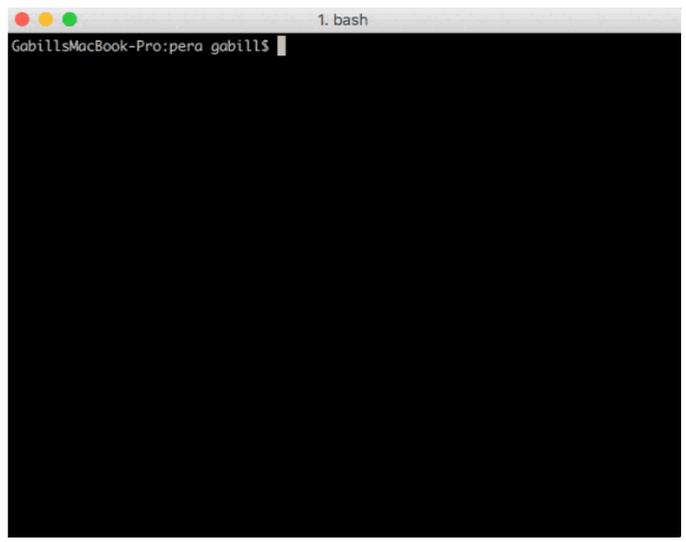
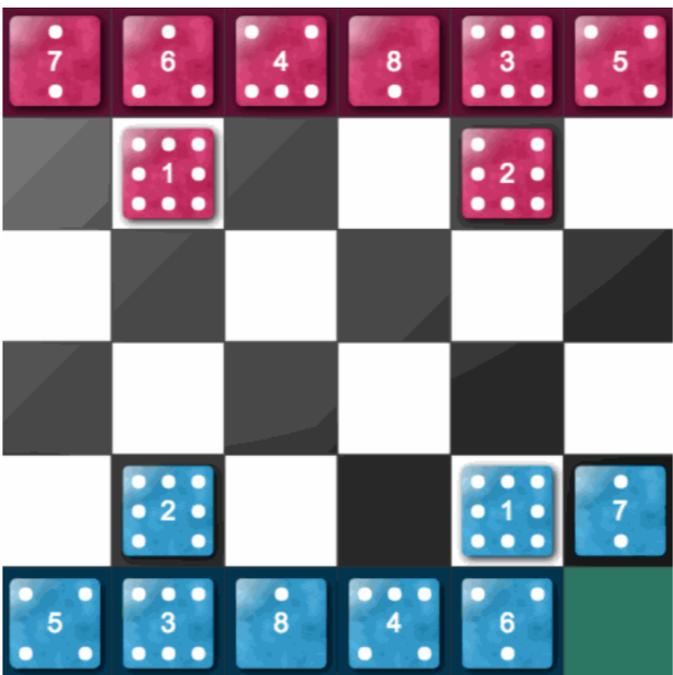
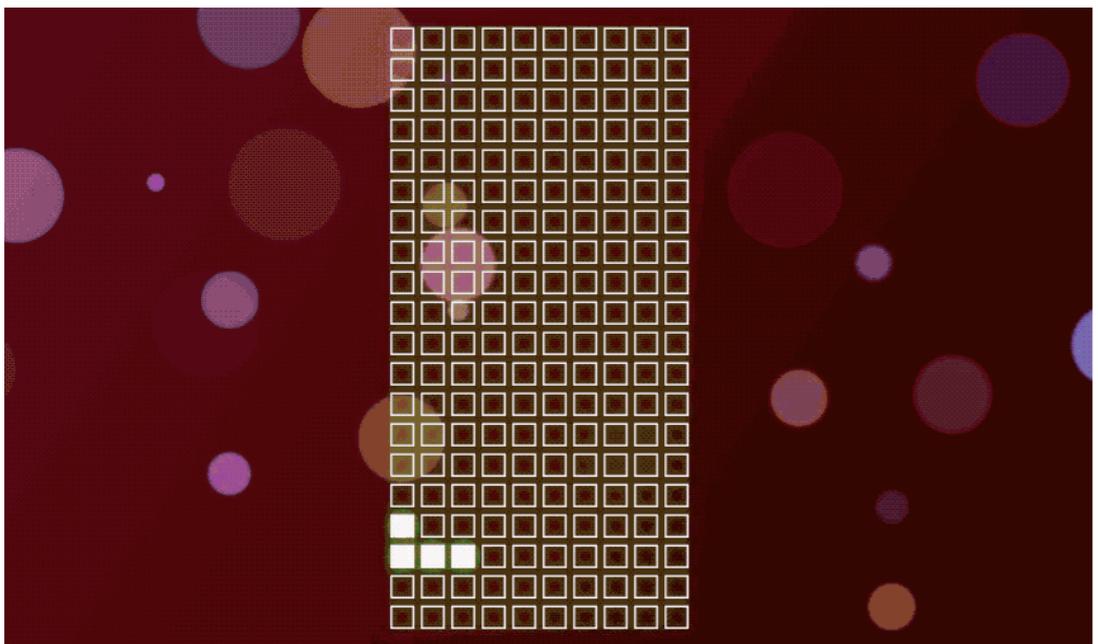
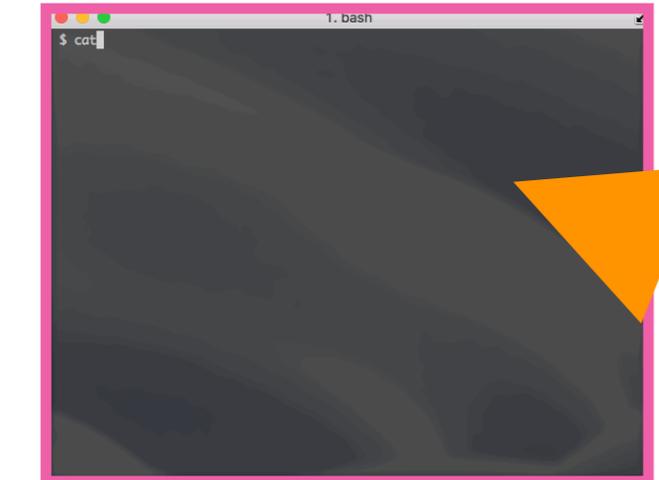
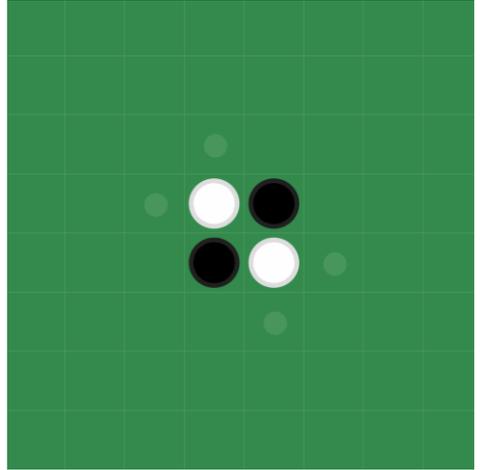
<https://github.com/kurehajime/pera>

# 自己紹介

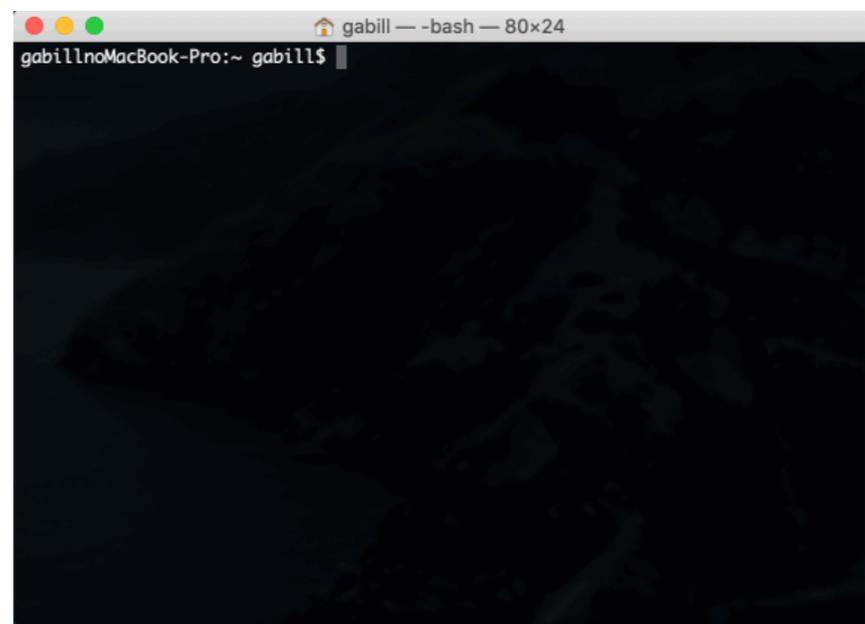
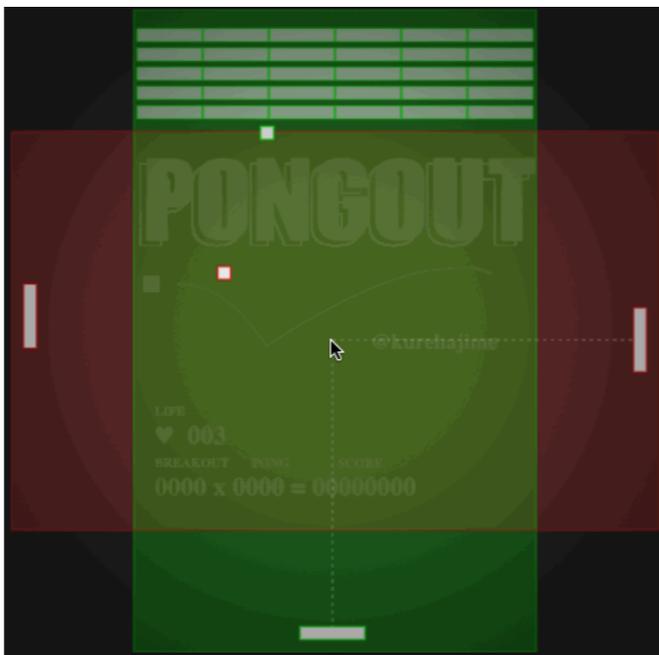
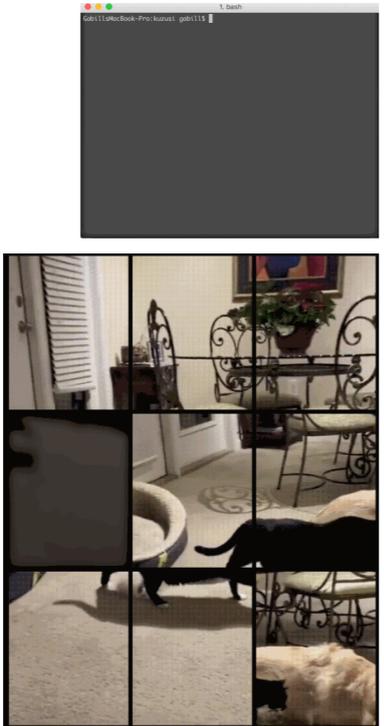
<https://github.com/kurehajime/dajarep>



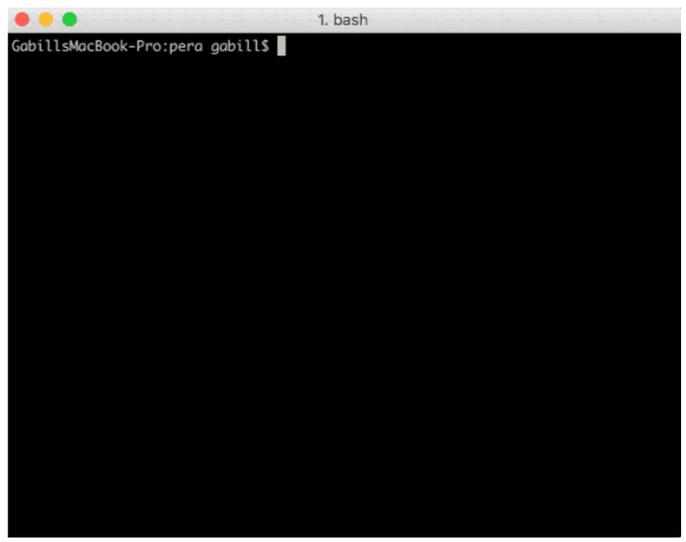
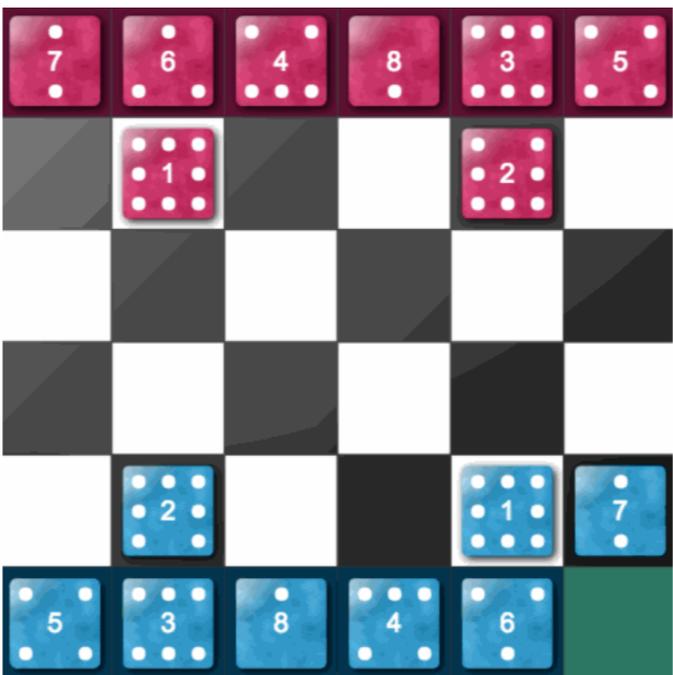
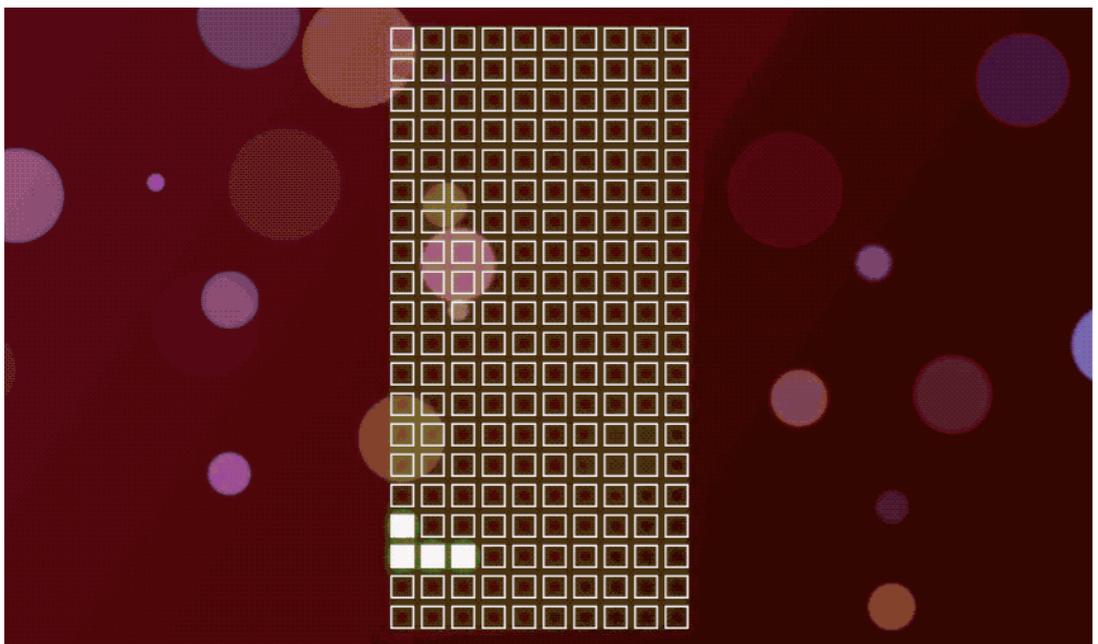
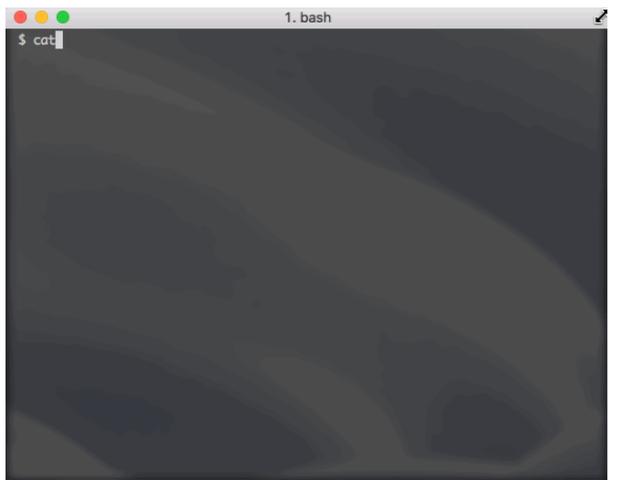
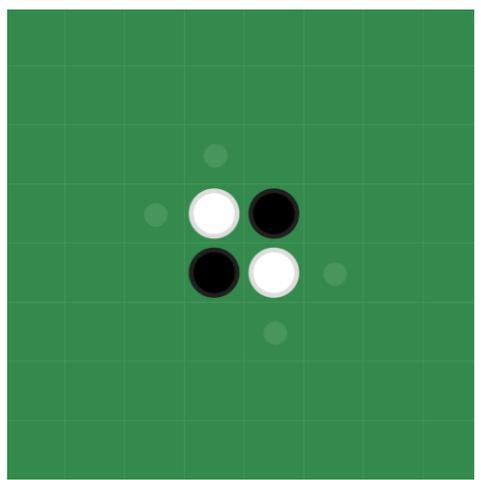
## 与えられた文章から ダジャレを検出できるコマンド 『dajarep』



# 自己紹介



いろいろコマンドラインツールをつかってきましたが...



ほん  
だい

コマンドラインツールは  
なかなか使って貰えない…



**FUKUOKA**

**GO**

**SHI**

**MA**

**CUIのツールを1行で  
Web GUI化する  
ライブラリを作ってみた**

**@ kurehajime**

これは  
ズズ



flagstone

**flagstone**

名

1. 敷石◆床や道路の舗装用の、石やコンクリート。◆【同】 [flag](#)
2. 《地学》板石

# 改 を...

```
package main
```

```
import (  
    "flag"  
    "fmt"
```

標準のflagパッケージ

```
)
```

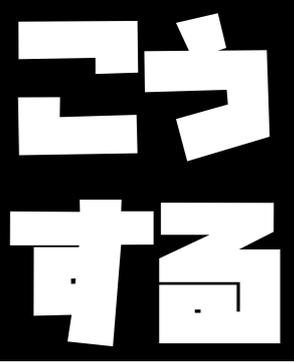
```
var who *string
```

```
func main() {  
    who = flag.String("who", "world", "say hello to ...")  
    flag.Parse()
```

```
    fmt.Println("hello " + *who + "!!")
```

flagの値を取得

```
}
```



```
package main

import (
    "flag"
    "fmt"
    "github.com/kurehajime/flagstone"
)

var who *string

func main() {
    who = flag.String("who", "world", "say hello to ...")
    flag.Parse()

    flagstone.Launch("helloworld", "flagstone sample")

    fmt.Println("hello " + *who + "!")
}
```

(厳密に言えば)

2ZUKUOKA

CUIのツールをX行で

GO

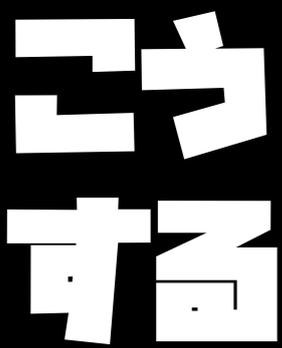
Web GUI化する

SHI

ライブラリを作ってみた

MA

@ kurehajime



```
package main

import (
    "flag"
    "fmt"
    "github.com/kurehajime/flagstone"
)

var who *string

func main() {
    who = flag.String("who", "world", "say hello to ...")
    flag.Parse()

    flagstone.Launch("helloworld", "flagstone sample")

    fmt.Println("hello " + *who + "!")
}
```



helloworld

localhost:8081/x/4b10346bdda201d952e79c7ab4daf23a7ddb9fc9c9c2e60ae0...

# helloworld

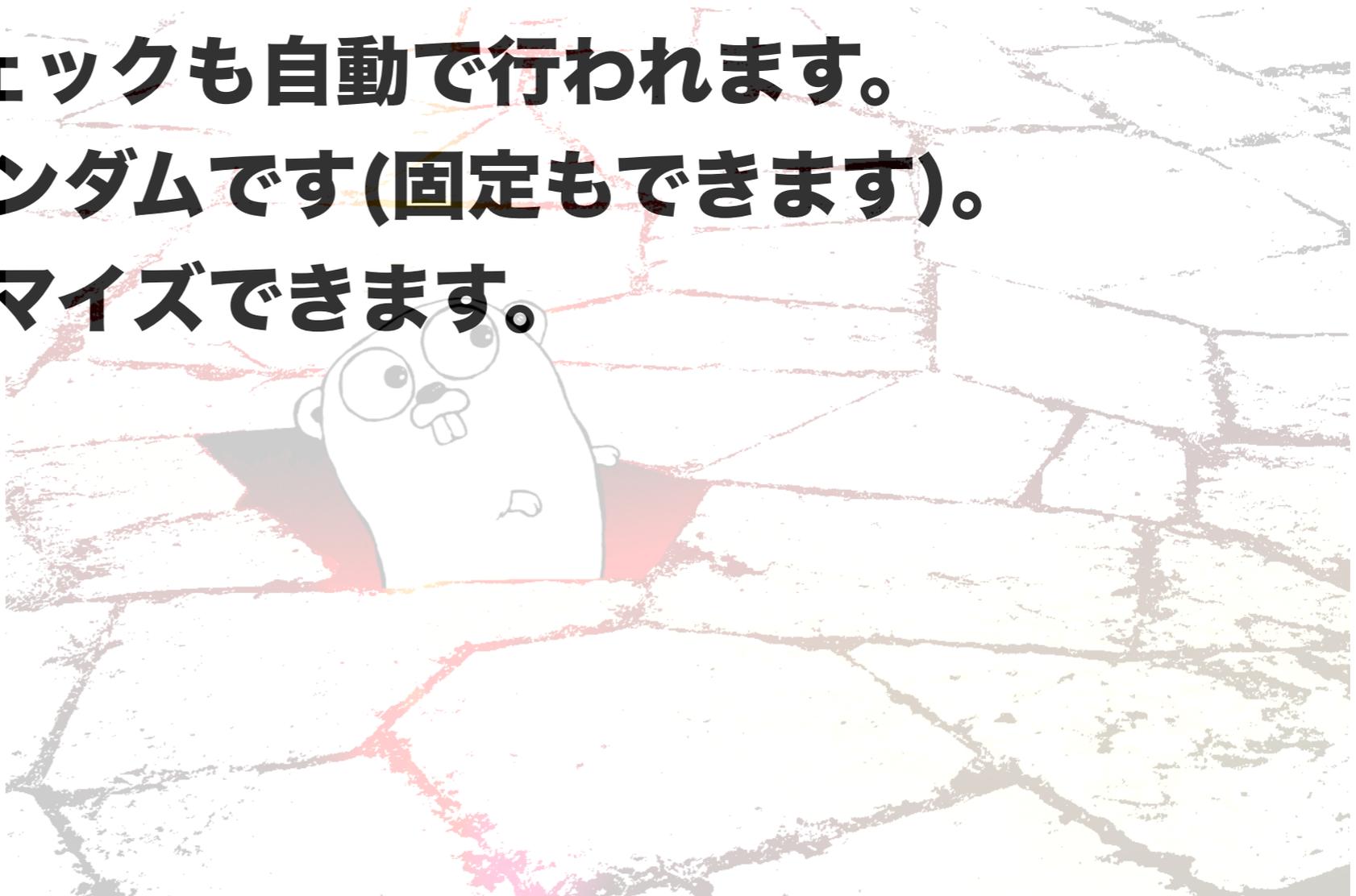
flagstone sample

Run

who world say hello to ...

# 出来る事

- **Webで動くのでクロスプラットフォームです。**
- **デフォルト値も反映されます。**
- **入力値の型チェックも自動で行われます。**
- **URLは毎回ランダムです(固定もできます)。**
- **CSSはカスタマイズできます。**

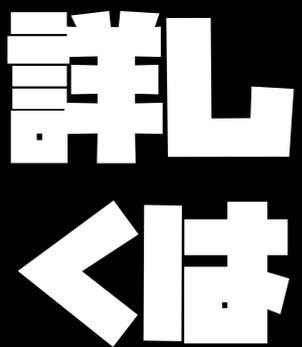


## flag

Go言語のflagパッケージは  
`flag.CommandLine.VisitAll`でflagをリストアップし  
たり  
`f.Value.Set('hoge')`  
で値のセットができます。  
flagstoneはこれを使って実  
現しています。

## gowut

gowutはgo言語の文法で  
Webアプリケーションを作成  
できるライブラリです。HTML  
を書かずにすべてGo言語でレ  
イアウトからイベントまで記述  
できます。もちろんHTMLや  
CSS/Javascriptを加えるこ  
とも可能です。



**GitHub**

<https://github.com/kurehajime/flagstone>

**Qiita**

<https://qiita.com/kurehajime/items/3ff4e6a5f6244aafa736>



ご視  
聴面

りが  
とウ

ごギ  
いま

した  
!!

