

Exam #2

create vending-machine.go in folder exercise :



Vending Machine

Coin: TEN(10), Five(5), TWO(2), ONE(1)
T F TW 0

Item: Soft Drink(18),
Canned Coffee(12),
Drinking Water(7)

Coin Return: returns all inserted money

#Criteria

Unlimited items

Unlimited change

Currently inserted money

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



Exam #2

vending-machine: Test Cases



1. Buy SD(soft drink) with exact change

Insert: T, F, TW, 0

Currently inserted money: 18

Choose: Select SD

Return: SD

2. Start adding change but hit coin return

Insert: T, T, F

Currently inserted money: 25

Choose: Coin Return

Return: T, T, F

3. Buy CC(canned coffee) without exact change

Insert: T, T

Currently inserted money: 20

Choose: Select CC

Return: CC, F, TW, 0

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



Exam #2

vending-machine:



```
func main() {  
    vm := NewVendingMachine(coins, items)  
  
    // Buy SD(soft drink) with exact change  
    vm.InsertCoin("T")  
    vm.InsertCoin("F")  
    vm.InsertCoin("TW")  
    vm.InsertCoin("O")  
    fmt.Println("Inserted Money:", vm.GetInsertedMoney())  
    // 18  
    can := vm.SelectSD()  
    fmt.Println(can) // SD  
    .  
    .  
    .  
}
```

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



Exam #2

vending-machine:



```
func main() {  
    vm := NewVendingMachine(coins, items)
```

```
    .  
    .  
    .  
    // Buy CC(canned coffee) without exact change  
    vm.InsertCoin("T")  
    vm.InsertCoin("T")  
    fmt.Println("Inserted Money:", vm.GetInsertedMoney())  
    // 20  
    can = vm.SelectCC()  
    fmt.Println(can) // CC, F, TW, 0  
    .  
    .  
    .
```

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



Exam #2

vending-machine:



```
func main() {  
    vm := NewVendingMachine(coins, items)
```

```
    .  
    .  
    .  
    // Start adding change but hit coin return  
    vm.InsertCoin("T")  
    vm.InsertCoin("T")  
    vm.InsertCoin("F")  
    fmt.Println("Inserted Money:", vm.GetInsertedMoney())  
    // 25  
    change := vm.CoinReturn()  
    fmt.Println(change) // T, T, F  
}
```

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



Exam #2

vending-machine:



```
func main() {  
    vm := NewVendingMachine(coins, items)
```

```
    // Buy SD(soft drink) with exact change
```

```
    vm.InsertCoin("T")
```

```
    vm.InsertCoin("F")
```

```
    vm.InsertCoin("TW")
```

```
    vm.InsertCoin("O")
```

```
    fmt.Println("Inserted Money:", vm.GetInsertedMoney()) // 18
```

```
    can := vm.SelectSD()
```

```
    fmt.Println(can) // SD
```

```
    // Buy CC(canned coffee) without exact change
```

```
    vm.InsertCoin("T")
```

```
    vm.InsertCoin("T")
```

```
    fmt.Println("Inserted Money:", vm.GetInsertedMoney()) // 20
```

```
    can = vm.SelectCC()
```

```
    fmt.Println(can) // CC, F, TW, O
```

```
    // Start adding change but hit coin return
```

```
    vm.InsertCoin("T")
```

```
    vm.InsertCoin("T")
```

```
    vm.InsertCoin("F")
```

```
    fmt.Println("Inserted Money:", vm.GetInsertedMoney()) // 25
```

```
    change := vm.CoinReturn()
```

```
    fmt.Println(change) // T, T, F
```

```
}
```

