# Laboratory 11 Designs

**Customer**

- name : String

- street : String

- cityStateZip : String

- phone : String

+ Set(uName : String, uStreet : String, uCityStateZip : String, uPhone : String) : void

Sets the customer attributes with given values passed in.

+ GetName(void) : String {name}

Gives back the name attribute.

+ GetStreet(void) : String {street}

Gives back the street attribute.

+ GetCSZ(void) : String {cityStateZip}

Gives back the cityStateZip attribute.

+ GetPhone(void) : String {phone}

Gives back the phone attribute.

**Pie**

- name : String

- price : double

- quantity : int

+ Load(pieFileSC : Scanner) : void

Reads the name and price attributes from a given scanner.

+ GetPieName(void) : String {name}

Gives back the name attribute.

+ GetPrice(void) : double {price}

Gives back the price attribute.

+ GetQuantity(void) : int {quantity}

Gives back the quantity attribute.

+ IncreaseQuantity(addedQuantity : int) : void

Adds the given addedQuantity to the quantity attribute.

+ GetExtendedPrice(void) : double {extendedPrice}

Computes and returns the extended price (quantity \* price).

+ Reset(void) : void

Sets the quantity attribute to 0.

None.

+ main(args : String [ ]) : void

Loads each pie's information and processes each customer.

+ LoadEachPieInformation(thePies : Pie [ ]) : void

Gets the information for each of the 10 pies from the pie file.

+ ProcessEachCustomer(thePies : Pie [ ]) : void

Repeatedly gets customer information, processes each pie purchase, displays bill, and stores summary.

+ GetCustomerInformation(theCustomer : Customer) : void

Gets the name, address, city/state/zip, and phone number of the customer from the user.

+ ResetPieQuantities(thePies : Pie[ ] ) : void

Resets the quantity of each pie to 0.

+ ProcessEachPiePurchase(thePies : Pie [ ]) : void

Repeatedly get a valid pie number, get a valid pie (added) quantity, and updates the corresponding total.

+ GetValidPieNumber(void) : int {pieNumber}

Gets the number 1 through ten (inclusive) from the user - adjusts by subtracting 1.

+ GetValidPieQuantity(void) : int {addedQuantity}

Gets a quantity to be added to the quantity of pies purchased. Allows only > 0.

+ DisplayBill(thePies : Pie [ ], theCustomer : Customer) : void

Displays a heading, processes each purchase, displays the total due.

+ DisplayHeading(theCustomer : Customer) : void

Displays the customer information and column headings for the report.

+ ProcessEachPurchase(thePies : Pie [ ]) : double {totalDue}

For each pie with a quantity > 0, displays the pie information and updates the total due.

+ DisplayOrderedPie(givenPie : Pie) : void

Displays the name, price, quantity, and extended price of the given pie.

+ UpdateTotalDue(givenPie : Pie, totalDue : double) : double {totalDue}

Increases the total due by the extended price of the given pie.

+ DisplayTotalDue(totalDue : double) : void

Displays the total due.

+ StoreSummary(theCustomer : Customer, thePies : Pie [ ]) : void

Stores the customer information and stores each ordered pies information.

+ StoreCustomerInformation(theCustomer : Customer) : void

Appends the customer name, street, city/state/zip, and phone to the order file.

+ StoreEachOrderedPieInformation(thePies : Pie [ ]) : void

For each pie whose quantity > 0, stores name and quantity