



Store

- balance : int
- accounting Balance : int
- suppliers : Supplier[]
- clients : Client[]
- products : Product[]
- sales : Sale[]
- orders : Order[]

- + registerClient (id: String, name: String, adress: String): void
- + showAllClients(): String
- + showClient (id: String): String
- + showClientTransactions (id: String): String
- + toggleProductNotifications (idClient: String, idProduct: String): String
- + changePrice (id: String): String
- + registerProductBox (id: String, price: int, minimumStock: int, idSupplier: String, serviceType: Service): void
- + registerProductContainer (id: String, price: int, minimumStock: int, idSupplier: String, serviceType: Service, serviceQuality: Quality): void
- + showAllProducts(): String
- + registerSupplier (id: String, name: String, adress: String): void
- + toggleTransactions (id: String): String
- + showSupplierTransactions (id: String): String
- + showTransaction (id: int): String
- + registerSale (idClient: String, deadline: int, idProduct: String, amount: int): void
- + registerOrder (idSupplier: String, idProduct: String, amount: int): void
- + pay (id: int): void
- + lookUpProductsUnderTopPrice (limit: int): String
- + lookUpPaymentsByClient (id: String): String
- + showGlobalBalance(): void

Storefront

- filename: String
- store: Store
- date: Date

- + save(): void
- + saveAs(): void
- + load(filename: String): void
- + registerClient(id: String, name: String, address: String): void
- + showAllClients(): String
- + showClient(id: String): String
- + showClientTransactions(id: String): String
- + toggleProductNotifications(idClient: String, idProduct: String): String
- + changePrice(id: String, price: int): void
- + registerProductBook(id: String, title: String, author: String, ISBN: String): void
- + registerProductBox(id: String, price: int, minimumStock: int, idSupplier: String, serviceType: Service, serviceQuality: Quality): void
- + showAllProducts(): String
- + registerSupplier(id: String, name: String, address: String): void
- + toggleTransactions(id: String): String
- + showGlobalBalance(): void
- + showSupplierTransactions(id: String): String
- + showTransaction(id: int): String
- + registerSale(idClient: String, deadline: int, idProduct: String, amount: int): void
- + registerOrder(idSupplier: String, idProduct: String, amount: int): void
- + pay(id: int): void
- + lookupProductsUnderTopPrice(limit: int): String
- + lookupPaymentsByClient(id: String): String
- + displayDate(): String
- + advanceDate(days: int): void

Product

-_supplier: Supplier
 -_price: int
 -_minimumStock: int
 -_stock: int
 -_id: String

+ changeProductPrice (price: int): void

Client

-_id: String
 -_name: String
 -_adress: String
 -_purchases: int[]
 -_rank: Rank

+ purchaseHistory(): void
 + activateNotifications (id: String): void
 + deactivateNotifications (id: String): void

Box

-_volume: int
 -_limit: int
 -_service: Service

Container

-_volume: int
 -_limit: int
 -_service: Service
 -_quality: Quality

Book

-_title: String
 -_author: String
 -_ISBN: String

Supplier

-_int: String
 -_name: String
 -_adres: String
 -_transactionHistory: int[]
 -_available: boolean

+ accessTransactionHistory(): void
 + enable(): void
 + disable(): void

<<abstract>>

Transaction

#_id: int
 -_type: String

Date

-_date: int
 + advanceDate (days: int): void

Notification

-_description: String
 -_delivery: DeliveryType
 -_clients: String[]

+ sendNotifications(): void

DeliveryType

-_delivery: String

Rank
- rank: String
- points: int
+ calculatePoints(m: int): void
+ promote(): void
+ demote(): void

Service
- service: String

Quality
- quality: String

Order
- supplier: Supplier
- totalPrice: int
- products: String[]
+ updateStock(store: Store): void
+ payment(store: Store): void

Sole
- supplier: Supplier
- products: String[]
- deadline: int
+ payment(client: Client): void

exceptions

Bad Entry Exception
- entrySpecification: String

Import File Exception

Missing File Association Exception

Unavailable File Exception
~ filename: String

Declaro por minha honra que este diagrama foi realizado apenas pelos elementos que constituem o grupo de projeto. (Grupo 53)

Tamós Pereira

Declaro por minha honra que este diagrama foi realizado apenas pelos elementos que constituem o grupo do projeto (grupo 53).

Mania Campos