



Cuestionario Teórico - Clase 18

*Obligatorio

Cuestionario Teórico - Clase 18

Interfaces

¿Qué es una interfaz? ¿Qué implica implementarla en una clase? *

tienen similitud a las clases abstractas, declaran solo la firma de metodos o propiedades que luego deben ser desarrollados obligatoriamente por clases que implementen dicha interfaz

¿Qué miembros puede especificar una interfaz? (atributos, métodos, propiedades, etc) *

Solo se pueden especificar metodos, propiedades, indexadores y eventos , atributos no

¿Qué nivel de visibilidad/acceso pueden tener los miembros especificados en la interfaz? *

no llevan identificador de visibilidad, pero cuando se implementan en las clases si o si tienen que ser publicos

¿Una clase puede implementar más de una interfaz? De un ejemplo de la sintaxis. *

una clase puede implementar una o mas interfaces

public class ClaseImplementadora : UnaInterfaz, OtraInterfaz()

¿Se puede elegir cuáles de las operaciones definidas en la interfaz implementar en una clase? *

☐ Sí

☒ No

¿Una clase puede heredar de otra y, al mismo tiempo, implementar una o más interfaces? Describa la sintaxis para lograr esto. *

se puede heredar de otras clases y a la vez implementar interfaces, la clase a heredar debe especificarse antes de las interfaces

public class ClaseImplementadora : UnaClase, UnaInterfaz, OtraInterfaz()

¿Qué pasa con las clases derivadas cuando una clase base implementa una interfaz? ¿Debo implementarla también en las derivadas? ¿Por qué? *

no deben implementarlas ya que la implementacion de la interfaz esta en la clase base y las clases derivadas heredan esos metodos ya implementados

¿Una interfaz puede implementar otra interfaz? ¿Qué sucede con las operaciones especificadas (se suman o se anulan)? Describa cómo sería la sintaxis y cómo se comportaría. *

Una interfaz puede implementar otra interfaz , los metodos heredados de la interfaz "base" no se anulan, sino que se suman, se deben implementar todos los metodos declarados en ambas interfaces

```
public interface Una Interfaz {  
  
    void UnMetodo();  
}
```

```

public interface Otrainterfaz : Unainterfaz {

    void OtroMetodo();

}

public class UnaClase : Otrainterfaz{

    public void UnMetodo(){}
    public void OtroMetodo(){}

}

```

¿Qué significa implementar una interfaz de forma explícita? ¿Qué utilidad tiene? ¿Qué consecuencias o efectos tiene? *

Cuando implementamos dos interfaces distintas que comparten la misma firma para un metodo, uno de ellos debe ser implementado de forma explícita, es decir, anteponiendo primero el nombre de la interfaz seguido de un punto y luego el nombre del metodo. La consecuencia de declarar un metodo de forma explícita es que luego en la llamada al metodo solo lo podremos invocar a traves de una referencia del tipo de la interfaz

¿Se pueden tener interfaces genéricas? ¿Se puede restringir sus parámetros de tipo? ¿Puede implementarse cualquier clase o sólo clases que también sean genéricas? De ejemplos de sintaxis para todos los casos. *

Es posible tener interfaces genericas y tambien se puede restringir sus parametros. Puede ser implementadas por clases no genericas

```

public interface Unainterfaz<T>{

    void UnMetodo(T parametro, int num);

}

public class UnaClase : Unainterfaz<string>{

    public void UnMetodo(string parametro, int num){}

}

```

¿En qué se diferencian una interfaz y una clase abstracta? *

La principal diferencia es que, por lo menos en c#, solo se acepta herencia simple y por consecuencia solo se puede heredar una clase, se pueden declarar mas cosas, permiten tener modificadores de acceso en sus metodos/propiedades/atributos y pueden contener miembros estaticos

Las interfaces se pueden implementar tantas como sea necesario, no hay un limite, estas permiten declarar menos cosas, no acepta modificadores de visibilidad y no pueden contener miembros estaticos

Si un método recibe un parámetro del tipo de una interfaz...

Considere esta condición para las siguientes preguntas.

¿Qué condición deberán cumplir los argumentos que le pase? *

que implementen la interfaz

Si no casteo el argumento a otro tipo, ¿podré acceder a todos los métodos y propiedades del objeto? *

sino casteo solo puedo acceder a los que estén declarados en la interfaz, para acceder a metodos y propiedades del objeto es necesario castear

Considerando que una interfaz no tiene implementación, ¿si llamo a uno de sus métodos qué implementación se ejecutará? Asocie con polimorfismo. *

se llamara a la implementacion del tipo del objeto en memoria

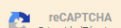
Se enviará una copia de tus respuestas por correo electrónico a la dirección que has proporcionado.

[Atrás](#)

[Enviar](#)

Página 2 de 2

Nunca envíes contraseñas a través de Formularios de Google.



Este contenido no ha sido creado ni aprobado por Google. [Notificar uso inadecuado](#) - [Términos del Servicio](#) - [Política de Privacidad](#)

Google Formularios

