package main

import (

"fmt"

"time"

)

// Función que muestra el menú del restaurante

// Recibe un canal para comunicar el estado de la operación

func mostrarMenu(canal chan string) {

// Simula un tiempo de espera de 1 segundo antes de mostrar el menú

time.Sleep(1 \* time.Second)

// Contenido del menú con las opciones disponibles

menu := `

Bienvenido al restaurante!

1. lentejas

2. arroz paisa

3. sopa

4. bandeja paisa

5. Bebidas:

1. Jugo de naranja

2. Agua

3. Gaseosa de manzana

6. Postres:

1. Flan

2. Helado

3. pastel de chocolate

7. Salir

`

// Imprime el menú en la consola

fmt.Println(menu)

// Envía un mensaje a través del canal indicando que el menú ha sido mostrado

canal <- "Menú mostrado. Elige una opción (1-7)."

}

// Función que toma el pedido del usuario

// Recibe un canal para enviar la respuesta del pedido

func tomarPedido(canal chan string) {

// Simula un tiempo de espera de 2 segundos antes de tomar el pedido

time.Sleep(2 \* time.Second)

// Variable para almacenar la opción elegida por el usuario

var seleccion int

// Solicita al usuario ingresar su elección del menú

fmt.Println("Ingresa el número de tu elección:")

fmt.Scanln(&seleccion) // Lee la entrada del usuario

// Variable para almacenar la respuesta final según la elección

var respuesta string

// Dependiendo de la opción elegida, se asigna una respuesta

switch seleccion {

case 1:

respuesta = "Has elegido lentejas."

case 2:

respuesta = "Has elegido arroz paisa."

case 3:

respuesta = "Has elegido sopa."

case 4:

respuesta = "Has elegido Bandeja paisa."

case 5:

// Si se elige la opción de bebidas, se solicita una bebida específica

var bebida int

fmt.Println("Elige una bebida: 1. Jugo de naranja, 2. Agua, 3. Gaseosa de manzana")

fmt.Scanln(&bebida) // Lee la entrada de la bebida

// Dependiendo de la opción elegida, se asigna una respuesta para la bebida

switch bebida {

case 1:

respuesta = "Has elegido Jugo de naranja."

case 2:

respuesta = "Has elegido Agua de piña."

case 3:

respuesta = "Has elegido Gaseosa de manzana."

default:

// Si la opción no es válida, se asigna un mensaje de error

respuesta = "Opción de bebida no válida."

}

case 6:

// Si se elige la opción de postres, se solicita un postre específico

var postre int

fmt.Println("Elige un postre: 1. arroz con leche, 2. helado, 3. pastel de chocolate")

fmt.Scanln(&postre) // Lee la entrada del postre

// Dependiendo de la opción elegida, se asigna una respuesta para el postre

switch postre {

case 1:

respuesta = "Has elegido Flan."

case 2:

respuesta = "Has elegido Helado."

case 3:

respuesta = " Has elegido pastel de chocolate"

default:

// Si la opción no es válida, se asigna un mensaje de error

respuesta = "Opción de postre no válida."

}

case 7:

// Si el usuario elige salir, se asigna una respuesta de despedida

respuesta = "Has decidido salir. ¡Gracias por visitarnos!"

default:

// Si la opción no es válida, se asigna un mensaje de error

respuesta = "Opción no válida."

}

// Envía la respuesta final al canal

canal <- respuesta

}

func main() {

// Crea un canal de comunicación de tipo string

canal := make(chan string)

// Llama a la función mostrarMenu en una goroutine para que se ejecute de forma concurrente

go mostrarMenu(canal)

// Llama a la función tomarPedido en una goroutine para que se ejecute de forma concurrente

go tomarPedido(canal)

// Lee los mensajes del canal y los imprime en la consola

// En este caso, esperamos dos mensajes: uno para el menú mostrado y otro con la respuesta del pedido

for i := 0; i < 2; i++ {

mensaje := <-canal // Recibe un mensaje del canal

fmt.Println(mensaje) // Imprime el mensaje recibido

}

}