Họ và Tên: Nguyễn Thanh Kiên.

MSSV: 22110092.

Bài tập:

Xây dựng lớp sản phẩm sữa Milk với các thuộc tính sau MilkName, MilkID, ProductionDate (ngày sản xuất), ExpiredDate (ngày hết hạn), Quantity (số lượng) với các yêu cầu sau

* + Các thuộc tính có các thao tác get và set tương ứng ✓
  + MilkID được cấu thành như sau: MILK01032016 (MILK + Ngày sản xuất) ✓
  + Viết lớp MilkMoreInfo như là một attribute cung cấp thêm thông tin Manufacturer (Nhà sản xuất), CompanyName (tên công ty phân phối sữa) cho lớp Milk ✓
  + Viết một interface gồm hai thao tác là Nhập và Xuất thông tin ra màn hình. Thực hiện gọi hai thao tác này cho bất kỳ một thực thể milk nào ✓
  + Viết delegate để thực hiện hai thao tác nhập và xuất này. ✓

**Làm thêm:**

**. Validation & Error Handling** ✓

**. Đọc/Ghi JSON** ✓

Bài làm:

1. Xây dựng lớp sản phẩm sữa Milk với các thuộc tính sau MilkName, MilkID, ProductionDate (ngày sản xuất), ExpiredDate (ngày hết hạn), Quantity (số lượng), Các thuộc tính có các thao tác get và set tương ứng, MilkID được cấu thành như sau: MILK01032016

+ Đầu tiên, em khai các các thư viện System, tạo Namespace đầu tiên có tên MilkManagement trong đó chứa lớp Program có chứa hàm main để hiển thị ra màn hình Console, ngoài ra nó còn dùng để chứa các tham số mặc định của các thuộc tính MilkName, MilkID, ProducTionDate, ExpiredDate và Quantity

A computer screen shot of a program code

Description automatically generated

+ Tiếp theo, em tạo Namespace thứ 2 – MainData với lớp Milk dùng để khai báo các thuộc tính yêu cầu và phương thức get set tương ứng với mỗi thuộc tính, trong đó có 2 phương thức thiết lập dạng chuỗi, 2 phương thức thiết lập dạng ngày tháng năm và 1 phương thức dạng giá trị số, tất cả các phương thức thiết lập này đều có tham số mặc định trong hàm main. Em sử dụng kiểu Datetime để hiển thị thông tin PublishDate trong MilkID, tuy vậy thông tin ExpiredDate không có trong MilkID, vì vậy, em cần phải sử dụng một điều kiện IF – ELSE để kiểm tra xem ExpiredDate đó có hay là không, tức giá trị của nó là NULL hay là một giá trị mặc định nào khác. Sau đó ở phương thức get set của ExpiredDate, em lại tiếp tục dùng thêm một điều kiện IF - ELSE để kiểm tra xem nó có phải NULL hay là không.

A screen shot of a computer code

Description automatically generated

A computer screen shot of text

Description automatically generated

A screenshot of a computer program

Description automatically generated

**Màn hình Console:**

A computer screen with white text

Description automatically generated

1. Viết lớp MilkMoreInfo như là một attribute cung cấp thêm thông tin Manufacturer (Nhà sản xuất), CompanyName (tên công ty phân phối sữa) cho lớp Milk

+ Em tạo Namespace thứ 3 tên là AttributeDate, trong đó em xây dựng một lớp MilkMoreInfo kế thừa từ System.Attribute, MilkMoreInfo sẽ cung cấp thông tin của Manufacturer và CompanyName. Namespace này cũng thêm phương thức khởi tạo có tham số và phương thức get & set.

A computer screen shot of a program code

Description automatically generated

+ Ở Namespace MainData, em thêm dòng ghi chú vào đầu lớp Milk để cho biết thông tin của Manufacturer và Company

A computer screen with white text

Description automatically generated

+ Ở Namespace MilkManagement, em bổ sung hàm main để thêm các thông tin của Manufacturer và CompanyName và vẫn giữ nguyên các thông tin ban đầu của Milk. Ở hàm main mới này, nếu em sử dụng MilkMoreInfo minfo = (MilkMoreInfo)Attribute, nó trả về tất cả các thuộc tính đính kèm trên lớp, bao gồm các thuộc tính không phải là MilkMoreInfo. Vậy nên khi chạy chương trình nó sẽ báo lỗi, do đó, em sử dụng một câu điều kiện IF – Else if (attributes is MilkMoreInfo minfo) để kiểm tra xem thuộc tính có của MilkMoreInfo hay là không, nếu có thì sẽ xuất hiện Output Message của Manufacturer và CompanyName

A screen shot of a computer program

Description automatically generated

**Màn hình Console:**

A screen shot of a computer

Description automatically generated

1. Viết một interface gồm hai thao tác là Nhập và Xuất thông tin ra màn hình. Thực hiện gọi hai thao tác này cho bất kỳ một thực thể milk nào

+ Em tạo một Interface trong Namespace MainData với hai phương thức Nhập (1 file .txt) và Xuất thông tin file txt đó bằng 1 file txt khác. Em cho lớp Milk kế thừa và sử dụng interface bằng class Milk : ImilkAction, sau đó em dùng StreamReader và StreamWriter để nhập và xuất file.

A screen shot of a computer program

Description automatically generated

A screen shot of a computer code

Description automatically generated

A computer code on a black background

Description automatically generated

+ Trong Namespace MilkManagement, em gọi lại hai phương thức Nhập và Xuất file này, và sử dụng một điều kiện IF để kiểm tra xem dữ liệu có rỗng hay không với điều kiện numline > 0.

A black screen with text

Description automatically generated

**Màn hình Console:**

A computer screen with white text

Description automatically generated

Màn hình console hiển thị thông tin sau khi đọc dữ liệu từ file input milk.txt và cập nhật một số thuộc tính trong đối tượng Milk, giá trị MilkName được cập nhật thành "THTrueMilk" sau khi đọc từ file milk.txt. Giá trị Quantity là 123456, khác với giá trị 13579 trong file input milk.txt. Đây là do thuộc tính Quantity được gán một giá trị mới 123456 trong phần code trước khi hiển thị trên màn hình console.

Input:

A screenshot of a computer

Description automatically generated

Output:

A screenshot of a computer

Description automatically generated

Sau khi đọc từ file milk.txt, thuộc tính Quantity của đối tượng Milk được thay đổi thành 789 và các thuộc tính còn lại vẫn giữ nguyên. Do đó, giá trị Quantity trong file output khác với giá trị trong file input (13579), và ExpiredDate cũng thay đổi từ 20/11/2017 sang 20/05/2017.

1. Viết delegate để thực hiện hai thao tác nhập và xuất này.

+ Bên ngoài Namespaces MilkManagement, em khai báo 1 delegate – con trỏ hàm MilkInputOuput trỏ đến hàm InputInfo và OutputInfo

+ Ở hàm main, em khai báo hai đối tượng action1 và action2 thuộc kiểu delegate đó và tham chiếu lần lượt đến các hàm nhập và xuất file trong lớp Milk, em dùng Invoke để yêu cầu 2 kiểu con trỏ hàm action1 và action2 thực thi 2 thao tác nhập và xuất như em mong muốn.

**Màn hình Console:**

A black screen with a black background

Description automatically generated

Input:

A screenshot of a computer

Description automatically generated

Output:

A screenshot of a computer

Description automatically generated

Không có sự khác biệt nào giữa 2 file OutputInfo và OutputInfoByDelegate. Ta sử dụng Delegate để tối ưu hóa code hơn, làm cho chương trình chạy nhanh hơn và tốt hơn so với việc sử dụng điều kiện IF.

Làm thêm:

1. **Validation & Error Handling**:

Em thêm kiểm tra ArgumentException trong ValMilkName, ValProductionDate, ValExpiredDate, và ValQuantity để đảm bảo dữ liệu đầu vào hợp lệ và sử dụng try-catch trong phương thức InputInfoJson và OutputInfoJson để xử lý các lỗi khi đọc và ghi dữ liệu.

A computer screen shot of a program code

Description automatically generated

A computer screen shot of text

Description automatically generated

1. **Đọc/Ghi JSON**:

Em thay đổi từ việc đọc/ghi file văn bản sang JSON bằng cách sử dụng JsonConvert.DeserializeObject và JsonConvert.SerializeObject từ thư viện Newtonsoft.Json và tạo file JSON input\_milk.json để nhập dữ liệu và output\_milk.json để xuất dữ liệu.

A computer screen shot of a program code

Description automatically generated

A computer screen with colorful text

Description automatically generated