SubTrip
`<<abstract>>`
SubTrip

- timeStart : LocalDateTime
- timeEnd : LocalDateTime
- startPoint : String
- endPoint : String
- fare: float
- connection : SubTrip
- enRoute : boolean
- path : String[ ]
- routeName : String
- numOfStops : int

- ~ SubTrip(timeStart: LocalDateTime, startPoint: String, routeName: String)
- ~ finishTrip(timeEnd: LocalDateTime, endPoint: String) : void
- ~ getFare( ) : float
- ~ setFare(fare : float) : void
- ~ setConnection(connection: SubTrip) : void
- ~ isEnRoute( ) : boolean
- ~ getTimeStart( ): LocalDateTime
- ~ getConnection( ): SubTrip
- ~ getRouteName( ): String
- ~ getNumOfStops( ) : int
- ~ getStartPoint( ) : String
- ~ getEndPoint( ) : String
- - endTrip(timeEnd: LocalDateTime, endPoint: String) : void
- - setPath( ): void
- - getTimeEnd( ): LocalDateTime
- - reverse(arrayToReverse : String[ ]) : String[ ]
- *calculateFare( ) : float*
- *getRoute( ) : String[ ]*
- + toString( ) : String

SubwaySubTrip
SubwaySubTrip

- FARE_PER_STATION : float {readOnly}
- subwayRoutes : HashMap<String, String[ ]>

- ~ SubwaySubTrip(timeStart: LocalDateTime, startPoint: String, route: String)
- ~ calculateFare( ) : float
- ~ getRoute( ) : String[ ]
- ~ getSubwayRoutes( ) : HashMap<String, String[ ]>
- + toString( ) : String
- + setSubwayRoutes(subwayRoutes: HashMap<String, String[ ]>) : void

BusSubTrip
BusSubTrip

- FARE_PER_TRIP : float {readOnly}
- busRoutes : HashMap<String, String[ ]>

- ~ BusSubTrip(timeStart: LocalDateTime, startPoint: String, route: String)
- ~ calculateFare( ) : float
- ~ getRoute( ) : String[ ]
- ~ getBusRoutes( ) : HashMap<String, String[ ]>
- + toString( ) : String
- + setBusRoutes(subwayRoutes: HashMap<String, String[ ]>) : void

Card
Card

- cardHolder: TransitUser {readOnly}
- cardId : String {readOnly}
- balance: double
- activated : boolean
- currentTrip : Trip
- trips : ArrayList<Trip>
- transactions : Map<LocalDateTime, Double> {readOnly}

- + Card(cardHolder: TransitUser, id: String)
- + viewLastThreeTrips(email: String) : String
- + tapIn(time: LocalDateTime, station: String, route: String): void
- + tapOut(time: LocalDateTime, station: String, route: String): void
- + toString( ) : String
- + setBalance(balance: double) : void
- + addValue(value: float, email: String) : void
- + deactivateCard(email: String) : void
- + getCardHolder( ) : TransitUser
- + activateCard(email: String): void
- + viewBalance(email: String) : double
- + getCardId( ) : String
- - pay(fare: double, time: LocalDateTime) : void
- - finishTrip(newSubTrip: SubTrip) : void
- - finishTrip( ): void
- - tapIntoBus(time: LocalDateTime, station: String, route: String): void
- - tapIntoSubway(time: LocalDateTime, station: String, route: String): void
- - tapOutOfBus(time: LocalDateTime, station: String): void
- - tapOutOfSubway(time: LocalDateTime, station: String): void
- - getMonthActive( ) : long
- - getTotalCosts( ) : double
- - isActivated( ) : boolean
- ~ getAverageMonthlyCost( ): double

UserAccount
UserAccount

- accountHolder: TransitUser

- + UserAccount(accountHolder: TransitUser)
- + getAccountHolder( ) : TransitUser
- + changeName(firstName: String, lastName: String): void
- + getName( ) : String
- + getEmail( ) : String
- + showThreeRecentTrips( ) : String
- + averageTransitCostPerMonth( ) : double
- + toString( ) : String
- - isValidName(fullName: String) : boolean

Trip
Trip

- start : SubTrip
- end : SubTrip

- ~ Trip(start: SubTrip)
- ~ appendTrip(connection: SubTrip): void
- ~ getEnd( ): SubTrip
- ~ getCurrentTotalFare( ) : float
- + toString( ) : String
- - canContinue(time: LocalDateTime, station: String) : boolean

User
`<<abstract>>`
User

- firstName: String
- lastName : String
- email: String

- + User(firstName: String, lastName: String, email: String)
- + getEmail( ) : String
- + toString( ) : String
- ~ getName( ): String
- ~ setName(firstName: String, lastName: String) : void

TransitUser
TransitUser

- myCards : List<Card>
- myTrips : List<Trip>

- + TransitUser(firstName: String, lastName: String, email: String)
- + addCard(c : Card) : void
- + toString( ) : String
- ~ latestTrips( ) : String
- ~ getMyCards( ) : List<Card>
- ~ addToMyTrips(newTrip : Trip) : void
- - getThreeRecentTrips( ) : List<Trip>

Relationships:
- SubTrip 1..* — 0..1 Trip
- Trip 0..1 — 1..* Card
- Trip 0..1 — 1..* TransitUser
- Card 1..* — 0..1 TransitUser
- UserAccount 0..1 — 1..* (TransitUser)

AdminUser

- getDailyRevenue(date: LocalDate) : double
- getMonthlyRevenue(date: LocalDate) : double
- getTotalStopReached(date: LocalDate) : int
- getMonthlyAverage( ): double
+ getDailyReport(date: LocalDate) : String
+ getMonthlyReport(date: LocalDate) : String
+ getAverageReport( ): String

0..1

1..*

SystemManager

- dailyRevenue : Map<LocalDate, Double> {readOnly}
- busStopsPerDay: Map<LocalDate, Integer> {readOnly}
- subwayStationsPerDay: Map<LocalDate, Integer> {readOnly}

~ getMonthlyStopReached(date: LocalDate): int
~ recordBusStops(date: LocalDate, stopsReached: int) : void
~ recordSubwayStations(date: LocalDate, stationsReached: int) : void
~ addRevenue(date: LocalDate, revenue: double): void
~ getTotalBusStopsReached(date: LocalDate) : int
~ getTotalSubwayStationsReached(date: LocalDate) : int
~ getDailyRevenue(date: LocalDate) : double
~ getMonthlyRevenue(date: LocalDate): double
~ getAverageMonthlyRevenue( ) : double
~ getMonthlyAverageStopReached( ) : int
- getNumMonths(set: Set<LocalDate>) : long