



GUÍA DE APRENDIZAJE N° 9

1. IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE

Programa de Formación: Técnico en Programación de Software	Código: Versión:	228120 102		
Nombre del Proyecto: Sistema de información para la gestión empresarial v1.3	Código:	704330		
Fase del proyecto:	Fase 1. Análisis			
Actividad (es) del Proyecto: Recolección de información y análisis de requerimientos.	Actividad (es) de Aprendizaje: Diseñar la arquitectura del software utilizando la programación orientada a objetos (POO)	Ambiente de formación ESCENARIO	MATERIALES DE FORMACIÓN	
			DEVOLUTIVO	CONSUMIBLE
Resultados de Aprendizaje: 22050103201 Identificar cada uno de los conceptos y principios que constituye la programación orientada a objetos para interpretar el diseño.	Competencia: 220501032 Analizar los requerimientos del cliente para construir el sistema de información.	Sala de informática con Conexión a internet banda ancha (mínimo 1Mb por equipo) Cableado estructurado. Equipos de cómputo con hardware y software actualizado (Un equipo por aprendiz). Mesas y Sillas ergonómicas. Ventilación e iluminación natural y/o artificial. Video beam, Tablero acrílico y/o digital	Equipos Portátiles y/o de escritorio. Video beam. Medios de almacenamiento extraíbles (Cd's, Usb y demás)	Marcadores de colores. Tinta de recarga. Borrador. Resaltador. Esferos y lápices.



Duración de la guía (en horas):	30			

2. INTRODUCCIÓN

En ocasiones es necesario restringir el acceso y la manipulación de datos y variables específicas con el ánimo de proteger la integridad del sistema.

Los modificadores de acceso permiten al desarrollador de clases establecer los límites entre la accesibilidad de los usuarios de la clase, y lo que es estrictamente privado.

De <http://www.arrakis.es/~abelp/ApuntesJava/ModificadoresAcceso.htm>

Con el uso de estos modificadores se consigue uno de los principios básicos de la Programación Orientada a Objetos, que es la encapsulación: Las clases tienen un comportamiento definido para quienes las usan conformado por los elementos que tienen un acceso público, y una implementación oculta formada por los elementos privados, de la que no tienen que preocuparse los usuarios de la clase.

Cuando se diseñan clases, es importante pararse a pensar en términos de quien debe tener acceso a qué. Qué cosas son parte de la implantación y deberían ocultarse (y en qué grado) y que cosas forman parte de la interface y deberían ser públicas.

Modificadores de Acceso

3. ESTRUCTURACION DIDACTICA DE LAS ACTIVIDADES DE APRENDIZAJE

3.1 Actividades de Reflexión inicial.

En la vida cotidiana nos encontramos con restricciones que impiden o limitan el acceso a diferentes escenarios. Por ejemplo, al ingresar a un restaurante podemos desplazarnos por el rededor de las mesas y sillas, incluso tenemos acceso a los baños pero al intentar ingresar al área de cocina seguramente nos encontraremos con un aviso **“solo personal autorizado”**.

En otros casos la restricción se da al comportamiento dentro del escenario. Por ejemplo, **“no usar el celular dentro de un banco”**. Claro esto no lo pueden hacer los usuarios sin embargo el gerente y otros administrativos gozan de este derecho.

De igual manera en el desarrollo de software, es necesario restringir la accesibilidad a algunos módulos,



datos y variables con el fin de proteger la integridad de la información.

Con base a la reflexión anterior, Participe en el foro “**Modificadores de acceso**” y redacte un ejemplo acerca de una restricción común con la que nos podamos encontrar. No olvide revisar y comentar los aportes de los compañeros.

3.2 Actividades de contextualización e identificación de conocimientos necesarios para el aprendizaje.

En software la distinción más evidente de estas restricciones se encuentra en la asignación de variables globales y locales, pero si observamos más afondo nos daremos cuenta que los métodos, miembros, funciones y clases entre otros también están limitados por estas restricciones o accesos.

Organice las letras en las casillas y encuentre las palabras ocultas

IVPRBELOTCPRUCTEDPTEIA

3.3 Actividades de apropiación del conocimiento (Conceptualización y Teorización).

De <https://msdn.microsoft.com/es-es/default.aspx>

3.3.1 Los modificadores de acceso

Son palabras clave que especifican la accesibilidad declarada de un miembro o un tipo.

Todos los tipos y miembros de tipo tienen un nivel de accesibilidad, que controla si pueden utilizarse por otro código de su ensamblado u otros ensamblados.

Los cuatro modificadores de acceso son:

Public, protected, internal, private

3.3.2. Niveles de accesibilidad (Referencia de C#)

Utilice los modificadores de acceso, public, protected, internal o private para especificar uno de los siguientes niveles de accesibilidad declarados para miembros.

Accesibilidad declarada	Significado
-------------------------	-------------



public	Acceso no restringido.
protected	Acceso limitado a la clase contenedora o a los tipos derivados de esta clase.
internal	Acceso limitado al ensamblado actual.
protected internal	Acceso limitado al ensamblado actual o a los tipos derivados de la clase contenedora.
private	Acceso limitado al tipo contenedor.

Sólo se permite un modificador de acceso para un miembro o tipo, excepto cuando se utiliza la combinación `protectedinternal`.

Los modificadores de acceso no se pueden utilizar en espacios de nombres. Los espacios de nombres no presentan restricciones de acceso.

Según el contexto en el que se produce una declaración de miembro, sólo se permite declarar ciertos tipos de acceso. Si no se especifica ningún modificador de acceso en una declaración de miembro, se utiliza el tipo de acceso predeterminado.

Los tipos de nivel superior, que no están anidados en otros tipos, sólo pueden tener accesibilidad `internal` o `public`. La accesibilidad predeterminada para estos tipos es `internal`.

Los tipos anidados, que son miembros de otros tipos, pueden tener un tipo de acceso declarado como el que se indica en la siguiente tabla.

Miembros de	Accesibilidad predeterminada	Accesibilidades declaradas permitidas
enum	public	None
class	private	public protected internal private protected internal
interface	public	None



struct

private

public

internal

private

La accesibilidad de un tipo anidado depende de su dominio de accesibilidad, el cual se determina mediante la accesibilidad declarada para el miembro y el dominio de accesibilidad del tipo inmediato que lo contiene. Sin embargo, el dominio de accesibilidad de un tipo anidado no puede superar al del tipo contenedor.

3.3.3 Dominio de accesibilidad (Referencia de C#)

El dominio de accesibilidad de un miembro especifica en qué secciones del programa se puede hacer referencia a dicho miembro. Si el miembro está anidado dentro de otro tipo, su dominio de accesibilidad viene determinado por el nivel de accesibilidad del miembro y por el dominio de accesibilidad del tipo contenedor inmediato.

El dominio de accesibilidad de un tipo de nivel superior es por lo menos el texto del programa del proyecto en el que se declara. Es decir, el dominio incluye todos los archivos de origen de este proyecto. El dominio de accesibilidad de un tipo anidado es, al menos, el texto del programa en el que el tipo aparece declarado. Es decir, el dominio es el cuerpo del tipo, que incluye todos los tipos anidados. El dominio de accesibilidad de un tipo anidado no puede superar al del tipo contenedor. Estos conceptos se muestran en el siguiente ejemplo.

3.3.4 Restricciones en el uso de los niveles de accesibilidad (Referencia de C#)

Cuando se especifica un tipo en una declaración, compruebe si el nivel de accesibilidad del tipo es dependiente del nivel de accesibilidad de un miembro o de otro tipo. Por ejemplo, la clase base directa debe ser al menos tan accesible como la clase derivada. Las siguientes declaraciones producirán un error del compilador, ya que la clase base BaseClass es menos accesible que MyClass:

```
class BaseClass {...}
public class MyClass: BaseClass {...} // Error
```

La siguiente tabla resume las restricciones de uso de los niveles de accesibilidad declarados.

Contexto	Comentarios
Clases	La clase base directa de un tipo de clase debe ser al menos tan accesible como el propio tipo de clase.
Interfaces	Las interfaces base explícitas de un tipo de interfaz deben ser al menos tan accesibles como el propio tipo de interfaz.



Delegados	El tipo de valor devuelto y los tipos de los parámetros de un tipo delegado deben ser al menos tan accesibles como el propio tipo delegado.
Constantes	El tipo de una constante debe ser al menos tan accesible como la propia constante.
Campos	El tipo de un campo debe ser al menos tan accesible como el propio campo.
Métodos	El tipo de valor devuelto y los tipos de parámetros de un método deben ser al menos tan accesibles como el propio método.
Propiedades	El tipo de una propiedad debe ser al menos tan accesible como la misma propiedad.
Eventos	El tipo de un evento debe ser al menos tan accesible como el propio evento.
Indizadores	El tipo y los tipos de parámetros de un indizador deben ser al menos tan accesibles como el propio indizador.
Operadores	El tipo de valor devuelto y los tipos de parámetros de un operador deben ser al menos tan accesibles como el propio operador.
Constructores	Los tipos de parámetros de un constructor deben ser al menos tan accesibles como el propio constructor.

Documentación. Modificadores de acceso(Referencia de C#)

Con base en la lectura anterior, realice un mapa conceptual.

Implemente y haga un análisis de los ejemplos propuestos en el msdn de Microsoft los cuales encontrara en los siguientes enlaces.

Internal <https://msdn.microsoft.com/es-es/library/7c5ka91b.aspx>

Private <https://msdn.microsoft.com/es-es/library/st6sy9xe.aspx>

Protected <https://msdn.microsoft.com/es-es/library/bcd5672a.aspx>

Public <https://msdn.microsoft.com/es-es/library/yzh058ae.aspx>

3.4 Actividades de transferencia del conocimiento.

Implemente los modificadores de acceso en el desarrollo de su proyecto de formación.

3.5 Actividades de evaluación.

Evidencias de Aprendizaje	Criterios de Evaluación	Técnicas e Instrumentos de Evaluación
---------------------------	-------------------------	---------------------------------------



SERVICIO NACIONAL DE APRENDIZAJE SENA

GUÍA DE APRENDIZAJE

SISTEMA INTEGRADO DE GESTIÓN

Proceso Gestión de la Formación Profesional Integral

Procedimiento Ejecución de la Formación Profesional Integral

Versión: 02

Código: GFPI-F-019

<p>Evidencias de Conocimiento :</p> <p>Participa en el foro de discusión</p>	<p>Implementa los modificadores de acceso de acuerdo a los requerimientos del proyecto de formación.</p> <p>Elabora propuestas de trabajo, de acuerdo con la interpretación de las necesidades tecnológicas, expuestas en el informe de requerimientos, según Normas y protocolos de la empresa.</p> <p>Realiza el informe de análisis del sistema de información requerido, presentando el estado actual de la organización, y proponiendo alternativas de solución, de acuerdo con las normas y protocolos establecidos.</p>	<p>Rubrica TIGRE</p>
<p>Evidencias de Desempeño:</p> <p>Implementa los ejemplos propuestos en el msdn</p>		<p>Lista de chequeo</p>
<p>Evidencias de Producto:</p> <p>Implementa los modificadores de acceso en el proyecto formativo.</p>		<p>Lista de chequeo</p>



4. RECURSOS PARA EL APRENDIZAJE

ACTIVIDADES DEL PROYECTO	DURACIÓN (Horas)	Materiales de formación devolutivos: (Equipos/Herramientas)		Materiales de formación (consumibles)		Talento Humano (Instructores)		AMBIENTES DE APRENDIZAJE TIPIFICADOS
		Descripción	Cantidad	Descripción	Cantidad	Especialidad	Cantidad	ESCENARIO (Aula, Laboratorio, taller, unidad productiva) y elementos y condiciones de seguridad industrial, salud ocupacional y medio ambiente
	30	Equipos de escritorio y/o Portátiles. Video beam y o tablero digital. Medios de almacenamiento extraíbles (Cd's, Usb y demás)	1 por aprendiz. 1 1 por aprendiz.	Marcadores de colores. Tinta de recarga. Borrador. Resaltador. Esferos y lápices.	4 4 1 2 2	Instructor de teleinformática.	1	Sala de informática con Conexión a internet banda ancha (mínimo 1Mb por equipo) Cableado estructurado. Equipos de cómputo con hardware y software actualizado (Un equipo por aprendiz). Mesas y Sillas ergonómicas. Ventilación e iluminación natural y/o artificial. Video beam, Tablero acrílico y/o digital



5. GLOSARIO DE TERMINOS

PUBLIC: es un modificador de acceso para tipos y miembros de tipos. El acceso de tipo public corresponde al nivel de acceso menos restrictivo. No existen restricciones para obtener acceso a los miembros públicos.

PROTECTED: es un modificador de acceso a miembros. Un miembro protegido es accesible dentro de su clase y por instancias de clases derivadas.

INTERNAL: es modificador de acceso para los tipos y miembros de tipo. Los tipos o miembros internos sólo son accesibles dentro de los archivos del mismo ensamblado.

PRIVATE: es un modificador de acceso de miembros. El acceso de tipo private corresponde al nivel de acceso más restrictivo. Los miembros privados sólo son accesibles dentro del cuerpo de la clase o struct en los que se declaran.

6. REFERENTES BIBLIOGRÁFICOS

Modificadores de Acceso

<http://www.arrakis.es/~abelp/ApuntesJava/ModificadoresAcceso.htm>

[Consulta: Miércoles, 25 de Febrero de 2015] © Antonio Bel Puchol

Modificadores de Acceso

<https://msdn.microsoft.com/es-es/library/wxh6fsc7.aspx>

[Consulta: Viernes, 27 de Febrero de 2015] © 2015 Microsoft

Modificadores de Acceso

<http://elvex.ugr.es/decsai/java/pdf/9D-access.pdf> Pg 18-20

[Consulta: Viernes, 27 de Febrero de 2015] © Fernando Berzal



SERVICIO NACIONAL DE APRENDIZAJE SENA

GUÍA DE APRENDIZAJE

SISTEMA INTEGRADO DE GESTIÓN

Proceso Gestión de la Formación Profesional Integral

Procedimiento Ejecución de la Formación Profesional Integral

Versión: 02

Código: GFPI-F-019

7. CONTROL DEL DOCUMENTO (ELABORADA POR)

Edwin Cañón Mozo