

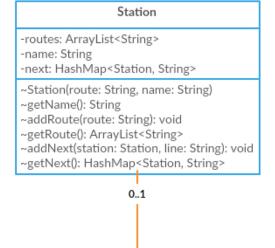
LoadValuePayment

- creditCardNumber: String
- cvv: String
- creditCardHolder: String
- value: double
- isPending: boolean
- isAccept: boolean
- transactionId: String
- time: LocalDateTime
- + Payment(value:double, creditCardHolder: String,cvv:String,creditCardNumber: String)
- + toString(): String
- + getTime(): LocalDateTime
- + setPendingPayment(pendingPayment: ArrayList<Payment>): void

confirm(): void

reject(): void

- search(transactionId: String): Payment
- confirmFinalize(): void
- rejectFinalize(): void
- ~ confirmId(transactionId: String): void
- ~ declineId(transactionId: String): void
- ~ getValue(): double



RouteFinder

1...*

- -stations: HashMap<String, Station>
- +RouteFinder(type: String)
- -stationGenerator(type: String): HashMap<String, Station>
- -getTreePath(start: String, end: String): ArrayList
- -getShortestPath(start: String, end: String): String[]
- -getOptimalPath(start: String, end: String): String [][]
- +getShortestPathLength(start: String, end: String): int
- +getRoutePlan(start: String, end: String): String

<<abstract>> Payment

- ~ logger: Logger {readOnly}
- pendingPayment: ArrayList<Payment>
- creditCardNumber: String
- cvv: String
- creditCardHolder: String
- value: double
- isPending: boolean
- isAccept: boolean
- time: LocalDateTime
- + Payment(value:double, creditCardHolder: String,cvv:String,creditCardNumber: String)
- + Payment(value:double, creditCardHolder: String,cvv:String,creditCardNumber: String)
- + toString(): String
- + getTime(): LocalDateTime
- + setPendingPayment(pendingPayment: ArrayList<Payment>): void
- + confirm(): void
- + reject(): void
- confirmFinalize(): void
- rejectFinalize(): void
- search(transactionId: String): Payment
- ~ confirmId(transactionId: String): void
- ~ declineId(transactionId: String): void

BuyCardPayment (This class is also associated with TransitUserAccount.)

- acceptedBuyCardPayment: ArrayList<BuyCardPayment>
- cardNumber: String
- account: TransitUserAccount
- ${\tt ~BuyCardPayment(cvv:String,creditCardNumber: String, creditCardHolder: String, account: TransitUserAccount)} \\ {\tt +setAcceptedBuyCardPayment(payment: ArrayList<BuyCardPayment>): void}$
- +confirm(): void
- +reject(): void
- +toString(): String

0..1

DataGenerator (This class is also associated with Card class)

- allCards: HashMap<String, Card>
- -acceptedBuyCardPayment: ArrayList<BuyCardPayment>
- +DataGenerator()
- + get Daily Bus Stops Reached (date: Local Date): int
- +getDailySubwayStationsReached(date: LocalDate): int
- +getDailyPassRevenue(date: LocalDate): double
- +getDailyFareRevenue(date: LocalDate): double
- +getDailyCardRevenue(date: LocalDate): double
- +getMonthlyPassRevenue(year: int, month: int): double +getMonlyFareRevenue(year: int, month: int): double
- +getMonthlyCardRevenue(year: int, month: int): double
- +getMonthlyBusStopsReached(year: int, month: int): int
- +getMonthlySubwayStationsReached(year: int, month: int): int

DataSource

(This class is associated with Card, TransitUserAccount, AdminUserAccount, BuyCardPayment and Payment classes)

-dataSource: DataSource

- -allCards: HashMap<String, Card>
- allTransitUserAccounts: HashMap<String, TransitUserAccount>
- -adminUserAccount: AdminUserAccount
- -acceptedBuyCardPayment: ArrayList<BuyCardPayment>
- -pendingPayment: ArrayList<Payment>
- -fareInfo: HashMap<String, Double>

-DataSource(filePath: String)

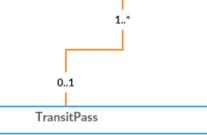
- +getDataSource: DataSource
- +saveToFile(filePath: String): void
- +getAllCards(): HashMap<String, Card>
- +getPendingPayment(): ArrayList<Payment>
- +getAcceptedBuyCardPayment(): ArrayList<BuyCardPayment>
- +getAdminUserAccount(): AdminUserAccount
- +getAllTransitUserAccounts(): HashMap<String, TransitUserAccount>
- +getFareInfo(): HashMap<String, Double>
- -initNew(): void
- -readFromFile(filePath: String): void
- -saveStations(filePath: String): void
- -saveFares(): void

AdminUserAccount (This class is associated with User class)

- defaultAdmin: AdminUserAccount
- accountHolder: User
- logger: Logger {readOnly}
- AdminUserAccount(accountHolder: User)
- + handleAdminlogin(email: String, plainPassword: String): AdminUserAccount
- + handleTransaction(transaction: String, approve: boolean): void
- + handleChangePassword(oldPassword: String, newPassword: String): void
- + handleChangePrice(fareToChange: Fares, newValue: double): void
- + setDefaultAdmin(defaultAdmin: AdminUserAccount): void
- getAccountHolder(): User
- ~ getDefaultAdmin(): AdminUserAccount

TransitUserAccount (This class is associated with TransitUser and Card classes)

- accountHolder: TransitUser
- allTransitUserAccount: HashMap<String, TransitUserAccount>
- logger: Logger {readOnly}
- + UserAccount(accountHolder: TransitUser)
- + changeName(firstName: String, lastName: String): void
- + getName(): String
- + averageTransitCostPerMonth(startDate: LocalDate, endDate: LocalDate) : double
- + toString(): String
- + handleTransitUserLogin(email: String, plainPassword: String): TransitUserAccount
- + handleNewAccount(email: String, firstName: String, lastName: String, plainPassword: String): void
- + addValue(value: float, card: Card, creditCardHolder: String, cvv: String, creditCardNumber: String): void
- + deactivateCard(card: Card): void
- + activateCard(card: Card): void
- + getAccountHolder(): TransitUser
- + viewCardRecentTrip(card: Card): String
- + removeCard(card: Card, transfer: Card): void
- + purchaseCard(creditCardHolder: String, cvv: String, creditCardNumber: String): void
- + purchaseTransitPass(purchaseTime: LocalDateTime, card: Card, startTime: LocalDate, duration: int): void
- + passInfo(time: LocalDateTime): String
- + viewCardTransactions(startDate: LocalDate, endDate: LocalDate, card: Card): String
- + hasPass(time: LocalDateTime): boolean
- + setAllTransitUserAccount(allTransitUserAccount: HashMap<String, TransitUserAccount>): void
- isValidName(fullName: String) : boolean
- getTotalMonths(startDate: LocalDate, endDate: LocalDate): long
- checkValidHolder(card: Card): void
- ~ finalizePurchaseCard(): String



- startTime: LocalDate
- endTime: LocalDate
- price: double
- type: String
- ~ makeTransitPass(startTime: LocalDate, duration: int): TransitPass
- ~ isValid(currentDate: LocalDate): boolean
- ~ getPrice(): double
- ~ getType(): String
- + toString(): String