Họ và Tên: Nguyễn Thanh Kiên.

MSSV: 22110092.

**Bài 1:**

**Xây dựng lớp Dog và Cat với các thuộc tính Name, Age, Height và Weight và thực hiện các chức năng sau:**

**. Tạo một danh sách gồm 3 con chó và 2 con mèo 🗸**

**. Thực hiện nhập và xuất thông tin của các con vật 🗸**

**. Bắt lỗi khi người dùng nhập tuổi âm hoặc lớn hơn 20 🗸**

**. Bắt lỗi khi người dùng nhập chuỗi hay ký tự (không phải số) các thông tin Height và Weight 🗸**

**Bài 2:**

**. Xây dựng lớp Football, Tenis, và lớp Volleyball với các thông tin số lượng người chơi, thời gian chơi, loại bóng chơi (tự nghĩ thêm thông tin khác) kèm hai phương thức nhập và xuất 🗸**

**. Suy nghĩ lớp cha Sport chứa các thông tin gì để các lớp con có thể kế thừa các thông tin đó 🗸**

**. Viết hàm nhập xuất danh sách các môn thể thao dựa vào mối quan hệ kế thừa trên (Khuyến khích dùng Đa Hình) 🗸**

**Làm thêm:**

**Bài 1: Tính BMI và chế độ ăn hằng ngày.**

**Bài 2: Sử dụng Interface để định nghĩa hành vi cụ thể cho môn thể thao**

Bài Lab: 01.

1. Xây dựng lớp Dog và Cat với các thuộc tính Name, Age, Height và Weight và thực hiện các chức năng sau:
   1. Tạo một danh sách gồm 3 con chó và 2 con mèo
   2. Thực hiện nhập và xuất thông tin của các con vật
   3. Bắt lỗi khi người dùng nhập tuổi âm hoặc lớn hơn 20
   4. Bắt lỗi khi người dùng nhập chuỗi hay ký tự (không phải số) các thông tin Height và Weight

Trước hết, em xây dựng 2 lớp Dog và Cat mẫu với các thông tin điền sẵn trong hàm Main.

Hàm Main:

A computer screen shot of text

Description automatically generated

Lớp Dog:

A computer screen with text

Description automatically generated

Lớp Cat:

A computer screen shot of a program

Description automatically generated

Kết quả sau khi chạy:

A computer screen with numbers and symbols

Description automatically generated

Ở đây, thông tin được Xuất ra 2 lần đối với mỗi đối tượng (Dog, Cat) bởi vì em đã gọi 2 phương thức DisplayInfo() và ToString() để xuất dữ liệu ra màn hình, tuy vậy chúng ta vẫn cần phải gọi phương thức ToString để ghi đè kiểu dữ liệu theo cách chúng ta muốn.

**Tạo một danh sách gồm 3 con chó và 2 con mèo**

Em tạo một mảng Object 5 phần tử, nhập Dog trước Cat, dùng cơ chế Boxing và unboxing để xuất đúng thông tin là chó hay là mèo, và dùng (obj[i].GetType() == typeof(Dog)) để kiểm tra object xem là chó hay là mèo.

A computer screen shot of a program

Description automatically generated

A computer screen with text

Description automatically generated

Kết quả sau khi chạy:

A computer screen shot of a black screen

Description automatically generated

**Thực hiện nhập và xuất thông tin của các con vật**

Em dùng hàm Console.Readline() để nhập thông tin các con vật,để nhập số thì em phải nhập chuỗi và dùng age = int.Parse(str) để chuyển chuỗi sang số, tuy vậy, em không thể khai báo str thêm 1 lần nào nữa vì sẽ gây ra lỗi trùng tên biến, vì vậy đối với thuộc tính height và weight, em sử dụng height = int.Parse(Console.ReadLine()); để có thể chuyển chuỗi sang số và xuất luôn.

Lớp Dog:

A computer screen with text

Description automatically generated

Lớp Cat:

A screen shot of a computer code

Description automatically generated

Hàm Main:

A computer screen with text

Description automatically generated

Kết quả sau khi chạy: A black screen with white text

Description automatically generated

**Bắt lỗi khi người dùng nhập tuổi âm hoặc lớn hơn 20**

Ở đây em chỉ nhập 1 Dog và 1 Cat cho đỡ mất thời gian.

Em sử dụng các kiến thức về Exception để bắt lỗi, với điều kiện tuổi <=0 & >20. Em sử dụng khai báo public class NegativeNumException : Exception và if (age <= 0 || age > 20)

                    throw new NegativeNumException();

Trước hết em khai báo Lớp NegativeNumException trong namespace MainData

A computer screen with text and symbols

Description automatically generated with medium confidence

Em viết hàm InputAge() riêng và trong đó đệ quy khi nhập sai, tức là em sẽ nhập sai nhiều lần cho đến đúng.

Lớp Dog:

A computer screen shot of a program

Description automatically generated

A screen shot of a computer

Description automatically generated

Lớp Cat:

A computer screen shot of a program

Description automatically generated

A screen shot of a computer

Description automatically generated

Kết quả sau khi chạy:

A computer screen with white text

Description automatically generated

**Bắt lỗi khi người dùng nhập chuỗi hay ký tự (không phải số) các thông tin Height và Weight**

Em tiếp tục sử dụng kiến thức của Exception và sử dụng  FormatException khi chuyển kiểu sai đối với phương thức height và weight, đề nghị nhập lại khi exception xảy ra. Em tạo thêm 2 hàm InputHeight() và InputWeight() để bắt lại FormatException, và sử dụng vòng lặp while (!isHeightValid) để nhập tới khi nào đúng định dạng số thì thôi.

Lớp Dog:

A screen shot of a computer program

Description automatically generated

A computer screen with colorful text

Description automatically generated

A computer screen shot of a program code

Description automatically generated

Lớp cat:

A screen shot of a computer program

Description automatically generated

A computer screen shot of a program

Description automatically generated

A computer screen shot of a program code

Description automatically generated

Kết quả sau khi chạy:

A screenshot of a computer

Description automatically generated

**Ý tưởng thêm: Tính BMI và chế độ ăn hằng ngày.**

A computer screen shot of a program

Description automatically generated

Em sử dụng if else cơ bản

Kết quả sau khi chạy:A screenshot of a computer

Description automatically generated

1. Xây dựng lớp Football, Tenis, và lớp Volleyball với các thông tin số lượng người chơi, thời gian chơi, loại bóng chơi (tự nghĩ thêm thông tin khác) kèm hai phương thức nhập và xuất. Suy nghĩ lớp cha Sport chứa các thông tin gì để các lớp con có thể kế thừa các thông tin đó. Viết hàm nhập xuất danh sách các môn thể thao dựa vào mối quan hệ kế thừa trên (Khuyến khích dùng Đa Hình) (Xem lại lý thuyết để làm)

**Xây dựng lớp Football, Tenis, và lớp Volleyball với các thông tin số lượng người chơi, thời gian chơi, loại bóng chơi kèm hai phương thức nhập và xuất.**

**( em thêm 2 thông tin là số trận đấu (match) và số banh dùng (ball)**

Hàm Main:

A computer screen shot of text

Description automatically generated

Bắt Exception:

A computer screen with text

Description automatically generated

Lớp Football:

A screen shot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated

A computer screen shot of a program code

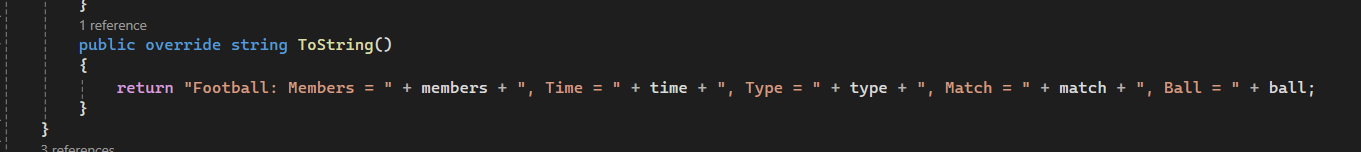
Description automatically generated

A screen shot of a computer program

Description automatically generated

A computer screen with text on it

Description automatically generated



Các lớp Tennis và Volleyball em làm tương tự.

Kết quả sau khi chạy:

A computer screen with white text

Description automatically generated

**Suy nghĩ lớp cha Sport chứa các thông tin gì để các lớp con có thể kế thừa các thông tin đó:** lớp cha Sport sẽ chứa các thông tin mà đề bài yêu cầu (số lượng người chơi members, thời gian chơi time, loại bóng chơi type. Các lớp con sẽ kế thừa các đặc tính của lớp cha Sport có thể thêm các đặc tính riêng cho các môn thể thao như UsingType(tay, chân, vợt), Ball( số lượng bóng chơi), Match(số lượng trận đấu).

**Viết hàm nhập xuất danh sách các môn thể thao dựa vào mối quan hệ kế thừa trên (Khuyến khích dùng Đa Hình) (Xem lại lý thuyết để làm)**

Hàm main:

A computer screen shot of a program code

Description automatically generated

Lớp cha Sport:

A screen shot of a computer program

Description automatically generated

A black background with white text

Description automatically generated

Lớp con Football

A screen shot of a computer program

Description automatically generated

Lớp con Tennis:

A screen shot of a computer program

Description automatically generated

Lớp con Volleyball:

A screen shot of a computer code

Description automatically generated

**Kết quả sau khi chạy**

A screen shot of a computer

Description automatically generated

Giải thích:

foreach (var sport in sports) { sport.InputInfo(); // Đa hình: phương thức InputInfo của từng lớp con sẽ được gọi }

foreach (var sport in sports) { sport.DisplayInfo(); // Đa hình: phương thức DisplayInfo của từng lớp con sẽ được gọi }

Khi gọi sport.InputInfo() và sport.DisplayInfo() trong vòng lặp foreach, chương trình sẽ gọi đúng phương thức của từng lớp con do tính đa hình, ngay cả khi biến sport có kiểu dữ liệu là Sport.

Cho phép các lớp con triển khai các hành vi riêng biệt thông qua phương thức ảo (virtual) và ghi đè (override).

Làm thêm: Sử dụng Interface để định nghĩa hành vi cụ thể cho môn thể thao

A computer screen shot of a program code

Description automatically generated

A screen shot of a computer code

Description automatically generated