leakwindBuffet.Data.Enum

Small Medium Large

Enumeration>> odaFlavor

Blackberry Cherry Grapefruit Lemon Peach Watermelon

Bleakwind Buffet. Data. Side

# agonbornWaffleFries

-size: Size = Size.Small

+Size: Size <<get, set>> +PropertyChanged:PropertyChangedEventHandler +Price: double <<get, set>>

+Calories: uint <<get, set>> +SpecialInstructions: List<string> << get, set>>

+Description: string <<get>> +ToString(): string {override}

## riedMiraak

-size: Size = Size.Small

+PropertyChanged:PropertyChangedEventHandler

+Size: Size <<get, set>> +Price: double <<get, set>>

+Calories: uint <<get, set>> +SpecialInstructions: List<string> << get, set>>

+Description: string <<get>> +ToString(): string {override}

# //adOtarGrits

-size: Size = Size.Small +PropertyChanged:PropertyChangedEventHandler +Size: Size <<get, set>>

+Price: double <<get, set>> +Calories: uint <<get, set>> +SpecialInstructions: List<string> <<get, set>>

+Description: string <<get>> +ToString(): string {override}

okunSalad/

-size: Size = Size.Small

+PropertyChanged:PropertyChangedEventHandler +Size: Size <<get, set>>

+Price: double <<get, set>> +Calories: uint <<get, set>>

+SpecialInstructions: List<string> << get, set>>

+Description: string <<get>> +ToString(): string {override} leakwindBuffet.Data.Entrees

## arden Orc Omelette

-broccoli: bool = true -mushrooms: bool = true

-tomato: bool = true -cheddar: bool = true

+PropertyChanged:PropertyChangedEventHandler +Broccoli: bool <<get, set>>

+Mushrooms: bool <<get, set>> +Tomato: bool <<get, set>> +Cheddar: bool <<get, set>> +Price: double <<get, set>>

+Calories: uint <<get, set>> +SpecialInstructions: List<string> << get, set>> +Description: string <<get>>

+ToString(): string {override}

nokehouseSkeleton

-sausageLink: bool = true

-hashBrowns: bool = true

+SausageLink: bool <<get, set>>

+HashBrowns: bool <<get, set>>

+Pancake: bool <<get, set>>

+Price: double <<get, set>>

+Calories: uint <<get, set>>

+Description: string <<get>>

+ToString(): string {override}

+Price: double <<get, set>>

+Calories: uint <<get, set>>

+Description: string <<get>>

+ToString(): string {override}

nugsTBone

-pancake: bool = true

+Egg: bool <<get, set>>

-egg: bool = true

### PhillyPoacher

-sirloin: bool = true -onion: bool = true -roll : bool = true +PropertyChanged:PropertyChangedEventHandler +Sirloin: bool <<get, set>> +Onion: bool << get, set>>

+Roll: bool <<get, set>> +Price: double <<get, set>> +Calories: uint <<get, set>>

+SpecialInstructions: List<string> << get, set>> +Description: string <<get>> +ToString(): string {override}

+PropertyChanged:PropertyChangedEventHandler

+SpecialInstructions: List<string> << get, set>>

+PropertyChanged:PropertyChangedEventHandler

+SpecialInstructions: List<string> << get, set>>

+Ketchup: bool <<get, set>> +Mustard: bool <<get, set>> +Pickle: bool <<get, set>>

+Tomato: bool <<get, set>> +Lettuce: bool <<get, set>>

+Price: double <<get, set>> +Calories: uint <<get, set>>

+ToString(): string {override}

# halmorTriple

-ketchup: bool = true -mustard: bool = true -pickle: bool = true -cheese: bool = true -tomato: bool = true -lettuce: bool = true -mayo: bool = true -bacon: bool = true -egg: bool = true +Bun: bool <<get, set>> +Ketchup: bool <<get, set>> +Mustard: bool <<get, set>> +Pickle: bool <<get, set>> +Cheese: bool << get, set>> +Tomato: bool <<get, set>> +Lettuce: bool <<get, set>> +Mayo: bool <<get, set>> +Bacon: bool <<get, set>> +Egg: bool <<get, set>> +Price: double <<get, set>>

+Description: string <<get>> +ToString(): string {override}

### Interface>> lOrderItem

+Price: double <<get, set>> +Calories: uint <<get, set>> +SpecialInstructions: List<string><<get, set>> +Description: string <<get>>

+Size: BleakwindBuffet.Data.Enums.Size <<get,set>> +Price: double <<get, set>>

+SpecialInstructions: List<string><<get, set>>

+Size: BleakwindBuffet.Data.Enums.Size <<get,set>>

+SpecialInstructions: List<string><<get, set>>

+Calories: uint <<get, set>>

+Calories: uint <<get, set>> +Description: string <<get>> +SpecialInstructions: List<string> <<get, set>> +Description: string <<get>>

-bun: bool = true -ketchup: bool = true -mustard: bool = true -pickle: bool = true -cheese: bool = true -tomato: bool = true -lettuce: bool = true -mayo: bool = true +PropertyChanged:PropertyChangedEventHandler +Bun: bool <<get, set>>

+PropertyChanged:PropertyChangedEventHandler

+Cheese: bool << get, set>> +Mayo: bool <<get, set>>

+SpecialInstructions: List<string> << get, set>>

+Description: string <<get>>

-bun: bool = true +PropertyChanged:PropertyChangedEventHandler +Calories: uint <<get, set>> +SpecialInstructions: List<string> << get, set>>

### Bleakwind Buffet. Data. Interface

+SpecialInstructions: List<string><<get, set>>

+Price: double <<get, set>>

+Calories: uint <<get, set>>

+Description: string <<get>>

+Price: double <<get, set>>

+Calories: uint <<get, set>>

+Description: string <<get>>

+ToString(): string {override}

## oubleDraugr

BriarheartBurger

-bun: bool = true

-ketchup: bool = true

-mustard: bool = true

-pickle: bool = true

-cheese: bool = true

+Bun: bool <<get, set>>

+Ketchup: bool <<get, set>>

+Mustard: bool <<get, set>>

+Pickle: bool <<get, set>>

+Cheese: bool << get, set>>

+Price: double <<get, set>>

+Entrees(): IEnumerable<IOrderItem> {static} +Sides(): IEnumerable<IOrderItem> {static} +Drinks(): IEnumerable<IOrderItem> {static} +FullMenu(): IEnumerable<IOrderItem> {static} +Description: string <<get>>

### leakwindBuffet.Data.Sides

## retinoAppleJuice

-size: Size = Size.Small

-price: double = 0.62

+Price: double <<get>>

+Calories: uint <<get>>

CandleHearthCoffee

-size: Size = Size.Small

-ice: bool = false

-decaf: bool = false

-price: double = 0.75

+Size: Size << get, set>>

+Ice: bool <<get, set>>

+Price: double <<get>>

+Calories: uint <<get>>

MarkarthMilk

-size: Size = Size.Small

-price: double = 1.05

-calories: uint = 56

+Size: Size <<get, set>>

+lce: bool <<get, set>>

+Price: double <<get>>

+Calories: uint <<get>>

-size: Size = Size.Small

-ice: bool = false

Soda Flavor. Cherry

-price: double = 1.42

+Size: Size <<get, set>>

+lce: bool <<get, set>>

+Price: double <<get>>

+Calories: uint <<get>>

+ToString(): string {override}

-specialinstructions = new List<string>

+PropertyChanged:PropertyChangedEventHandler

+SpecialInstructions: List<string> << get, set>>

-calories: uint = 117

<<get, set>>

WarriorWater

-size: Size = Size.Small

-lemon: bool = false

-price: double = 0

-calories: uint = 0

+Lemon: bool <<get, set>>

+Description: string <<get>> +ToString(): string {override}

+Size: Size <<get, set>> +Ice: bool <<get, set>> +Price: double <<get>> +Calories: uint <<get>>

-ice: bool = false

SailorSoda

+Description: string <<get>>

+ToString(): string {override}

-ice: bool = false

+ToString(): string {override}

-specialinstructions = new List<string>

+PropertyChanged:PropertyChangedEventHandler

+SpecialInstructions: List<string> <<get, set>>

-flavor: BleakwindBuffet.Data.Enums.SodaFlavor =

+PropertyChanged:PropertyChangedEventHandler

+Flavor: BleakwindBuffet.Data.Enums.SodaFlavor

+SpecialInstructions: List<string> << get, set>>

-specialinstructions = new List<string>

-calories: uint = 7

+Description: string <<get>>

+ToString(): string {override}

-roomForCream: bool = false

-specialinstructions = new List<string>

+RoomForCream: bool <<get, set>>

+PropertyChanged:PropertyChangedEventHandler

+SpecialInstructions: List<string> << get, set>>

-ice: bool = false -specialinstructions = new List<string>

+SpecialInstructions: List<string> << get, set>>

-calories: uint = 44 +PropertyChanged:PropertyChangedEventHandler

+Size: Size << get, set>> +Ice: bool <<get, set>>

# rivacy.cshtml

leakwind Buffet. Website

ndex.cshtml

dex.cshtml.cs

+Items:IEnumerable<IOrderItem> <<get, set>>

+OnGet(string,string[],double,double,double,double: void

rror.cshtml

+SearchTerms: string <<get, set>>

+ItemTypes:string[] <<get, set>>

About.cshtml

### eakwind Buffet. Website

# mFilter.cs {static}

-items: IEnumerable<IOrderItem> {static}

+ItemType: string[] << get>> {static} +All: IEnumerable<IOrderItem><<get>> {static}

-ltemFilter(): constructor {static}

+Search(IEnumerable<IOrderItem>, string): IEnumerable<IOrderItem> {static} +FilterByType(IEnumerable<IOrderItem>, IEnumerable<string>: IEnumerable<IOrderItem> {static}

+FilterByPrice(IEnumerable<IOrderItem>, double, double): IEnumerable<IOrderItem>{static}

+FilterBvCalories(IEnumerable<IOrderItem>, double, double): IEnumerable<IOrderItem> {static}