Remote

+ Invoke(ICommand): void

ICommand

- Commands: List<ICommand>
- qty: int
- name: string

+ Execute(): void

DrinksCommand

.
.
.
.
.

+ Execute(): void

ProductsCommand

- button: Button
- name: string
- qty: int
- Commands: List<ICommand>
- checkStockHandler:CheckStockHandler
- observerHandler: ObserverHandler
- paymentHandler: PaymentHandler

+ Execute(): void

IHandler

+ SetNext(handler: IHandler): IHandler

Client

+ ClientCode(handler: AbstractHandler, cmd: ICommand): void

AbstractHandler

- _nextHandler: IHandler

+ Handle(cmd: ICommand): object
+ SetNext(handler: IHandler): IHandler

PaymentHandler   3

+ Handle(cmd: ICommand): object

ObserverHandler   2

+ Handle(cmd: ICommand): object

CheckStockHandler   1

+ Handle(cmd: ICommand): object

Client

IComponent

BaseDecerator