



<p>Price</p> <ul style="list-style-type: none"> - regularPrice: float - salePrice: float - cost: final float - onSale: boolean - priceHistory: ArrayList<String> <p> ~ Price (regularPrice: float, cost: float) - priceHistoryAltered (price: float): void ~ getArrayPriceHistory(): ArrayList<String> ~ getPriceHistory (): void ~ putOnSale (salePrice: float, salePeriod: String): void ~ putOnSaleReadIn (salePrice: float, salePeriod: String): void ~ takeOffSale (): void ~ changeRegularPrice (price: float): void ~ changeSalePrice (price: float): void ~ changeSalePeriod (newPeriod: String): void ~ isOnSale (): boolean ~ getRegularPrice (): float ~ getSalePrice (): float ~ getSalePeriod (): String ~ getCost (): float ~ addToPriceHistory (text: String): void </p>	<p>Location</p> <ul style="list-style-type: none"> - aisle: String - subsection: String - section: String <p> ~ Location (aisle: String, subsection: String, section: String) ~ locationDisplay (): void ~ getLocationInfo (): String + setSubsection (): void + setAisle (): void </p> <p>DailyOperations</p> <p> ~ revenue: HashMap<String,Float> ~ costs: HashMap<String,Float> ~ costs: HashMap<String,Float> </p> <p> ~ addToDailyRevenue (rev: float): void ~ subtractFromDailyRevenue (rev: float): void ~ addToDailyCost (cost: float): void ~ subtractFromDailyCost (cost: float): void </p>	<p>FileLoader</p> <p> ~ loadInFiles (): void ~ loadProducts (): void ~ readInEmployees (): void ~ readInMoney (): void ~ readInSaleProducts (): void ~ readInOrders (): void ~ receiveProductHelper (aProduct: String): void </p> <p>FileWriter</p> <ul style="list-style-type: none"> - path: String - path2: String - path3: String - path4: String - path5: String - append: boolean - textLine: String <p> - writeToFile (): void - writeToOrder (): void - writeToMemo (): void - writeToFileEmployees (): void - writeToOrder (): void </p>
<p>LoggerHelper</p> <p>~ logger: Logger</p> <p> ~ makeLog (logMessage: String): void ~ addToOrderLog (logMessage: String): void </p>	<p>SystemHandler</p> <p> ~ isInteger (str: String): boolean + main (args: String[]): void </p>	<p>PrintDisplay</p> <p> + printUser (user: String): void + printMemos (user: String): void </p>

Product	order	Inventory
<ul style="list-style-type: none"> - productName: final String - distributor: String - upc: final int - quantity: int - threshold: int - shortage: boolean - location: Location ~ price: Price 	<ul style="list-style-type: none"> - upc: final int - quantity: final int - orderNumber: int - processed: boolean - productName: final String - distributorName: final String - cost: float 	<ul style="list-style-type: none"> ~ stock: ArrayList ~ orderHistory: ArrayList ~ currentOrderNum: int
<ul style="list-style-type: none"> ~ Product (productName: String, location: Location, upc: int, quantity: int, price: Price, threshold: int, Distributor: String) ~ changeThreshold (amount: Int): void ~ getQuantity (): int ~ getThreshold (): int ~ getUpc (): int ~ getProductName (): String - isAShortage (): boolean ~ getDistributor (): String ~ increaseQuantity (amount: int): void ~ decreaseQuantity (amount: int): void ~ getProductInformation (): String + getLocation (): Location 	<ul style="list-style-type: none"> + Order (orderNumber: int, upc: int, quantity: int, processed: boolean, productName: String, distributorName: String, cost: float) ~ isProcessed (processed: boolean): boolean ~ setProcessed (processed: boolean): void ~ getOrderNumber (): int ~ getQuantity (): int ~ getUpc (): int ~ getDistributorName (): String ~ getProductName (): String ~ getCost (): float + toString (): String 	<ul style="list-style-type: none"> ~ getStock (): ArrayList ~ getOrderHistory (): ArrayList ~ productInInventory (upc: int): boolean - productLocationStock (upc: int): int ~ getProduct (upc: int): Product ~ orderInOrderHistory (orderNum: int): boolean ~ removeProductFromStock (upc: int): void
		RegisteredEmployees
		<ul style="list-style-type: none"> ~ listOfEmployees: ArrayList<Integer>
		<ul style="list-style-type: none"> ~ loginCheck (ID: Integer): boolean ~ listAllEmployees (): void ~ addEmployee (newEmployee: String): void ~ removeEmployee (fireEmployee: String): void ~ getEmployeeList (): ArrayList<Integer>