

Programació orientada a objectes

Encapsulament

Aquest concepte consisteix en l'ocultació de l'estat o de les pròpies dades d'un objecte. De tal manera que només és possible modificar els mateixos mitjançant els mètodes definits per aquest objecte (com es veurà més endavant get/set).

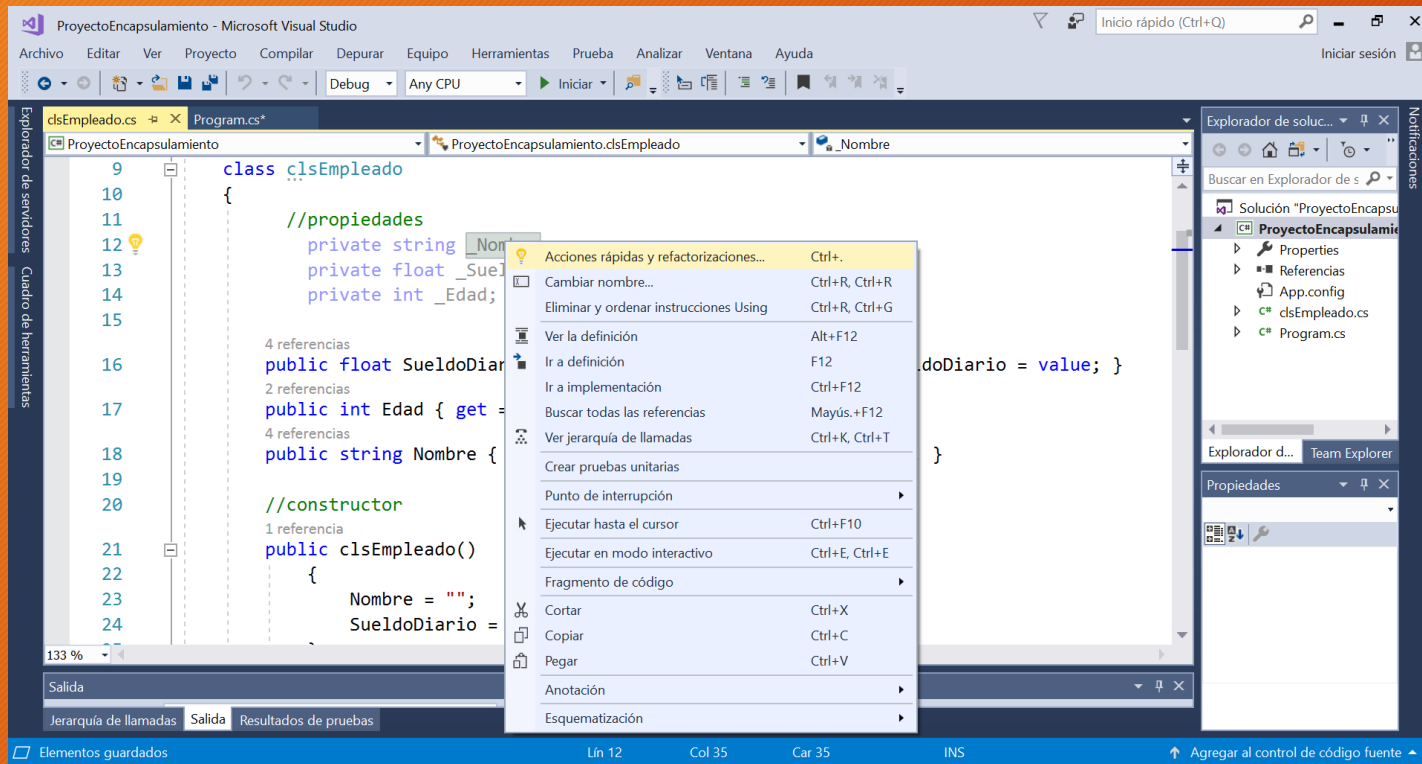
Hi han coses que s'han de fer des de la pròpia classe i no des de qualsevol classe.

Mètodes Getter / Setter:

- Set: estableix el valor d'una propietat
- Get: retorna el valor d'una propietat
- Un de cada per a cada propietat

Àmbit de les variables i mètodes

- public: es pot accedir des de fora de la classe, és a dir des de la propia instància de la classe.
- private: només es pot accedir des de dins de la propia classe , per un mètode de la classe. No es pot accedir des d'una instància.
- protected: només es pot accedir per la classe i les subclasses (herència). No es pot accedir des d'una instància.



```
//propiedades
```

```
private string _Nombre;
private float _SueldoDiario;
private int _Edad;
```

4 referencias

```
public float SueldoDiario { get => _SueldoDiario; set => _SueldoDiario = value; }
```

2 referencias

```
clsEmpleado empleado1 = new clsEmpleado();  
//accedemos desde el método no la propiedad  
empleado1.Nombre = "Pedro";  
empleado1.Edad = 33;  
empleado1.SueldoDiario = 80.0F;  
float Salario1 = empleado1.CalculaSueldo(15);  
//accedemos desde el método no la propiedad  
  
Console.WriteLine(empleado1.Nombre + ":");  
Console.WriteLine("El salario es:{0:F2}", Salario1);  
Console.ReadKey();
```


Exercici.

Fer la classe Facebook en la qual es desa el nom, la edat i el password.

Tenir en compte la modularització i l'encapsulament.

Fer com a mínim un constructor.

```

class clsFacebook
{
    private string _Nombre;
    private int _Edad;
    private string _Pass;

    1 referencia
    public clsFacebook(string n,int e,string p) {
        Nombre=n;
        Edad = e;
        Pass = p;
    }

    2 referencias
    public string Nombre { get => _Nombre; set => _Nombre = value; }

    3 referencias
    public int Edad { get => _Edad; set => _Edad = value; }

    1 referencia
    public string Pass { get => _Pass; set => _Pass = value; }
}

```



```
Console.WriteLine("Para dar de alta su cuenta introduzca los siguientes datos  
por favor..");  
Console.WriteLine("Su nombre?");  
string nombre = Console.ReadLine();  
string contra = Console.ReadLine();  
int laEdad = Convert.ToInt16(Console.ReadLine());  
Console.WriteLine("Su edad?");  
Console.WriteLine("Q password desea?");  
clsFacebook cuentaLaura = new clsFacebook(nombre, laEdad, contra);  
Console.WriteLine("Bienvenid@" + cuentaLaura.Nombre + " , edad:" + cuentaLaura.Edad  
+ ", tiene el password:" + cuentaLaura.Edad);  
  
Console.ReadKey();
```

Volem afegir un mètode que ens doni la informació en comptes de fer-ho nosaltres des del programa principal.

```
public string Pass { get => _Pass; set => _Pass = value; }  
1 referencia  
public string VerInfo() {  
    .  
    return "Bienvenid@" + this.Nombre + " , edad:" + this.Edad + ", tiene el  
    password:" + this.Edad; }  
    .
```

```
Console.WriteLine(cuentaLaura.VerInfo());  
    .
```

Crear la classe 'triangulo' tenint en compte els principis de la POO

Exercici

Passes:

1. Pensar en la classe (propietats)
2. Codi (encapsulament)
3. Crear una instància de la classe.

Tenint en compte l' encapsulament i la modularització.

Característiques d'un triangle:

- Altura (private)
- Base (private)

Un possible mètode:

- Calcular l'àrea (public)

Crear una instància de triangle donats
uns valors per consola.

Definir un mètode de la classe que ens doni la informació
de la seva àrea (public).

Sol