

Hotel
HotelID: String
Name: String
Rate: Double
+add Room()
+book Room()

Works

Employee
EmpID: String
Name: String
+assign Room()

Housekeeper
+keep up: String
+clean: String
+Salary: Double
+clean()

Receptionist
+staffID: String
+name: String
+Salary: Double
+works()

Hotel Management

Rooms

Room
RoomID: String
RoomType: String
Price: Double
+reserve Room()
+free Room()

RoomType
SINGLE
DOUBLE

Reserves

Reservation
ReservationID: String
+checkDate: Date
+checkOut: Date
+confirm Reservation()

Employee
empID: String
RoomNo: Integer

Customer

+guestID: String
+name: String
+phoneNo: Integer
+email: String
+makeReservation()

Overlapping

Payment

+paymentID: String
+amount: Double
+paymentMethod: PaymentMethod
+processPayment()

Payment Method

Cash  
Online



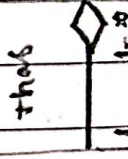
# Credit card

Statement	+ Statement ID: String
+ SSN: Date: String	
+ total Amount Due: Double	
+ generate Statement (customer: Customer)	

Transaction Status	Pending Confirmed Cancelled Refunded
--------------------	-----------------------------------------------

## Credit Card

CardNo: String	CardCVV: int	expiry Date: String	balance: Double
+ validate Card CVV: Boolean	+ block Card (void)		



Customer	+ Customer ID: Integer (unique)	+ Name: String	+ Contact: Integer (0..10)	+ Make Payment (amount: Double)	+ Request Card Replacement()
----------	---------------------------------	----------------	----------------------------	---------------------------------	------------------------------

## Transaction

Transaction	+ Transaction ID: int	+ amt: Double	+ date: Date	+ status: Transaction Status
	+ initiate Transaction()	+ cancel Transaction()	+ refund Transaction()	

## Participant

Bank	+ BankID: String	+ BankName: String	+ Contact: String	+ authorize Transaction()	+ block Card()	+ settle Transaction()
------	------------------	--------------------	-------------------	---------------------------	----------------	------------------------

Transaction

belongs



# Library Management System

Person

contains

Library

+ LibID: String  
+ name: String  
+ address: String  
+ addBook(): void  
+ removeBook(): void

Book  
+ ISBN: int  
+ title: String  
+ author: String  
+ borrowBook(): void  
+ returnBook(): void  
+ ISBN

Journal

newspaper

manages

involves

Librarian

+ LibrarianID: String  
+ name: String  
+ contact: int  
+ manageInventory(): void  
+ assignMember(): void

ShiftType

Morning

Evening

Borrow Transaction

+ transactionID: String  
+ borrowDate: Date  
+ returnDate: Date  
+ fineAmount: Double  
+ calculateFine(): Double

Fine

+ fineID: String  
+ fineAmount: float  
+ status: FineStatus  
+ calculateFine(): void  
+ payFine(): void

Enumeration

FineStatus

Paid

Unpaid

Member

+ memberID: String  
+ name: String  
+ contact: int  
+ membership: MembershipType  
+ borrowBook(): void  
+ returnBook(): void

Reservation

+ reservationID: String  
+ reservedDate: Date  
+ status: Status  
+ reserveBook(): void  
+ cancelReservation(): void

Enumeration

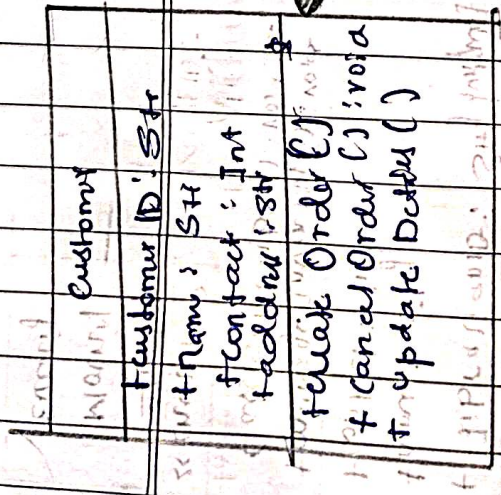
ReservationStatus

Active

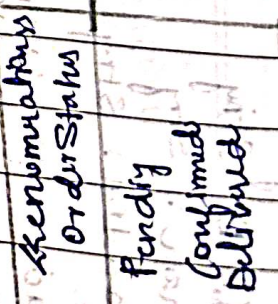
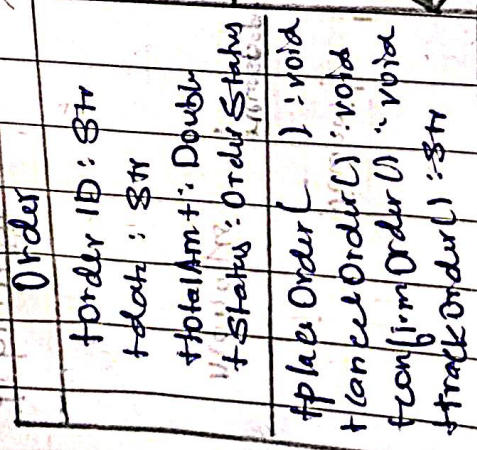
Cancelled

Completed

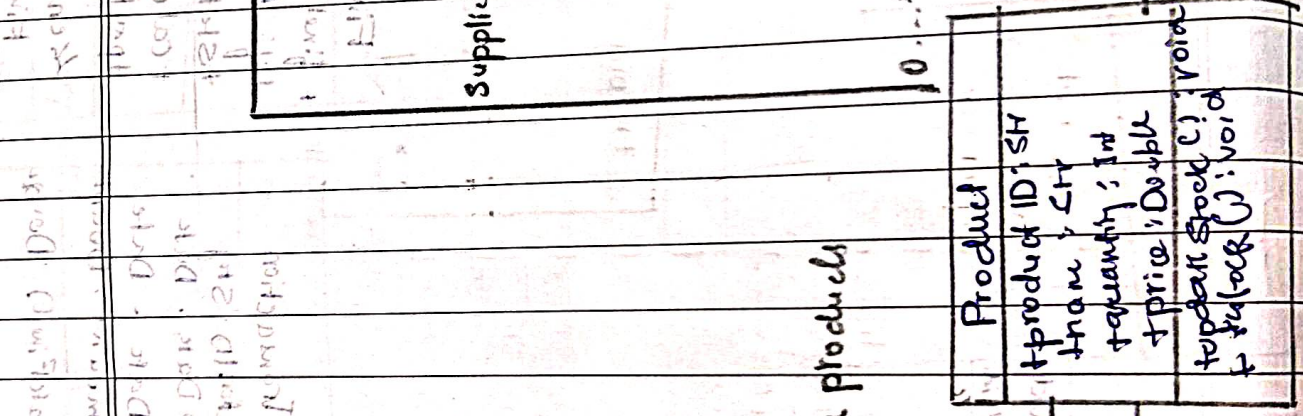




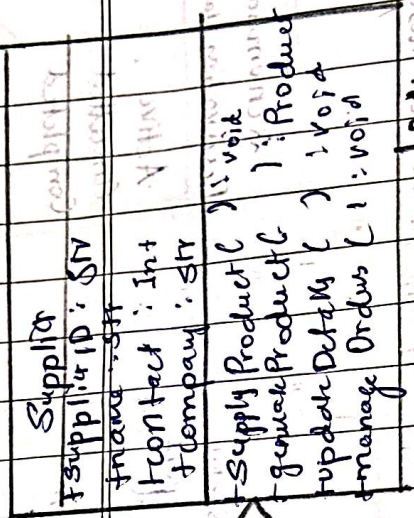
CustomerOrder



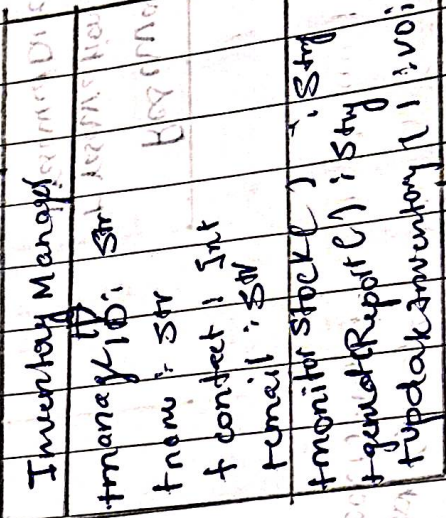
Stock Maintenance System



torderID products



Instructions



manage products



## Passport Automation System

**Application**  
 + applicationID : Str  
 + submit : Str  
 + status : Boolean  
 + applicationType : Str  
 + submit Application ( ) : void  
 + check Status ( ) : Boolean  
 + approve Application ( ) : void  
 + reject Application ( ) : void

+ applicationID

+ applicationID

issues

trsubmit

<<enumeration>>  
 passport type  
 Foreign  
 National

Passport

+ passportNumber : Int  
 + issue Date : Str  
 + expiry Date : Str  
 + passportType : Str

+ renew Passport ( ) : void

+ renewe Passport ( ) : void

+ issue Passport ( ) : void

Standard

Diplomatic

issues

## Passport Automation System

contains

Application

+ applicationID : Str  
 + name : Str  
 + address : Str  
 + contact : Str

+ applyForPassport ( ) : Application  
 + uploadDoc ( ) : void  
 + updateDefault ( ) : void  
 + track Application ( ) : Boolean

validity

provides

Officer

+ officerID : Str  
 + name : Str  
 + dept : Str  
 + contact : Str

+ apply Doc ( ) : Boolean

+ approve Doc ( ) : void

+ reject Doc ( ) : void

+ issue Passport (application) : void

Document

+ docID : Str  
 + docType : Str  
 + issue Date : Str  
 + validateDoc ( ) : Boolean