





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

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

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

