Foundation1

using System;

using System.Collections.Generic;

public class Comment

{

public string CommenterName { get; set; }

public string CommentText { get; set; }

public Comment(string commenterName, string commentText)

{

CommenterName = commenterName;

CommentText = commentText;

}

public override string ToString()

{

return $"{CommenterName}: {CommentText}";

}

}

public class Video

{

public string Title { get; set; }

public string Author { get; set; }

public int LengthInSeconds { get; set; }

private List<Comment> Comments { get; set; }

public Video(string title, string author, int lengthInSeconds)

{

Title = title;

Author = author;

LengthInSeconds = lengthInSeconds;

Comments = new List<Comment>();

}

public void AddComment(Comment comment)

{

Comments.Add(comment);

}

public int GetCommentCount()

{

return Comments.Count;

}

public List<Comment> GetComments()

{

return Comments;

}

public void DisplayVideoInfo()

{

Console.WriteLine($"Title: {Title}");

Console.WriteLine($"Author: {Author}");

Console.WriteLine($"Length: {LengthInSeconds} seconds");

Console.WriteLine($"Number of comments: {GetCommentCount()}");

Console.WriteLine("Comments:");

foreach (var comment in Comments)

{

Console.WriteLine(comment);

}

Console.WriteLine();

}

}

class Program

{

static void Main(string[] args)

{

// Create videos

Video video1 = new Video("Learn C# Basics", "John Doe", 600);

Video video2 = new Video("Advanced C# Programming", "Jane Smith", 1200);

Video video3 = new Video("C# OOP Concepts", "Alex Johnson", 800);

// Add comments to video1

video1.AddComment(new Comment("Alice", "Great video!"));

video1.AddComment(new Comment("Bob", "Very informative."));

video1.AddComment(new Comment("Charlie", "Helped me a lot, thanks!"));

// Add comments to video2

video2.AddComment(new Comment("David", "Loved the examples."));

video2.AddComment(new Comment("Eva", "Very well explained."));

video2.AddComment(new Comment("Frank", "I learned so much!"));

// Add comments to video3

video3.AddComment(new Comment("George", "OOP made easy."));

video3.AddComment(new Comment("Hannah", "Thanks for this explanation."));

video3.AddComment(new Comment("Ian", "Concise and clear."));

// Store videos in a list

List<Video> videos = new List<Video> { video1, video2, video3 };

// Display information for each video

foreach (var video in videos)

{

video.DisplayVideoInfo();

}

}

}

Foundation 2

using System;

using System.Collections.Generic;

public class Address

{

private string streetAddress;

private string city;

private string stateOrProvince;

private string country;

public Address(string streetAddress, string city, string stateOrProvince, string country)

{

this.streetAddress = streetAddress;

this.city = city;

this.stateOrProvince = stateOrProvince;

this.country = country;

}

public bool IsInUSA()

{

return country.ToLower() == "usa";

}

public string GetFullAddress()

{

return $"{streetAddress}\n{city}, {stateOrProvince}\n{country}";

}

}

public class Customer

{

private string name;

private Address address;

public Customer(string name, Address address)

{

this.name = name;

this.address = address;

}

public bool LivesInUSA()

{

return address.IsInUSA();

}

public string GetCustomerName()

{

return name;

}

public Address GetCustomerAddress()

{

return address;

}

}

public class Product

{

private string productName;

private string productId;

private decimal pricePerUnit;

private int quantity;

public Product(string productName, string productId, decimal pricePerUnit, int quantity)

{

this.productName = productName;

this.productId = productId;

this.pricePerUnit = pricePerUnit;

this.quantity = quantity;

}

public decimal GetTotalCost()

{

return pricePerUnit \* quantity;

}

public string GetProductName()

{

return productName;

}

public string GetProductId()

{

return productId;

}

}

public class Order

{

private List<Product> products;

private Customer customer;

public Order(Customer customer)

{

this.customer = customer;

products = new List<Product>();

}

public void AddProduct(Product product)

{

products.Add(product);

}

public decimal CalculateTotalCost()

{

decimal totalCost = 0;

foreach (var product in products)

{

totalCost += product.GetTotalCost();

}

decimal shippingCost = customer.LivesInUSA() ? 5 : 35;

return totalCost + shippingCost;

}

public string GetPackingLabel()

{

string label = "Packing Label:\n";

foreach (var product in products)

{

label += $"{product.GetProductName()} (ID: {product.GetProductId()})\n";

}

return label;

}

public string GetShippingLabel()

{

return $"Shipping Label:\n{customer.GetCustomerName()}\n{customer.GetCustomerAddress().GetFullAddress()}";

}

}

class Program

{

static void Main(string[] args)

{

// Create addresses

Address address1 = new Address("123 Main St", "New York", "NY", "USA");

Address address2 = new Address("456 Maple Ave", "Toronto", "Ontario", "Canada");

// Create customers

Customer customer1 = new Customer("John Doe", address1);

Customer customer2 = new Customer("Jane Smith", address2);

// Create products

Product product1 = new Product("Laptop", "P001", 999.99m, 1);

Product product2 = new Product("Mouse", "P002", 49.99m, 2);

Product product3 = new Product("Keyboard", "P003", 79.99m, 1);

// Create orders

Order order1 = new Order(customer1);

order1.AddProduct(product1);

order1.AddProduct(product2);

Order order2 = new Order(customer2);

order2.AddProduct(product2);

order2.AddProduct(product3);

// Display packing label, shipping label, and total cost for order1

Console.WriteLine(order1.GetPackingLabel());

Console.WriteLine(order1.GetShippingLabel());

Console.WriteLine($"Total Cost: ${order1.CalculateTotalCost()}\n");

// Display packing label, shipping label, and total cost for order2

Console.WriteLine(order2.GetPackingLabel());

Console.WriteLine(order2.GetShippingLabel());

Console.WriteLine($"Total Cost: ${order2.CalculateTotalCost()}\n");

}

}