Message Passing

Michael Van Sickle

@vansimke

Overview

What is message passing?

Strategies

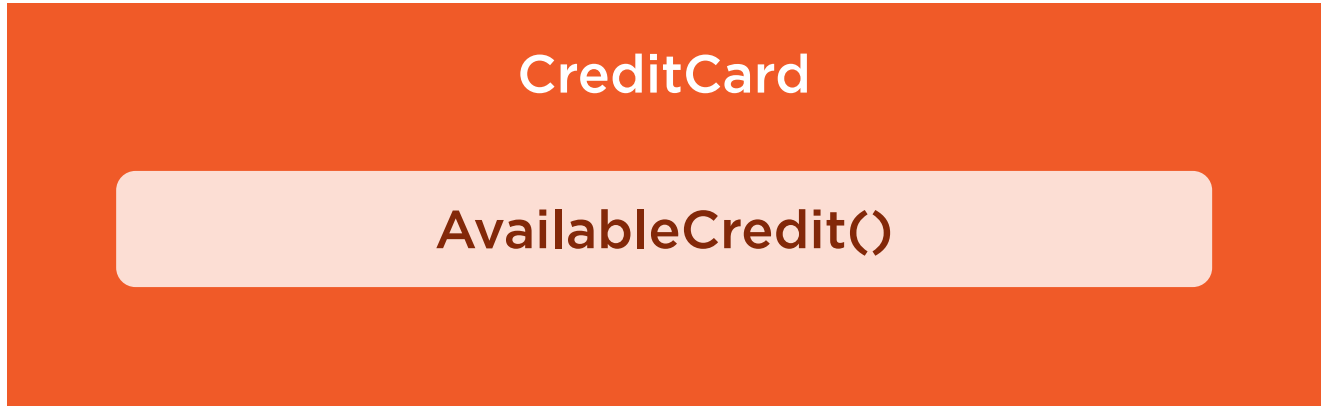
Message Passing

Sending a message to an object, but letting that object determine what to do with it

Conventional Invocation

Message Passing

Message Passing



Encapsulation

Message passing

Strategies

Interfaces

Channels

Interfaces

```
type PaymentOption interface {
    ProcessPayment(float32) bool
}
type CashAccount struct {}
func (c *CashAccount) ProcessPayment(amount float32) bool { ... }
type CreditAccount struct { ... }
func (c *CreditAccount) ProcessPayment(amount float32) bool { ... }
```

Interfaces

```
var paymentOption PaymentOption

paymentOption = &CashAccount{}

ok := paymentOption.ProcessPayment(500)

paymentOption = &CreditAccount{ ... }

ok = paymentOption.ProcessPayment(500)
```

Channels

```
type CreditAccount struct { ... }

func (c *CreditAccount) processPayment(amount float32) { ... }

func CreateCreditAccount(chargeCh chan float32) *CreditAccount {
    creditAccount := &CreditAccount{ ... }
    go func(chargeCh chan float32) {
        for amount := range chargeCh {
            creditAccount.processPayment(amount)
        }
    }(chargeCh)
    return creditAccount
}
```

Channels

```
chargeCh := make(chan float32)

account := CreateCreditAccount(chargeCh)

chargeCh <- 500
```

Summary

What is message passing?

Strategies

- Interfaces
- Channels