Data

PanDeCampo
+ Price:double <<get>> {override}
+ Calorie:uint <<get>> {override}
+ SpecialInstructions:IEnumerable<string> <<get>> {override}

CornDodgers
+ Price:double <<get>> {override}
+ Calorie:uint <<get>> {override}
+ SpecialInstructions:IEnumerable<string> <<get>> {override}

<<enum>> Size
Small
Medium
Large

Order
- lastOrderNumber:uint
- items:List<IOrderItem>
- subtotal:double
+ Items:IEnumerable<IOrderItem> <<get>>
+ Subtotal:double<<get>>
+ OrderNumber:uint<<get>>
+ Add(item:IOrderItem)
+ Remove(item:IOrderItem)
- OnItemPropertyChanged(sender:object, e:PropertyChangedEventArgs)

<<interface>> IOrderItem
+ Price:double<<get>>
+ Calorie:int<<get>>
+ Type:string<<get>>
+ SpecialInstructions:IEnumerable<string><<get>>

Drink | Entree | Side

ChiliCheeseFries
+ Price:double <<get>> {override}
+ Calorie:uint <<get>> {override}
+ SpecialInstructions:IEnumerable<string> <<get>> {override}

<> Side
+ Size:Size <<get,set>> = Size.Small
+ Price:double <<get>> {abstract}
+ Calorie:uint <<get>> {abstract}
+ SpecialInstructions:IEnumerable<string> <<get>> {abstract}
+ Type:string = Side
# NotifyPropertyChanged(propertyName:string)

<<interface>> System.ComponentModel.INotifyPropertyChanged
+ <<event>> PropertyChanged:PropertyChangedEventHandler

<<interface>> System.ComponentModel.INotifyPropertyChanged
+ <<event>> PropertyChanged:PropertyChangedEventHandler

<<enum>> Size
Small,
Medium,
Large

<> Drink
+ Size:Size <<get,set>> = Size.Small
+ Price:double <<get>> {abstract}
+ Calories:uint <<get>> {abstract}
+ Ice:bool <<get,set>> = true
+ SpecialInstructions:List<string> <<get>> {abstract}
+ Type:string = Drink
# NotifyPropertyChanged(propertyName:string)

Menu
+ Entrees:IEnumerable<IOrderItem>
+ Sides:IEnumerable<IOrderItem>
+ Drinks:IEnumerable<IOrderItem>
+ CompleteMenu:IEnumerable<IOrderItem>
+ TypeOfItem:string[]
+ Search(terms:string):IEnumerable<IOrderItem>
+ FilterByCategory(menu:IEnumerable<IOrderItem>, types:IEnumerable<string>):IEnumerable<IOrderItem>
+ FilterByCalories(menu:IEnumerable<IOrderItem>, min:uint?, max:uint?):IEnumerable<IOrderItem>
+ FilterByPrice(menu:IEnumerable<IOrderItem>, min:double?, max:double?):IEnumerable<IOrderItem>

BakedBeans
+ Price:double <<get>> {override}
+ Calorie:uint <<get>> {override}
+ SpecialInstructions:IEnumerable<string> <<get>> {override}

ChiliCheeseFries
+ Price:double <<get>> {override}
+ Calorie:uint <<get>> {override}
+ SpecialInstructions:IEnumerable<string> <<get>> {override}

TexasTea
+ Price:double <<get>> {override}
+ Calories:uint <<get>> {override}
+ Sweet:bool <<get,set>>
+ Lemon:bool <<get,set>>
+ SpecialInstructions:List<string> <<get>> {override}

CowboyCoffee
+ Price:double <<get>> {override}
+ Calories:uint <<get>> {override}
+ Decaf:bool <<get,set>>
+ RoomForCream:bool <<get,set>>
+ SpecialInstructions:List<string> <<get>> {override}

TrailBurger
+ Bun:bool <<get,set>> = true
+ Ketchup:bool <<get,set>> = true
+ Mustard:bool <<get,set>> = true
+ Pickle:bool <<get,set>> = true
+ Cheese:bool <<get,set>> = true
+ Price:double <<get>> = 5.99 {override}
+ Calories:uint <<get>> = 190 {override}
+ SpecialInstructions:IEnumerable<string> <<get>> {override}

RustlersRibs
+ Price:double <<get>> = 7.50 {override}
+ Calories:uint <<get>> = 894 {override}
+ SpecialInstructions:IEnumerable<string> <<get>> {override}

DakotaDoubleBurger
+ Bun:bool <<get,set>> = true
+ Ketchup:bool <<get,set>> = true
+ Mustard:bool <<get,set>> = true
+ Pickle:bool <<get,set>> = true
+ Cheese:bool <<get,set>> = true
+ Tomato:bool <<get,set>> = true
+ Lettuce:bool <<get,set>> = true
+ Mayo:bool <<get,set>> = true
+ Price:double <<get>> = 5.20 {override}
+ Calories:uint <<get>> = 464 {override}
+ SpecialInstrucions:IEnumerable<string> <<get>> {override}

PecosPulledPork
+ Bread:bool <<get,set>> = true
+ Pickle:bool <<get,set>> = true
+ Price:double <<get>> = 5.88 {override}
+ Calories:uint <<get>> = 528 {override}
+ SpecialInstructions:IEnumerable<string> <<get>> {override}

<> Entree
+ Price:double <<get>> {abstract}
+ Calorie:uint <<get>> {abstract}
+ SpecialInstructions:List<string> <<get>> {abstract}
+ Type:string = Entree
# NotifyPropertyChanged(propertyName:string)

TexasTripleBurger
+ Bun:bool <<get,set>> = true
+ Ketchup:bool <<get,set>> = true
+ Mustard:bool <<get,set>> = true
+ Pickle:bool <<get,set>> = true
+ Cheese:bool <<get,set>> = true
+ Tomato:bool <<get,set>> = true
+ Lettuce:bool <<get,set>> = true
+ Mayo:bool <<get,set>> = true
+ Bacon:bool <<get,set>> = true
+ Egg:bool <<get,set>> = true
+ Price:double <<get>> = 6.45 {override}
+ Calories:uint <<get>> = 698 {override}
+ SpecialInstructions:IEnumerable<string> <<get>> {override}

JerkedSoda
+ Price:double <<get>> {override}
+ Calories:uint <<get>> {override}
+ Flavor:SodaFlavor <<get,set>>
+ SpecialInstructions:List<string> <<get>> {override}

Water
+ Price:double <<get>> {override}
+ Calories:uint <<get>> {override}
+ Lemon:bool <<get,set>>
+ SpecialInstructions:List<string> <<get>> {override}

CowpokeChili
+ Cheese:bool <<get,set>> = true
+ SourCream:bool <<get,set>> = true
+ GreenOnions:bool <<get,set>> = true
+ TortillaStrips:bool <<get,set>> = true
+ Price:double <<get>> = 6.10 {override}
+ Calories:uint <<get>> = 171 {override}
+ SpecialInstructions:IEnumerable<string> <<get>> {override}

AngryChicken
+ Bread:bool <<get,set>> = true
+ Pickle:bool <<get,set>> = true
+ Price:double <<get>> = 5.99 {override}
+ Calories:uint <<get>> = 190 {override}
+ SpecialInstructions:IEnumerable<string> <<get>> {override}

<<interface>> System.ComponentModel.INotifyPropertyChanged
+ <<event>> PropertyChanged:PropertyChangedEventHandler

<<enum>> SodaFlavor
CreamSoda,
OrangeSoda,
Sarsaparilla,
BirchBeer,
RootBeer

PointOfSale

CustomizeSide
- RadioButtonClicked(sender:object, e:RoutedEventArgs)

CustomizeWater
- RadioButtonClicked(sender:object, e:RoutedEventArgs)

CustomizeCowboyCoffee
- RadioButtonClicked(sender:object, e:RoutedEventArgs)

CustomizeJerkedSoda
- SizeRadioButtonClicked(sender:object, e:RoutedEventArgs)
- SodaFlavorRadioButtonClicked(sender:object, e:RoutedEventArgs)

CustomizeTexasTea
- RadioButtonClicked(sender:object, e:RoutedEventArgs)

CustomizeTrailBurger

CustomizeTexasTripleBurger

CustomizePecosPulledPork

OrderControl
- drawer:CashDrawer
- OnItemSelectionButtonClicked(sender:object,e:RoutedEventArgs)
- OnCancelOrderButtonClicked(sender:object, e:RoutedEventArgs)
- OnCompleteOrderButtonClicked(sender:object, e:RoutedEventArgs)

MenuItemSelectionControl
- OnItemAddButtonClicked(sender:object, e:RoutedEventArgs)
- AddItemAndOpenCustomizationScreen(item:IOrderItem, screen:FrameworkElement)

OrderSummaryControl
- OnOrderItemClicked(sender:object, e:RoutedEventArgs)
- OpenCustomizationScreen(item:IOrderItem, screen:FrameworkElement)
- OnRemoveButtonClicked(sender:object, e:RoutedEventArgs)

CustomizeDakotaDoubleBurger | CustomizeCowpokeChili | CustomizeAngryChicken | MainWindow | Window

TransactionControl
+ drawer:CashDrawer
+ CalculateTaxAndTotal()
- CreditButtonClicked(sender:object, e:RoutedEventArgs)
- CashButtonClicked(sender:object, e:RoutedEventArgs)
- CancelButtonClicked(sender:object, e:RoutedEventArgs)

CashDrawer

CardTerminal
+ ProcessTransaction(amount:double):ResultCode

UserControl

CustomizePecosPulledPork | CustomizeTexasTripleBurger | CustomizeSide
CustomizeJerkedSoda | CustomizeDakotaDoubleBurger
CustomizeCowboyCoffee | CustomizeTrailBurger
CustomizeAngryChicken | CustomizeCowpokeChili
MenuItemSelectionControl | CustomizeWater | OrderSummaryControl | CustomizeTexasTea

<<enum>> Coins
Penny,
Nickel,
Dime,
Quarter,
HalfDollar,
Dollar

<<enum>> Bills
One,
Two,
Five,
Ten,
Twenty,
Fifty,
Hundred

CashDrawer
+ Pennies {readonly}
+ Nickels {readonly}
+ Dimes {readonly}
+ Quarters {readonly}
+ HalfDollars {readonly}
+ Dollars {readonly}
+ Ones {readonly}
+ Twos {readonly}
+ Fives {readonly}
+ Tens {readonly}
+ Twenties {readonly}
+ Fifties {readonly}
+ Hundreds {readonly}
+ TotalValue{readonly}
+ AddCoin(type:Coins, quantity:int)
+ AddBill(type:Bills, quantity:int)
+ RemoveCoin(type:Coins, quantity:int)
+ RemoveBill(type:Bills, quantity:int)

CashControl
+ drawer:CashDrawer
+ list:IOrderItem[]
- total:double = 0.00
- totalPaid:double = 0.00
- quantity:int = 1
- AddingBillsAndCoinsButtonClicked(sender:object, e:RoutedEventArgs)
- QuantityButtonClicked(sender:object, e:RoutedEventArgs)
- ChangeCalculator(change:int)

<<enum>> ResultCode
Success,
InsufficientFunds,
CancelledCard,
ReadError,
UnknownError

ReceiptPrinter
+ Print(receipt:string)

ReceiptPrinter
+ Print(receipt:string)

OrderControl — CashDrawer

<<exception>> DrawerOverdrawException

Website

About | Error | Page Model | Privacy

Index
+ Menu:IEnumerable<IOrderItem>
+ SearchTerms:string
+ TypeOfItem:string[]
+ CaloriesMin:uint?
+ CaloriesMax:uint?
+ PriceMin:double?
+ PriceMax:double?
+ OnGet()