Report Car Rental and Hotel Reservation

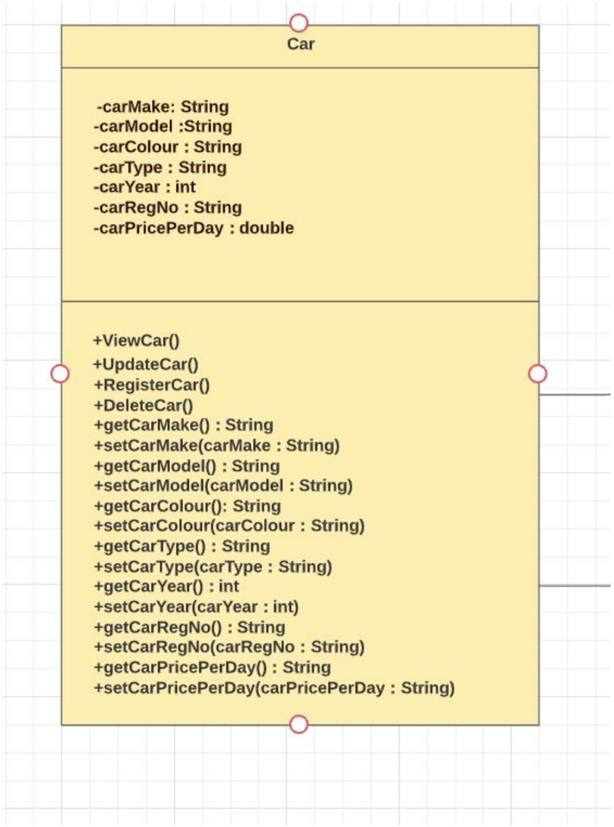
Overview:

Car Rental and Hotel Reservation systems can be extended to the Airline Management System. A customer when going to multiple cities or visiting a new place requires car rental and hotel booking. For this purpose we can extend our system to car rental and Hotel Reservation System. Below we have explained how we can implement this system and extended Airline Management System and what all entities we have used in this system.

Java Implementation:

We have extended airline management system where we are adding car rental and hotel reservation which has Customer, Reservation class as common class among all three systems. We need to create two POJO classes for cars and hotels which include specific attributes and methods to extend the system.
We have reused various classes of Airline Management System to extend the project

For Example:
CAR

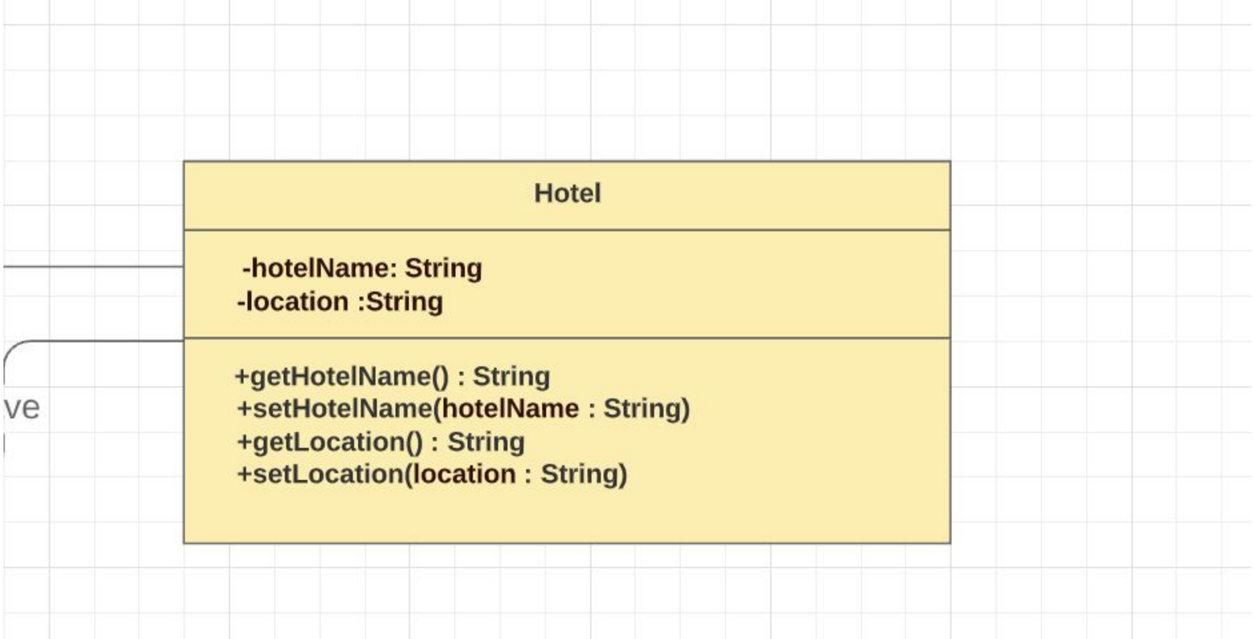


```
Car

-carMake: String
-carModel :String
-carColour : String
-carType : String
-carYear : int
-carRegNo : String
-carPricePerDay : double

+ViewCar()
+UpdateCar()
+RegisterCar()
+DeleteCar()
+getCarMake() : String
+setCarMake(carMake : String)
+getCarModel() : String
+setCarModel(carModel : String)
+getCarColour(): String
+setCarColour(carColour : String)
+getCarType() : String
+setCarType(carType : String)
+getCarYear() : int
+setCarYear(carYear : int)
+getCarRegNo() : String
+setCarRegNo(carRegNo : String)
+getCarPricePerDay() : String
+setCarPricePerDay(carPricePerDay : String)
```

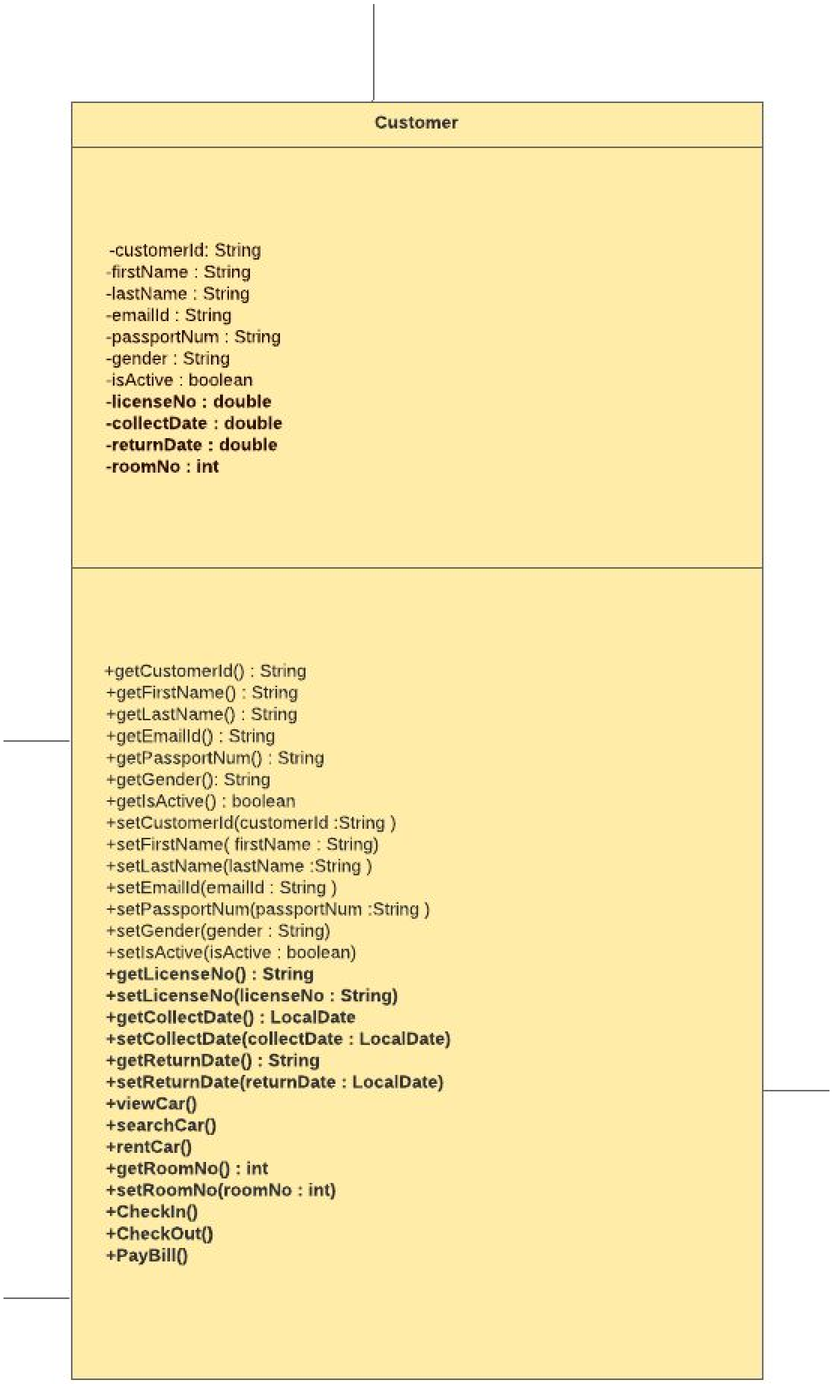
HOTEL:


ve
Hotel
-hotelName: String
-location :String
+getHotelName() : String
+setHotelName(hotelName : String)
+getLocation() : String
+setLocation(location : String)

Entities:

CUSTOMER:

Customer is reused class from Airline Management System



Customer

-customerId: String
-firstName : String
-lastName : String
-emailId : String
-passportNum : String
-gender : String
-isActive : boolean
-licenseNo : double
-collectDate : double
-returnDate : double
-roomNo : int

+getCustomerId() : String
+getFirstName() : String
+getLastName() : String
+getEmailId() : String
+getPassportNum() : String
+getGender(): String
+getIsActive() : boolean
+setCustomerId(customerId :String)
+setFirstName(firstName : String)
+setLastName(lastName :String)
+setEmailId(emailId : String)
+setPassportNum(passportNum :String)
+setGender(gender : String)
+setIsActive(isActive : boolean)
+getLicenseNo() : String
+setLicenseNo(licenseNo : String)
+getCollectDate() : LocalDate
+setCollectDate(collectDate : LocalDate)
+getReturnDate() : String
+setReturnDate(returnDate : LocalDate)
+viewCar()
+searchCar()
+rentCar()
+getRoomNo() : int
+setRoomNo(roomNo : int)
+CheckIn()
+CheckOut()
+PayBill()

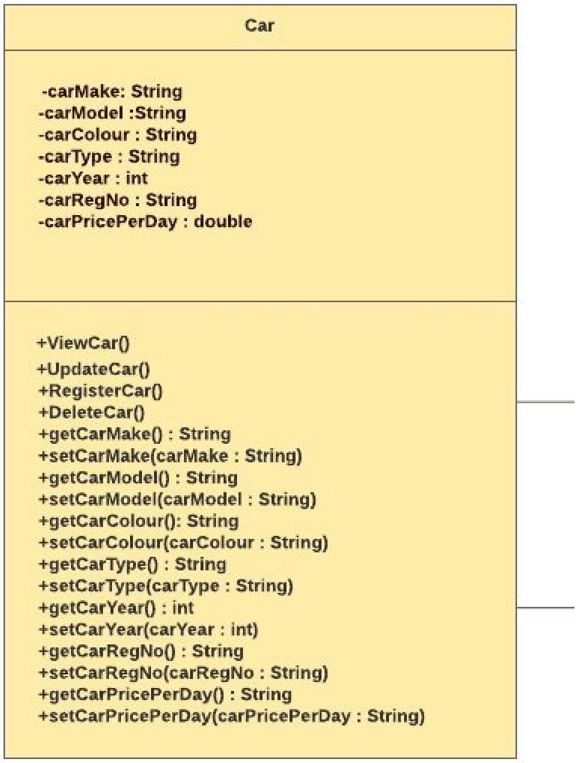
Attributes : customerID, firstName , lastName, emailID, passportNum, gender, isActive, licenseNo, collectDate, returnDate, roomNo

Methods : Customer Getter and Setter Method
+getCustomerId() : String
+getFirstName() : String
+getLastName() : String
+getEmailId() : String
+getPassportNum() : String
+getGender(): String
+getIsActive() : boolean
+setCustomerId(customerId :String)
+setFirstName(firstName : String)
+setLastName(lastName :String)
+setEmailId(emailId : String)
+setPassportNum(passportNum :String)
+setGender(gender : String)
+setIsActive(isActive : boolean)
+getLicenseNo() : String
+setLicenseNo(licenseNo : String)
+getCollectDate() : LocalDate
+setCollectDate(collectDate : LocalDate)
+getReturnDate() : String
+setReturnDate(returnDate : LocalDate)
+getRoomNo()
+setRoomNo(roomNo : int)

All these methods are getter setter methods

Functional Methods Implementation:
+viewCar() : this functionality will help view car details.
+searchCar() : we will add car details in the Array List and fetch car details based on car Name
+rentCar() : In car rental functionality we will assign a car to a customer with a driver details and view it on the booking detail page.
+CheckIn() : it will store customer checkIn time
+CheckOut() : It will store customer checkout time
+PayBill() : This functionality will calculate bill to by paid by customer based on number of days he rents the car or books the hotel.

CAR:



Attributes :
carMake, carMode, carColour, carType, carYear, carRegNo, carPricePerDay

Methods : Getter and Setter method of Car

+getCarMake() : String
+setCarMake(carMake : String)
+getCarModel() : String
+setCarModel(carModel : String)
+getCarColour(): String
+setCarColour(carColour : String)
+getCarType() : String
+setCarType(carType : String)
+getCarYear() : int
+setCarYear(carYear : int)
+getCarRegNo() : String
+setCarRegNo(carRegNo : String)
+getCarPricePerDay() : String
+setCarPricePerDay(carPricePerDay : String)

Team : Shivi Bhatt and Mayank Deshpande

Functional Methods Implementation:

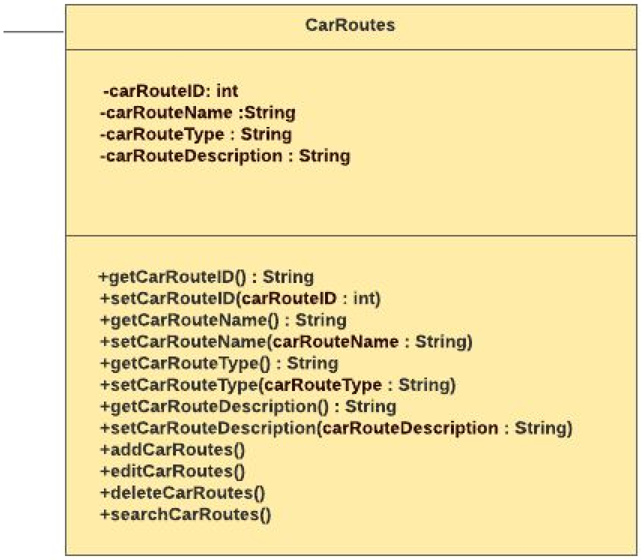
+ViewCar() : Using an object car we will fetch details of all car attributes and display it.

+UpdateCar() : Update car details, for example car Number is changed can be updated through this functionality.

+RegisterCar() : Register car with customer by assigning carID to customer

+DeleteCar() : to unregister the car we need to delete or make soft delete using a status delete car.

CAR ROUTES :



CarRoutes

-carRouteID: int
-carRouteName :String
-carRouteType : String
-carRouteDescription : String

+getCarRouteID() : String
+setCarRouteID(carRouteID : int)
+getCarRouteName() : String
+setCarRouteName(carRouteName : String)
+getCarRouteType() : String
+setCarRouteType(carRouteType : String)
+getCarRouteDescription() : String
+setCarRouteDescription(carRouteDescription : String)
+addCarRoutes()
+editCarRoutes()
+deleteCarRoutes()
+searchCarRoutes()

Attributes:

carRouteID, carRouteName, carRouteType , carRouteDescription

Methods :

+getCarRouteID() : String
+setCarRouteID(carRouteID : int)
+getCarRouteName() : String
+setCarRouteName(carRouteName : String)
+getCarRouteType() : String
+setCarRouteType(carRouteType : String)
+getCarRouteDescription() : String
+setCarRouteDescription(carRouteDescription : String)

Functional Methods Implementation:

+addCarRoutes() : Add car routes using ArrayList of Car Type

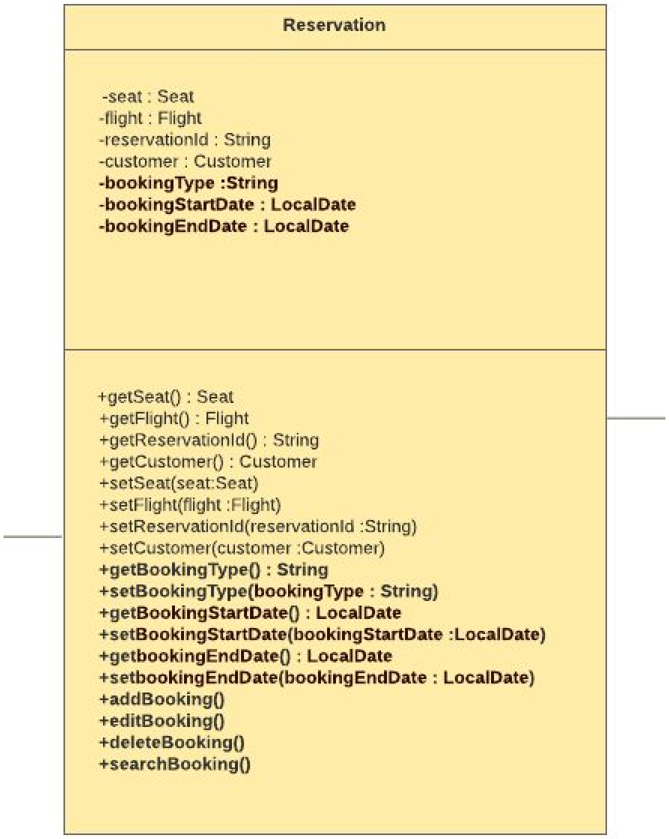
+editCarRoutes() : Update car routes if suppose car routes are changed can be updated through this functionality

+deleteCarRoutes() Using Array List remove method customers can remove car.

+searchCarRoutes() Using Array List customer can search for the car routes he want to book for rental.

RESERVATION:

Reservation is reused class from Airline Management System



Reservation

-seat : Seat
-flight : Flight
-reservationId : String
-customer : Customer
-bookingType :String
-bookingStartDate : LocalDate
-bookingEndDate : LocalDate

+getSeat() : Seat
+getFlight() : Flight
+getReservationId() : String
+getCustomer() : Customer
+setSeat(seat:Seat)
+setFlight(flight :Flight)
+setReservationId(reservationId :String)
+setCustomer(customer :Customer)
+getBookingType() : String
+setBookingType(bookingType : String)
+getBookingStartDate() : LocalDate
+setBookingStartDate(bookingStartDate :LocalDate)
+getbookingEndDate() : LocalDate
+setbookingEndDate(bookingEndDate : LocalDate)
+addBooking()
+editBooking()
+deleteBooking()
+searchBooking()

Attributes : bookingID, bookingType, bookingStartDate, bookingEndDate

Methods : Getter and Setter methods for booking attributes

+getBookingID() : String
+setBookingID(bookingID : int)
+getBookingType() : String
+setBookingType(bookingType : String)
+getBookingStartDate() : LocalDate
+setBookingStartDate(bookingStartDate :LocalDate)
+getbookingEndDate() : LocalDate
+setbookingEndDate(bookingEndDate : LocalDate)

Functional Methods Implementation:

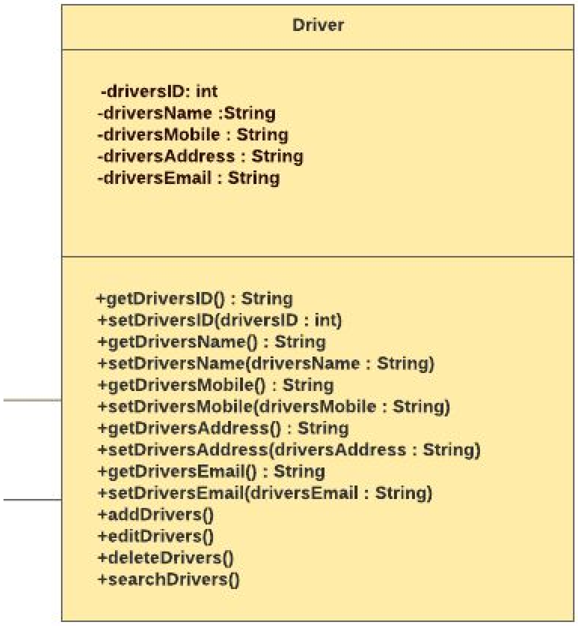
+addBooking() **:** Using Array List to add booking of the cars against the customer, Receptionist can add booking for the customers who wants to book a hotel room.

+editBooking() **:** Update booking by setting key attributes based on the user requirements.

+deleteBooking() **:** Using Array List delete booking can be done if customer wants to delete the booking or cancel the booking

+searchBooking() **:** Search Booking can be done using Array List and booking can be searched based on booking ID

DRIVER:



Driver

-driversID: int
-driversName :String
-driversMobile : String
-driversAddress : String
-driversEmail : String

+getDriversID() : String
+setDriversID(driversID : int)
+getDriversName() : String
+setDriversName(driversName : String)
+getDriversMobile() : String
+setDriversMobile(driversMobile : String)
+getDriversAddress() : String
+setDriversAddress(driversAddress : String)
+getDriversEmail() : String
+setDriversEmail(driversEmail : String)
+addDrivers()
+editDrivers()
+deleteDrivers()
+searchDrivers()

Attributes: driversID, driversName, driversMobile, driversAddress, driversEmail

Methods : Drivers basic Details Getter and Setter

+getDriversID() : String
+setDriversID(driversID : int)
+getDriversName() : String
+setDriversName(driversName : String)
+getDriversMobile() : String

+setDriversMobile(driversMobile : String)
+getDriversAddress() : String
+setDriversAddress(driversAddress : String)
+getDriversEmail() : String
+setDriversEmail(driversEmail : String)

Functional Methods Implementation:

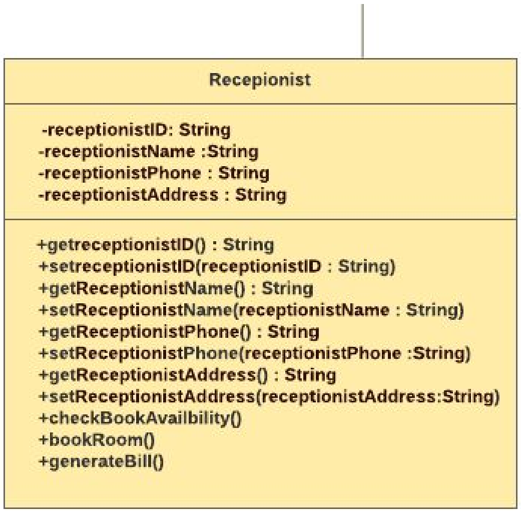
+addDrivers() : Add driver details using Array List where we can store number of drivers

+editDrivers() : Update driver details such as address or phone number using setter and getter method and than updating the list for same.

+deleteDrivers() : Delete driver details if driver is no longer associated with Car rental Company

+searchDrivers() : search driver details based on driver ID using Array List

RECEPTIONIST:


Recepionist
-receptionistID: String
-receptionistName :String
-receptionistPhone : String
-receptionistAddress : String
+getreceptionistID() : String
+setreceptionistID(receptionistID : String)
+getReceptionistName() : String
+setReceptionistName(receptionistName : String)
+getReceptionistPhone() : String
+setReceptionistPhone(receptionistPhone :String)
+getReceptionistAddress() : String
+setReceptionistAddress(receptionistAddress:String)
+checkBookAvailbility()
+bookRoom()
+generateBill()

Attributes: receptionistID, receptionistName, receptionistPhone, receptionistAddress

Methods : Receptionist Basic Details Getter and Setter Method

+getreceptionistID() : String
+setreceptionistID(receptionistID : String)
+getReceptionistName() : String
+setReceptionistName(receptionistName : String)
+getReceptionistPhone() : String
+setReceptionistPhone(receptionistPhone :String)
+getReceptionistAddress() : String
+setReceptionistAddress(receptionistAddress:String)

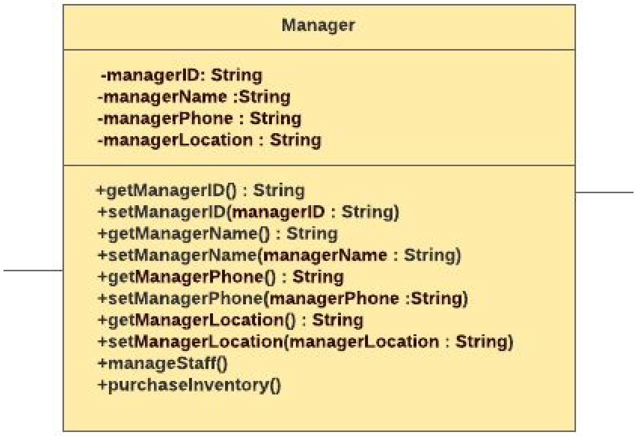
Functional Methods Implementation:

+checkBookAvailbility() **:** This method will calculate how many number of booking is made , in this we can use boolean values to check for availability.

+bookRoom() **:** Assigning a room to a customer and setting boolean value false if room is not available

+generateBill() : After room is booked, we can generate bill based on per day tariff, this method will basically do more of calculations.

MANAGER:



Manager

-managerID: String
-managerName :String
-managerPhone : String
-managerLocation : String

+getManagerID() : String
+setManagerID(managerID : String)
+getManagerName() : String
+setManagerName(managerName : String)
+getManagerPhone() : String
+setManagerPhone(managerPhone :String)
+getManagerLocation() : String
+setManagerLocation(managerLocation : String)
+manageStaff()
+purchaseInventory()

Attributes: managerID, managerName, managerPhone, managerLocation
Methods: Manager Basic details Getter and Setter Method

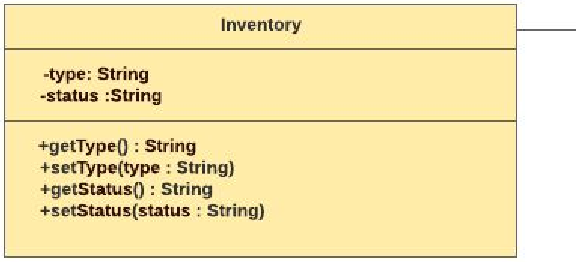
+getManagerID() : String
+setManagerID(managerID : String)
+getManagerName() : String
+setManagerName(managerName : String)
+getManagerPhone() : String
+setManagerPhone(managerPhone :String)
+getManagerLocation() : String
+setManagerLocation(managerLocation : String)

Functional Methods Implementation:

+manageStaff() **:** Manager staff method will store list of employees working under him using Array List

+purchaseInventory() **:** Purchase inventory will store list of purchased items and it is stored in array list

INVENTORY:



Attributes: type, status
Methods :

+getType() : String
+setType(type : String)
+getStatus() : String
+setStatus(status : String)

Description : getter and setter method of inventory which will store type of item and its status if it is available in stock or not.

MULTI STOP TRAVEL PLAN:
To provide support for multiple stops on a travel plan we will extend car rental and hotel management system to implement multiple car rental and multiple hotel booking for each stop for a customer booking, to implement this in java we will add multiple booking in Reservation class based on the user search where user is selecting multi city option, this business logic can be reused for multi stop option.