Crie um programa que peça pro usuário digitar 0 se a forma geométrica for quadrado, 1 se a forma geométrica for retângulo. (main) (tentem usar enum)  
Crie uma classe chamada Retangulo com 2 construtores. (retangulo)  
Um construtor vai receber somente o valor da largura e atribuir o mesmo valor para a altura, já que em um quadrado os lado são iguais. (retangulo)  
O outro construtor vai receber o lado da altura e da largura para o retângulo.  
Crie um método que calcule a área de retângulo chamado AreaDoRetangulo. (retangulo)  
Peça para o usuário digitar o lado do quadrado e crie um objeto do tipo Retangulo caso seja um quadrado. (main)  
Peça para o usuário digitar a largura e a altura e crie um objeto do tipo Retangulo caso seja um retangulo. (main)  
Chame o método AreaDoRetangulo e imprima a área do retângulo. (main)

using System; var pessoa = new Pessoa();  
pessoa.nome = "thays";  
pessoa.email = "thaystg@gmail.com"; Console.WriteLine(pessoa.nome);  
Console.WriteLine(pessoa.email);  
class Pessoa  
{  
public string nome; //atributos  
public string email; //atributos  
}

using System;  
var pessoa = new Pessoa("thays", "thaystg@gmail.com");  
var pessoa2 = new Pessoa("raquel");  
Console.WriteLine(pessoa.nome);  
Console.WriteLine(pessoa.email); Console.WriteLine(pessoa2.nome);  
Console.WriteLine(pessoa2.email); class Pessoa  
{  
public string nome;  
public string email;  
public Pessoa(string nome, string email)  
{  
this.nome = nome;  
this.email = email;  
} public Pessoa(string nome)  
{  
this.nome = nome;  
this.email = "nao informado";  
}  
}

**Enum**. O **Enum** é um conceito bastante útil para algumas áreas de programação. Na verdade trata-se da abreviação de Enumeração, que na prática nada mais é do que um grupo de constantes. A grande vantagem das **Enums** é que elas podem expressar números através de nomes.

num FormaDaFigura  
{  
Quadrado = 0,  
Retangulo = 1,  
Circulo = 2,  
Trapezio = 3  
}  
class Retangulo  
{  
public int largura;  
public int altura;

public Retangulo(int largura, int altura)  
{  
this.largura = largura;  
this.altura = altura;  
}  
public Retangulo(int lado)  
{  
this.largura = lado;  
this.altura = lado;  
}  
public int AreaDoRetangulo()  
{  
return largura \* altura;  
}  
}

enum FormaDaFigura  
{  
Quadrado = 0,  
Retangulo = 1,  
Circulo = 2,  
Trapezio = 3  
}  
class Retangulo  
{  
public int largura;  
public int altura;

public Retangulo(int largura, int altura)  
{  
this.largura = largura;  
this.altura = altura;  
}  
public Retangulo(int lado)  
{  
this.largura = lado;  
this.altura = lado;  
}  
public int AreaDoRetangulo()  
{  
return largura \* altura;  
}  
}

using System;  
var pessoa = new Pessoa("thays", "thaystg@gmail.com");  
var pessoa2 = new Pessoa("raquel");  
Console.WriteLine(pessoa.nome);  
Console.WriteLine(pessoa.email);

Console.WriteLine(pessoa2.nome);  
Console.WriteLine(pessoa2.email);

class Pessoa  
{  
public string nome;  
public string email;  
public Pessoa(string nome, string email)  
{  
this.nome = nome;  
this.email = email;  
}

public Pessoa(string nome)  
{  
this.nome = nome;  
this.email = "nao informado";  
}  
}

Console.WriteLine("Digite 0 se for quadrado, 1 se for retangulo");  
string formaStr = Console.ReadLine();  
int formaInt = Convert.ToInt32(formaStr);  
FormaDaFigura forma;

if (formaInt == 0)  
forma = FormaDaFigura.Quadrado;  
else  
forma = FormaDaFigura.Retangulo;

//FormaDaFigura forma = (FormaDaFigura) formaInt;

Retangulo retangulo;

if (forma == FormaDaFigura.Quadrado)  
{  
Console.WriteLine("digite o lado do quadrado");  
int lado = Convert.ToInt32(Console.ReadLine());  
retangulo = new Retangulo(lado);  
}  
else  
{  
Console.WriteLine("digite a altura do retangulo");  
int altura = Convert.ToInt32(Console.ReadLine());  
Console.WriteLine("digite a largura do retangulo");  
int largura = Convert.ToInt32(Console.ReadLine());  
retangulo = new Retangulo(altura, largura);  
}

Console.WriteLine($"A Area é {retangulo.AreaDoRetangulo()}");

class Assinatura  
{  
public static string Iniciais(string nomeCompleto)  
{  
/\*string ret = "";  
var nomeCompletoArr = nomeCompleto.ToCharArray();  
for (int i = 0; i < nomeCompletoArr.Length; i++)  
{  
if (i == 0 || nomeCompletoArr[i - 1] == ' ')  
{  
ret += nomeCompletoArr[i];  
}  
}  
return ret;\*/  
string ret = "";  
var nomeCompletoSplit = nomeCompleto.Split(' ');  
foreach(string s in nomeCompletoSplit)  
{  
ret += s[0];  
}  
return ret;  
}  
}

<https://www.hackerrank.com/challenges/simple-array-sum/problem?isFullScreen=true>

<https://www.hackerrank.com/challenges/mini-max-sum/problem?isFullScreen=true>

int maiorNumero = 0;

int menorNumero = 0;

long soma = 0;

for (int i = 0; i < arr.Count; i++)

{

soma += arr[i];

if (i == 0 || arr[i] > maiorNumero)

maiorNumero = arr[i];

if (i == 0 || arr[i] < menorNumero)

menorNumero = arr[i];

}

Console.WriteLine($"{soma - maiorNumero} {soma - menorNumero}");