Guía de Estudio: Abstract y Virtual

1. ¿Cuáles son las diferencias entre sobrecargar (overload) y sobrescribir (override) un método?   
   Overload es para sobrecargar un método, cambiando el orden, cantidad o tipo de dato de sus parámetros, es algo opcional. Overrida a diferencia, es para sobreescribir el comportamiento del método, generalmente heredado de otra clase, a veces abstracto, obligado a realizar.

|  |  |  |
| --- | --- | --- |
| **Criterio** | **Sobrecargar / Overload** | **Sobrescribir / Override** |
| Firma (Diferencias o no diferencias en las firmas) | Int hacerAlgo(float a, float b);  Int hacerAlgo(int a, int b); | Virtual int hacerAlgo(int n);  Override int hacerAlgo(int n); |
| Ubicación (Misma clase / Clases diferentes) |  |  |
| Tiempo de resolución (Compilación /  Ejecución) |  |  |
| Tipo objeto / Tipo Referencia (¿Qué determina cuál implementación se utilizará?) |  |  |

1. Si quiero declarar un método que **pueda** ser sobrescrito en las clases derivadas, ¿qué modificador debo usar?   
   virtual

1. Si quiero declarar un método que **deba** ser sobrescrito en las clases derivadas, ¿qué modificador debo usar?   
   Abstract

1. Las clases no-abstractas que derivan de una clase abstracta, ¿**deben** implementar **todos** sus métodos abstractos?  
   Sí.

1. Las clases abstractas que derivan de una clase abstracta, ¿**deben** implementar **todos** sus métodos abstractos?   
   Sí.

1. ¿Se pueden declarar miembros abstractos en clases no-abstractas?

No, porque necesitan de un cuerpo.

1. ¿Para sobrescribir un método se debe heredar de una clase abstracta?

No necesariamente.

1. Llenar los campos de la siguiente tabla con SÍ o NO según corresponda.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tipo de Clase** | **Puede heredar de otras clases** | **Puede heredarse**  **de ella** | **Puede ser instanciada** |
| ***normal (sin modificadores)*** | Sí | Sí | Sí |
| ***abstract*** | Sí | Sí | No |
| ***sealed*** | Sí | No | Sí |
| ***static*** | No | No | No |