using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace FactoryDesignePattern

{

//abstract class Shape

//{

// public int X { get; set; }

// public int Y { get; set; }

// public Shape()

// {

// }

// public abstract Shape Clone();

//}

//class Rectangle : Shape

//{

// public int Width { get; set; }

// public int Height { get; set; }

// public Rectangle(int x,int y,int width,int height)

// {

// X = x;

// Y = y;

// Width = width;

// Height = height;

// }

// public Rectangle(Rectangle rectangle)

// {

// X=rectangle.X;

// Y=rectangle.Y;

// Width = rectangle.Width;

// Height = rectangle.Height;

// }

// public Rectangle()

// {

// }

// public override Shape Clone()

// {

// return new Rectangle(this);

// }

//}

//class Circle : Shape

//{

// public int Radius { get; set; }

// public Circle(int x, int y, int radius)

// {

// X = x;

// Y = y;

// Radius = radius;

// }

// public Circle(Circle circle)

// {

// X=circle.X;

// Y=circle.Y;

// Radius=circle.Radius;

// }

// public override Shape Clone()

// {

// return new Circle(this);

// }

//}

//class Program

//{

// static void Main(string[] args)

// {

// //Rectangle rect = new Rectangle(10, 20, 100, 300);

// //var rect\_c = rect;

// //var rect\_copy = rect.Clone();

// //Console.WriteLine(rect.X);

// }

//}

class Program

{

static void Main(string[] args)

{

ConcreteProduct\_2 concreteProduct\_2 = new ConcreteProduct\_2(10);

ConcreteProduct\_1 concreteProduct\_1 = new ConcreteProduct\_1();

concreteProduct\_1.Method();

concreteProduct\_2.Method();

Console.WriteLine();

var Copy = (ConcreteProduct\_2)concreteProduct\_2.Clone();

concreteProduct\_2.Method();

concreteProduct\_2.Value = 20;

Copy.Method();

concreteProduct\_2.Method();

}

}

interface IPrototype

{

object Clone();

}

interface IProduct

{

void Method();

}

class ConcreteProduct\_1 : IProduct

{

public void Method()

{

Console.WriteLine($"Hash: {this.GetHashCode()}");

}

}

class ConcreteProduct\_2 : IProduct, IPrototype

{

public int Value;

public ConcreteProduct\_2(int value)

{

this.Value = value;

}

public object Clone()

{

return (ConcreteProduct\_2)this.MemberwiseClone();

}

public void Method()

{

Console.WriteLine($"Hash: {this.GetHashCode()} Value {Value}");

}

}

}