



Herencia

JAVA

Herencia

Herencia

Según la UNAM (2022), la herencia es un mecanismo que permite definir una clase en función de la definición de otra clase existente. La herencia ayuda a compartir automáticamente métodos y datos entre clases, subclases y objetos.

La herencia se encuentra estrechamente relacionada con la reutilización de código en la Programación Orientada a Objetos. Esto significa que el código de cualquier clase se puede usar simplemente creando una clase o subclase derivada de ella.

Existen dos tipos de herencia (UNAM, 2022):

- Herencia simple: indica que las nuevas clases solo se pueden definir en base a una clase inicial.
- Herencia múltiple: indica que las nuevas clases se pueden definir en base a dos o más clases originales.

Resulta importante señalar que Java solo permite herencia simple.

Superclase y Subclases

El concepto de herencia conduce a una estructura de clases jerárquica o estructura de árbol; por tanto, en la Programación Orientada a Objetos, todas las relaciones entre clases deben seguir esta estructura (UNAM, 2022).

En dicha jerarquía, cada clase tiene solo una clase padre. La clase padre de una clase se conoce como superclase. Las clases hijas de las superclases se denominan subclases (UNAM, 2022).

* Una superclase puede tener cualquier número de subclases.

* Una clase secundaria solo puede tener una clase principal.

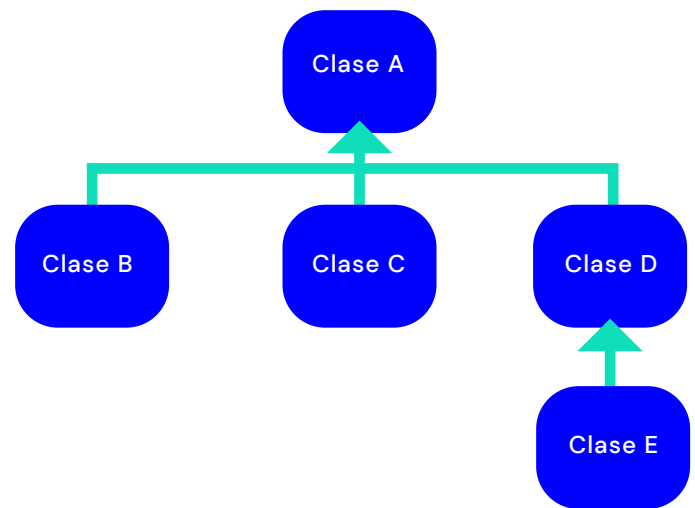


Imagen adaptada de UNAM (2022). *Programación con Java*. Recuperado de http://profesores.fi-b.unam.mx/carlos/java/java_basico3_4.html

- o A es la superclase de B, C y D.
- o D es la superclase de E.
- o B, C y D son subclases de A.
- o E es una subclase de D.



Referencias bibliográficas

- UNAM. (2022). *Programación con Java*. Recuperado de <https://bit.ly/3NLIfn3>