

function



GO



golang : function

```
package main

func main() {
}

func f() {
}

func fWithReturn(i int) int {
    return i
}

func fWithMultipleReturn(i int, s string) (int, error) {
    return i, nil
}
```



golang : function

create main.go in folder chapter6-1 :

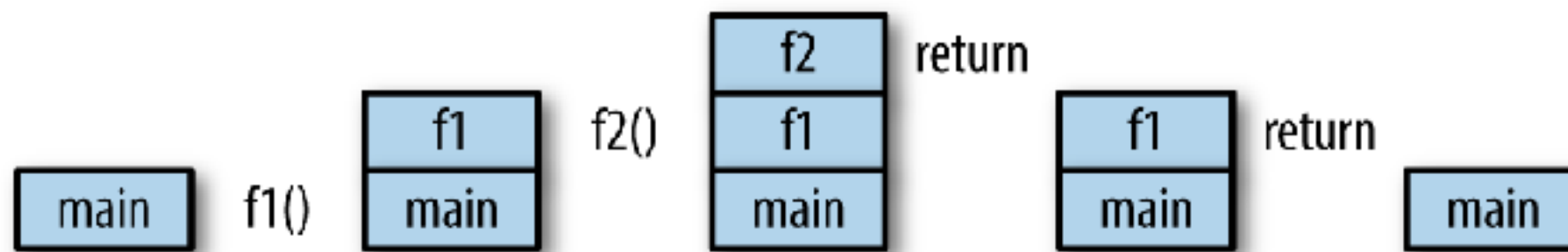
```
package main

import "fmt"

func main() {
    fmt.Println(f1())
}

func f1() int {
    return f2()
}

func f2() int {
    return 1
}
```



golang : function

create main.go in folder chapter6-2 :

```
package main

import "fmt"

func main() {
    fmt.Println(f2())
}

func f2() (r int) {
    r = 1
    return
}
```

Return types can have names



golang : function

create main.go in folder chapter6-3 :

```
package main

import "fmt"

func main() {
    x, y := f()
    fmt.Println(x, y)
}

func f() (int, int) {
    return 5, 7
}
```

Multiple values can be returned



golang : function

create main.go in folder chapter6-4 :

```
package main

import "fmt"

func main() {
    fmt.Println(add(1,2,3))

    xs := []int{1,2,3}
    fmt.Println(add(xs...))
}

func add(args ...int) int {
    total := 0
    for _, v := range args {
        total += v
    }
    return total
}
```

Variadic Function



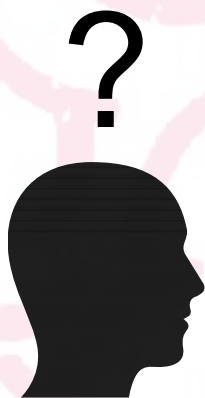
Exercise

create exercise.go in folder chapter6-5 :

REFACTOR FIZZBUZZ ใน CHAPTER5-2

ให้เรียกใช้งาน FUNCTION

แทนที่จะทำงานทุกอย่างใน FUNC MAIN



1



2



3



4

run -> no error -> push to your git repository

