

PROGRAMACIÓN ORIENTADA A OBJETOS

3er. PARCIAL

Práctica 7

Polimorfismo y Herencia 2

Daniel Vázquez de la Rosa
d_vazquez@outlook.com

Instituto Politécnico Nacional



Escuela Superior de Cómputo

Septiembre 2018

Datos Personales

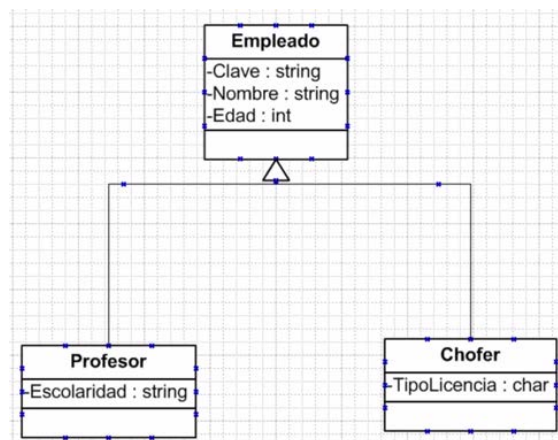
Crear un Documento en Word que incluya una caratula que contenga tus datos. Dicho documento será guardado como: Grupo_ApellidoPaterno_NombreAlumno_Practica7_POO y deberá contener toda la información de la practica creada.

Objetivos de la Práctica

- Comprender los fundamentos de C# como el Polimorfismo y la Herencia.
- Entender la creación de elementos dentro del polimorfismo y la Herencia en la Programación Orientada a Objetos.

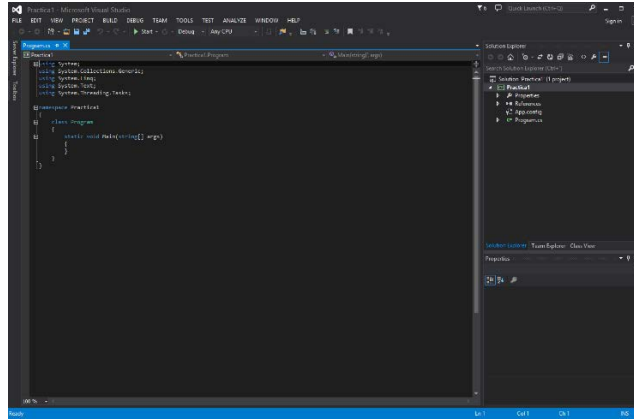
Desarrollo

1. El programa que se pretende hacer esta descrito en el siguiente diagrama UML:

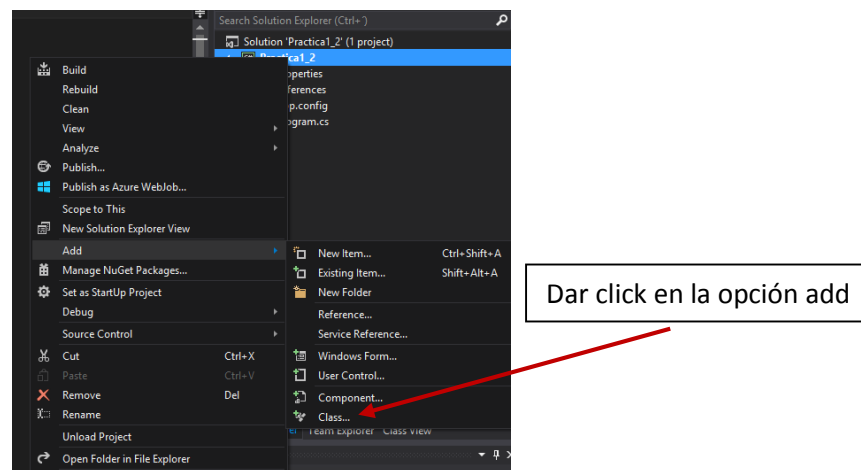


2. Al abrir Visual Studio Community podrás acceder a la creación de proyectos, da click en New Project, al hacer este click te mostrara una ventana donde tendrás que elegir el lenguaje de programación a utilizar, así como, asignar un nombre al proyecto o practica que realizaras y seleccionar la ubicación donde guardaras el proyecto.

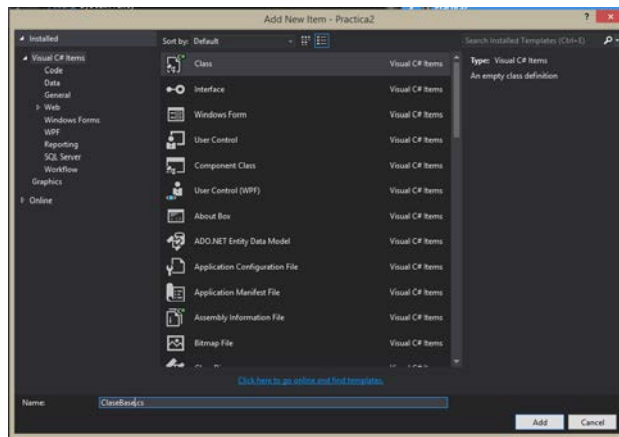
3. Selecciona en los lenguajes de la parte Izquierda a Visual C#, en la parte central trabajaremos con las aplicaciones de consola (Console Application), el nombre que le darás a esta práctica será Practica7 y te recomiendo crear una carpeta en C: llamada practicas C#, debido que en el apartado de location tendrás que direccionar tu proyecto a esa carpeta a través del botón Browse como se muestra en la imagen.
4. Una vez que has hecho los cambios, da click en Ok, se creara el proyecto y podrás comenzar a realizar la aplicación que desees.



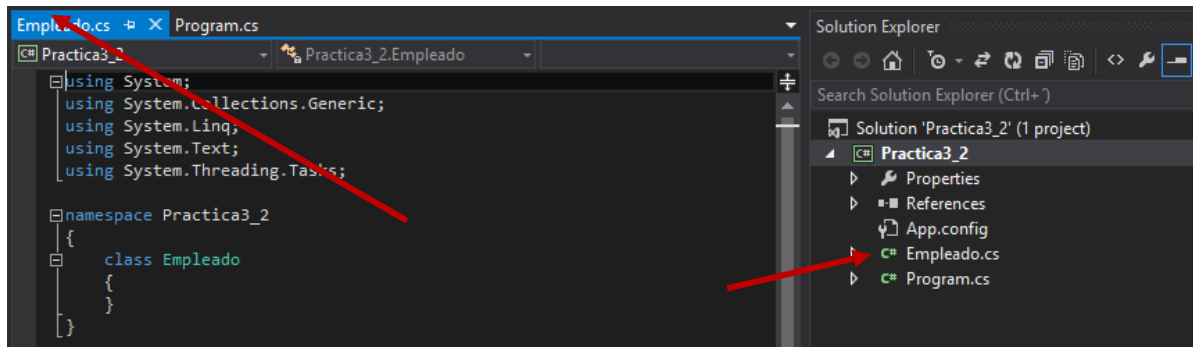
5. Para esta práctica tendrás que crear una clase llamada Empleado (que será nuestra clase padre), para poder hacer esto, en la ventana de lado izquierdo localiza el nombre Practica7 y da click con el botón derecho del mouse, aparecerá un menú donde localizaras la opción add(agregar) y dentro de add, localizaras la opción class



Al dar click en class, aparecerá una ventana, donde tendrás que tener seleccionada la opción de class y en nombre escribir Empleado.cs

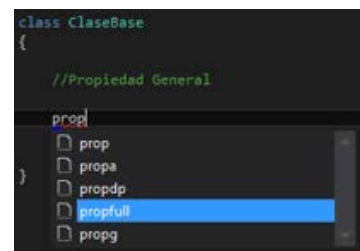
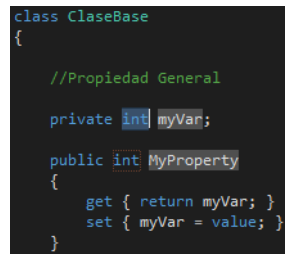
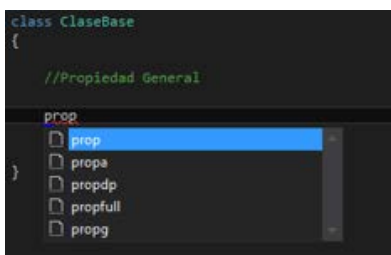


Una vez hecho lo anterior da click en Add y observarás que la clase Cuenta se ha creado a través de las fichas del proyecto o en la ventana izquierda.



6. Ahora selecciona la pestaña(ficha) Empleado.cs, y dentro de la clase Padre se creará una propiedad general de la siguiente forma:

Teclea la palabra propfull y el sistema te mostrara una lista de propiedades, selecciona propfull y da enter, posteriormente presiona la tecla tabulador. Puedes utilizar Ctrl + R + E.



Y cambiaremos la propiedad, para que quede de la siguiente forma:

```
class Empleado
{
    private int clave;

    public int Clave
    {
        get { return clave; }
        set { clave = value; }
    }
}
```

Posteriormente crearemos dos métodos más que llamaremos Profesor y Chofer y acomodaremos el código para que se vea de la siguiente forma:

```
class Empleado
{
    private string clave;
    private string nombre;
    private int edad;

    public string Clave
    {
        get { return clave; }
        set { clave = value; }
    }

    public string Nombre
    {
        get { return nombre; }
        set { nombre = value; }
    }

    public int Edad
    {
        get { return edad; }
        set {
            //La edad valida será de 0 a 100
            //una persona con más de 100 años
            //será invalidada por el programa
            if (value > 0 && value <= 100)
            {
                edad = value;
            }
            else
            {
                edad = 0;
            }
        }
    }

    //Constructor
    public Empleado(string claveE,
                    string nombreE,
                    int edadE)
    {
        this.Clave = claveE;
        this.Nombre = nombreE;
        this.Edad = edadE;
    }
}
```

7. Se creara una clase nueva llamada Profesor que heredara de la clase Empleado, y dentro de la clase se escribirá el siguiente código:

```

class Profesor:Empleado
{
    private string escolaridad;

    public string Escolaridad
    {
        get { return escolaridad; }
        set { escolaridad = value; }
    }

    //Constructor

    public Profesor(string claveP,
                    string nombreP,
                    int edadP,
                    string escolaridadP)
    :
    base(claveP,nombreP,edadP)
    {
        this.Escolaridad = escolaridadP;
    }
}

```

8. Se creara una clase nueva llamada Chofer que heredara de la clase Padre Empleado.

```

class Chofer:Empleado
{
    private char licencia;

    public char Licencia
    {
        get { return licencia; }
        set { char licencia = value; }
    }

    //Constructor

    public Chofer( string claveC,
                  string nombreC,
                  int edadC,
                  char licenciaC)
    :
    base(claveC, nombreC, edadC)
    {
        this.Licencia = licenciaC;
    }
}

```

9. Una vez creado las dos clases derivadas, se procede a crear los objetos en la clase principal (Program).

```

class Program
{
    static void Main(string[] args)
    {
        Empleado empleado1 = new Empleado("0123", "Isabel Hernández", 25);
        Profesor profesor1 = new Profesor("1111", "Adolfo Grovas", 28, "Maestria");
        Chofer chofer1 = new Chofer("123123", "Arturo Flores", 45, 'A');

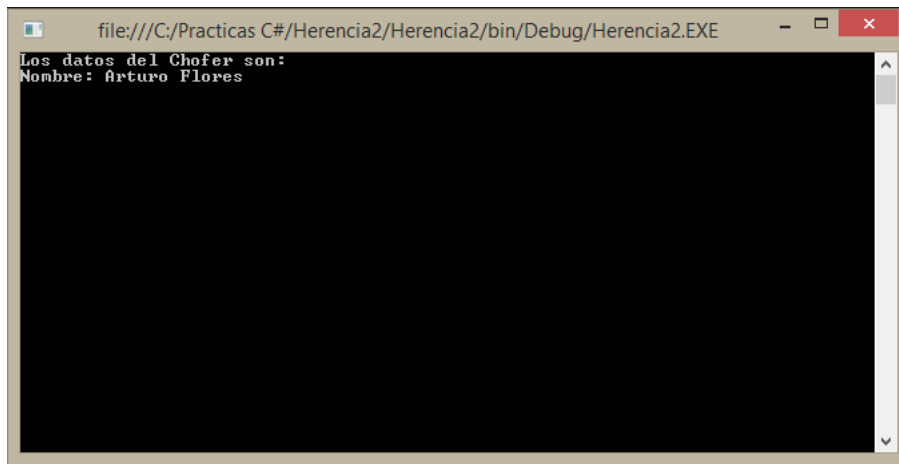
        Console.WriteLine("Los datos del Chofer son:");
        Console.WriteLine("Nombre: {0}",chofer1.Nombre);

        Console.WriteLine("");

        Console.ReadKey();
    }
}

```

Ejecutar con F5 y el resultado debe de ser:



Responde:

- i) ¿Cuál es la diferencia de este programa con relación a los anteriores que has utilizado?
- ii) Una vez creados el programa de la práctica, deberás de completar el programa que permita imprimir los datos de un empleado el cual puede ser: Profesor, Intendencia, Administrativo, Chofer, donde tendrás que crear tres objetos de cada clase y a través de un **menú** poder seleccionar la operación capturar o imprimir datos del profesionista.
- iii) Crea un programa que permita imprimir los datos de una clase padre llamada Persona con tres clases que serán Empleado, Estudiante, Abuelo, a su vez, la clase Empleado será una clase padre para dos clases que serán, Constructor y Profesor, donde tendrás que crear tres objetos de cada clase y a través de un **menú** poder seleccionar la operación capturar o imprimir datos del objeto seleccionado.

Estas Prácticas deberán de entregarse a más tardar el día Jueves 04/10/2018 antes de las 10:00 PM, toda práctica que no sea enviada dentro de este horario será evaluada con cero.