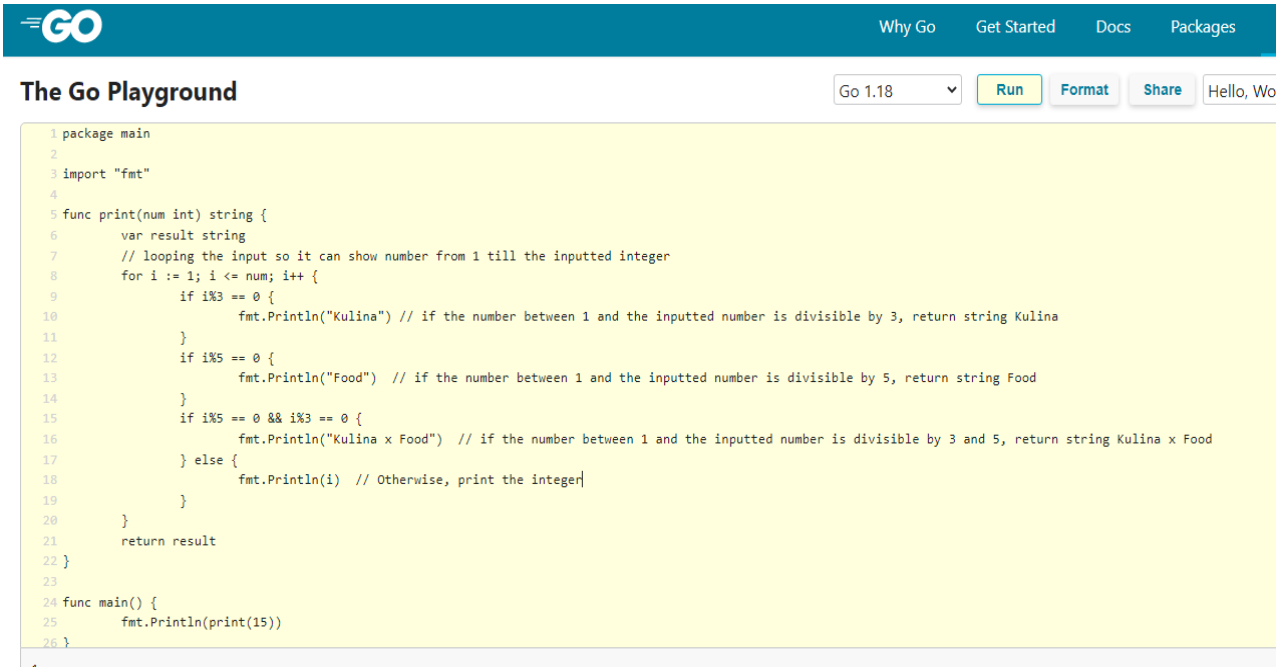
Answer for Kulina x Food



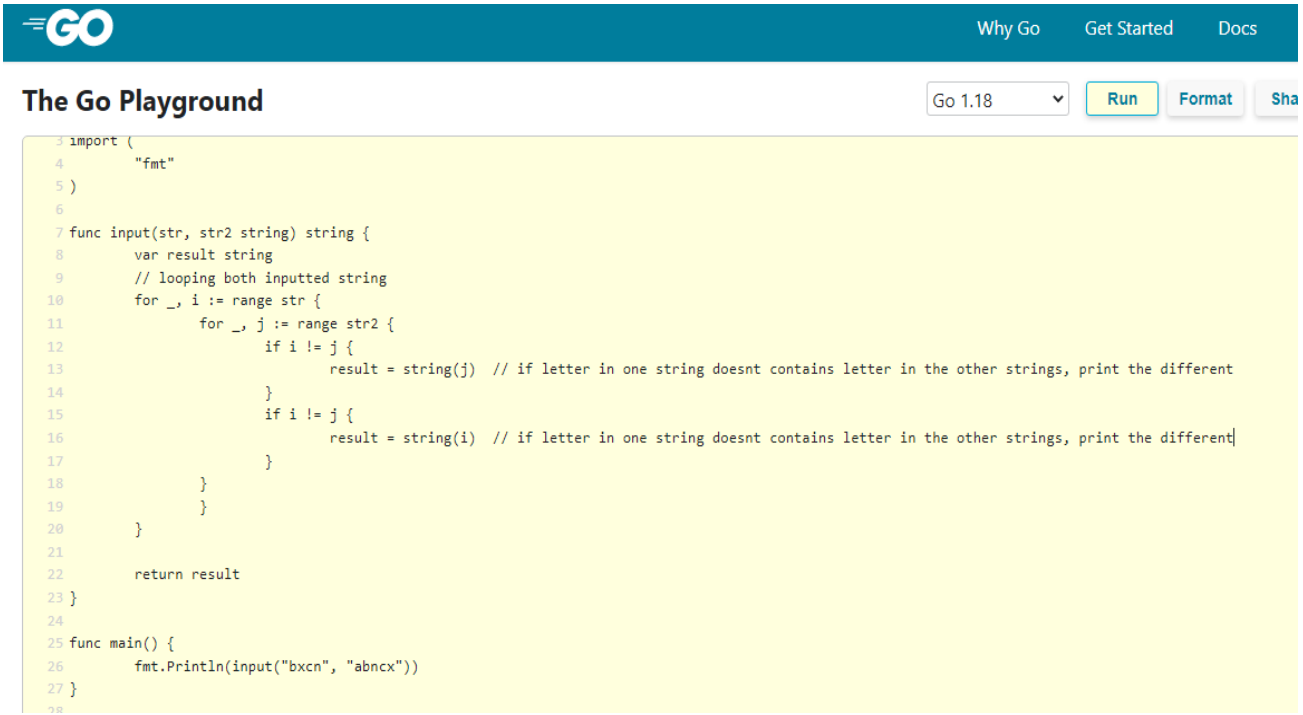

```go
package main

import "fmt"

func print(num int) string {
	var result string
	// looping the input so it can show number from 1 till the inputted integer
	for i := 1; i <= num; i++ {
		if i%3 == 0 {
			fmt.Println("Kulina") // if the number between 1 and the inputted number is divisible by 3, return string Kulina
		}
		if i%5 == 0 {
			fmt.Println("Food")  // if the number between 1 and the inputted number is divisible by 5, return string Food
		}
		if i%5 == 0 && i%3 == 0 {
			fmt.Println("Kulina x Food")  // if the number between 1 and the inputted number is divisible by 3 and 5, return string Kulina x Food
		} else {
			fmt.Println(i)  // Otherwise, print the integer
		}
	}
	return result
}

func main() {
	fmt.Println(print(15))
}
```

Answer for Find the dissimilarity



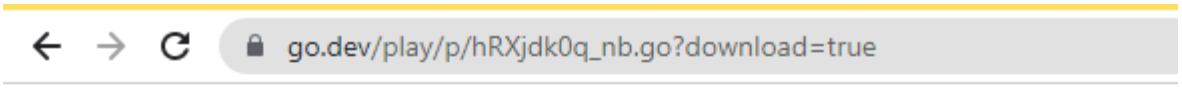

```go
import (
	"fmt"
)

func input(str, str2 string) string {
	var result string
	// looping both inputted string
	for _, i := range str {
		for _, j := range str2 {
			if i != j {
				result = string(j)  // if letter in one string doesnt contains letter in the other strings, print the different
			}
			if i != j {
				result = string(i)  // if letter in one string doesnt contains letter in the other strings, print the different
			}
		}
		}
	}

	return result
}

func main() {
	fmt.Println(input("bxcn", "abncx"))
}
```

Answer for Roman to Integer




```
package main

import (
        "fmt"
)
// mapping all the list of roman numerals code
var decoder = map[rune]int{
        'I': 1,
        'V': 5,
        'X': 10,
        'L': 50,
        'C': 100,
        'D': 500,
        'M': 1000,
}

// making function to convert roman to integer

func Roman(roman string) int {
        if len(roman) == 0 {
                return 0
        }
        first := decoder[rune(roman[0])]
        if len(roman) == 1 {
                return first
        }
        next := decoder[rune(roman[1])]
        if next > first {
                return (next - first) + Roman(roman[2:])
        }
        return first + Roman(roman[1:])
}

func main() {
        fmt.Println(Roman("XII"))
}
```