1. Create an console application to book train tickets. Create a Passanger class with (Name, Age) and write a function called TicketBooking(no\_of\_tickets) that takes no.of tickets to be booked. If the no of tickets is > 2 per booking, raise an user defined exception, and print "cannot book more than 2 tickets". Else Print "Ticket Booked Successfully". Add a Test class to call TicketBooking method by accepting all required details.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Assignment5

{

class Passenger

{

internal string PName;

internal int PAge;

internal Passenger(string PName, int PAge)

{

this.PName = PName;

this.PAge = PAge;

}

public void TicketBooking(int no\_of\_tickets)

{

try

{

if (no\_of\_tickets > 2)

{

Console.WriteLine("cannot book more than 2 tickets");

}

else

{

Console.WriteLine("Ticket Booked Successfully");

}

}

catch

{

Console.WriteLine("e.Message");

}

}

}

class TrainTickets

{

static void Main()

{

Passenger passenger = new Passenger("Haari",23);

Console.WriteLine("Enter name of the passenge:{0}",passenger.PName);

Console.WriteLine("Enter the age of the passenger:{0}",passenger.PAge);

Console.WriteLine("Enter no of tickets to be booked");

int nooftickets = Convert.ToInt32(Console.ReadLine());

passenger.TicketBooking(nooftickets);

Console.Read();

}

}

}

2. Write a class called LoanProcess with Loan\_No, Customer Name, LoanAmount, EMI\_Amount, Account\_Balance as its members. Create a method calculate\_EMI() for the LoanAmount , with the rate of interest as 13% for a total of 3 years and store it in the EMI\_Amount. The rest of the information to be passed through constructors. Write another function CheckBalance() which checks if the Account\_Balance is less than the EMI\_AMount. If yes then throw a custom exception. Display " Not Sufficient Balance to repay Loan" in the finally. Give explanatory comments.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Assignment5

{

class LoanProcess

{

int LoanNumber;

string CustomerName;

internal int LoanAmount;

float EMI\_Amount;

internal int AccountBalance;

internal LoanProcess(int LoanNumber, string CustomerName, int LoanAmount, int AccountBalance)

{

this.LoanNumber = LoanNumber;

this.LoanAmount = LoanAmount;

this.CustomerName = CustomerName;

this.AccountBalance = AccountBalance;

}

internal float Calculate\_EMI(int LoanAmount, float ROI, int time)

{

float r = (ROI / (12 \* 100));

int t = time \* 12;

EMI\_Amount = (LoanAmount \* r \* (float)Math.Pow(1 + r, t)) / (float)(Math.Pow(1 + r, t) - 1);

return EMI\_Amount;

}

public void CheckBalance(int AccountBalance, float EMI\_Amount)

{

try

{

if (AccountBalance < EMI\_Amount)

{

throw new Exception("Not Sufficient Balance to Repay the Loan");

}

}

catch (Exception e)

{

Console.WriteLine(e.Message);

}

finally

{

Console.WriteLine("Not Sufficient Balance to Repay the Loan");

}

}

}

class Loan

{

static void Main()

{

LoanProcess loanProcess = new LoanProcess(123456,"Harika",45000,7570);

float EMI = loanProcess.Calculate\_EMI(loanProcess.LoanAmount,13,3);

loanProcess.CheckBalance(loanProcess.AccountBalance, EMI);

Console.Read();

}

}

}