using System;

using System.Net;

using System.Threading.Tasks;

using System.Xml.Linq;

public class Item

{

private double shippingWeight;

private string description;

public Item() { } //defualt constractor

public Item(double shippingWeight, string description)

{

this.shippingWeight = shippingWeight;

this.description = description;

}

public double ShippingWeight

{

get { return shippingWeight; }

set { shippingWeight = value; }

}

public string Description

{

get { return description; }

set { description = value; }

}

public double GetPriceForQuantity()

{

return 0.0;

}

public double GetTax()

{

return 0.0;

}

public bool InStock()

{

return true;

}

public string tostring()

{

return $"shippingWeight {shippingWeight} " + $"description {description}";

}

}

public class OrderDetail

{

private int quantity;

private string taxStatus;

public OrderDetail() { } //defualt constractor

public OrderDetail(int quantity, string taxStatus)

{

this.quantity = quantity;

this.taxStatus = taxStatus;

}

public int Quantity

{

get { return quantity; }

set { quantity = value; }

}

public string TaxStatus

{

get { return taxStatus; }

set { taxStatus = value; }

}

public double CalcSubTotal()

{

return 0.0;

}

public double CalcWeight()

{

return 0.0;

}

public double CalcTax()

{

return 0.0;

}

public string tostring()

{

return $"quantity {quantity} " + $"taxStatus {taxStatus}" ;

}

}

public class Order

{

private DateTime date;

private string status;

private OrderDetail od1;

private List<OrderDetail> orderDetails;

public Order() { } //defualt constractor //relationship is aggrecation so the obj will list or common obj

public Order(DateTime date, string status, OrderDetail od1, List<OrderDetail> orderDetails)

{

this.date = date;

this.status = status;

this.od1 = od1;

this.orderDetails = orderDetails;

}

public DateTime Date

{

get { return date; }

set { date = value; }

}

public string Status

{

get { return status; }

set { status = value; }

}

public double CalcSubTotal()

{

return 0.0;

}

public double CalcTax()

{

return 0.0;

}

public double CalcTotal()

{

return 0.0;

}

public double CalcTotalWeight()

{

return 0.0;

}

public string tostring()

{

return $"date {date} " + $"status {status}" + $"od1 {od1}" +$"orderDetails {orderDetails}";

}

}

public class Customer

{

private string name;

private string address;

public Customer() { } //defualt constractor

public Customer(string name, string address)

{

this.name = name;

this.address = address;

}

public string Name

{

get { return name; }

set { name = value; }

}

public string Address

{

get { return address; }

set { address = value; }

}

public string tostring()

{

return $"name {name} " +$"address {address}";

}

}

public class Payment

{

private float amount;

public Payment() { } //defualt constractor //superclass

public Payment(float amount)

{

this.amount = amount;

}

public float Amount

{

get { return amount; }

set { amount = value; }

}

public string tostring()

{

return $"amount {amount} ";

}

}

public class Cash : Payment

{

private float cashTendered;

public Cash() { }//defualt constractor //subclass

public Cash(float amount, float cashTendered) : base(amount)

{

this.cashTendered = cashTendered;

}

public string tostring()

{

return $"cashTendered {cashTendered} " ;

}

}

public class Check : Payment

{

private string name;

private string bankID;

public Check() { }//defualt constractor //subclass

public Check(float amount, string name, string bankID) : base(amount)

{

this.name = name;

this.bankID = bankID;

}

public string Name

{

get { return name; }

set { name = value; }

}

public string BankID

{

get { return bankID; }

set { bankID = value; }

}

public string tostring()

{

return $"name{name}" + $"bankID {bankID}";

}

public bool Authorized()

{

return true;

}

}

public class Credit : Payment

{

private string name;

private string type;

private DateTime expDate;

public Credit() { }//defualt constractor //subclass

public Credit(float amount, string name, string type, DateTime expDate) : base(amount)

{

this.name = name;

this.type = type;

this.expDate = expDate;

}

public string Name

{

get { return name; }

set { name = value; }

}

public string Type

{

get { return type; }

set { type = value; }

}

public DateTime ExpDate

{

get { return expDate; }

set { expDate = value; }

}

public string tostring()

{

return $"name{name}" + $"type {type}" +$"expDate{expDate}";

}

public bool Authorized()

{

return true;

}

}

public class HelloWorld

{

public static void Main(string[] args)

{

Item item1 = new Item(2.5, "Widget");

OrderDetail od1 = new OrderDetail(3, "Taxable");

List<OrderDetail> orderDetails = new List<OrderDetail> { od1 };

Order o1 = new Order(DateTime.Now, "Pending", od1, orderDetails);

Customer customer1 = new Customer("John Doe", "123 Main St");

Payment p1 = new Payment(50.0f);

Cash cash1 = new Cash(50.0f, 60.0f);

Credit credit1 = new Credit(50.0f, "John Doe", "Visa", DateTime.Now.AddYears(2));

Check check1 = new Check(50.0f, "John Doe", "123456");

// Displaying the inserted data

Console.WriteLine("Item: " + item1.tostring());

Console.WriteLine("Order Detail: " + od1.tostring());

Console.WriteLine("Order: " + o1.tostring());

Console.WriteLine("Customer: " + customer1.tostring());

Console.WriteLine("Payment: " + p1.tostring());

Console.WriteLine("Cash Payment: " + cash1.tostring());

Console.WriteLine("Credit Card Payment: " + credit1.tostring());

Console.WriteLine("Check Payment: " + check1.tostring());

}

}