**Solid Design Principles on DIP Activity**

In Online Ticket Booking App while booking the ticket customers can book ticket through credit card or debit card and for the same way existing class design like below

**public** **class** TicketBookingApp {

**private** DebitCard debitCard;

**public** TicketBookingApp(DebitCard debitCard) {

**this**.debitCard = debitCard;

}

**public** **void** doPayment(**int** noOfTickets, **int** amount) {

debitCard.doTransaction(amount);

}

**public** **static** **void** main(String[] args) {

DebitCard debitCard = **new** DebitCard();

TicketBookingApp ticketApp = **new** TicketBookingApp(debitCard);

ticketApp.doPayment(4,5000);

}

}

**public** **class** DebitCard {

**public** **void** doTransaction(**int** amount) {

System.***out***.println("tx done with DebitCard");

}

}

**public** **class** CreditCard {

**public** **void** doTransaction(**int** amount) {

System.***out***.println("tx done with CreditCard");

}

}

From the above design customer can do payment only through transaction debit card, but customer want to pay through credit card we should change the code in TicketBookingApp class.

TicketBookingApp is tightly coupled with DebitCard which is not suggested according to the one of the design principle.

**Task 5:** Design the Ticket Booking App in a loosely coupled manner so that customer can do payment through credit card or debit card.

**Note:** In the given application solid task where you can find package called

**Com.epam.dip.activity.solution** design your code in that package.