

# Herencia

## Uso de la Herencia

Java Básico, enero 2019

# Uso de la Herencia

Existen 4 pilares fundamentales de la programación orientada a objetos:

- Abstracción
- Encapsulamiento
- Polimorfismo
- **Herencia**



# Uso de la Herencia

## herencia

*nombre femenino*

1. Derecho a heredar que tiene una persona por ley o por testamento.  
"estas tierras me corresponden por herencia"
2. Conjunto de bienes, derechos y obligaciones que se heredan de una persona tras su muerte.  
"salió de sus apuros económicos gracias a la sustanciosa herencia que le dejó su difunta hermana"

Traducciones, origen de palabras y más definiciones



PROTECO

# Uso de la Herencia

- En programación orientada a objetos, la herencia es, un mecanismo utilizado para alcanzar algunos de los objetivos más preciados en el desarrollo de software como la reutilización.
- La herencia facilita la creación de objetos a partir de otros ya existentes e implica que una subclase obtiene todo el comportamiento (métodos) y eventualmente los atributos (variables) de su superclase.
- La herencia permite compartir automáticamente métodos y datos entre clases, subclases y objetos.





# Uso de la Herencia

- Herencia Simple : Una clase sólo puede heredar de una clase base y de ninguna otra.
- Herencia Múltiple: Una clase puede heredar las características de varias clases base, es decir, puede tener varios padres.
- - Java sólo admite herencia Simple



# Uso de la Herencia

## Clase Object

La clase Object es la clase raíz del árbol de la jerarquía de las clases en Java. Proporciona algunos métodos útiles que todos los objetos heredan, por ejemplo:

- Un método por el que un objeto se puede comparar con otro objeto (equals())
- Un método para convertir un objeto a una cadena (toString())
- Un método para devolver la clase de un objeto (getClass())



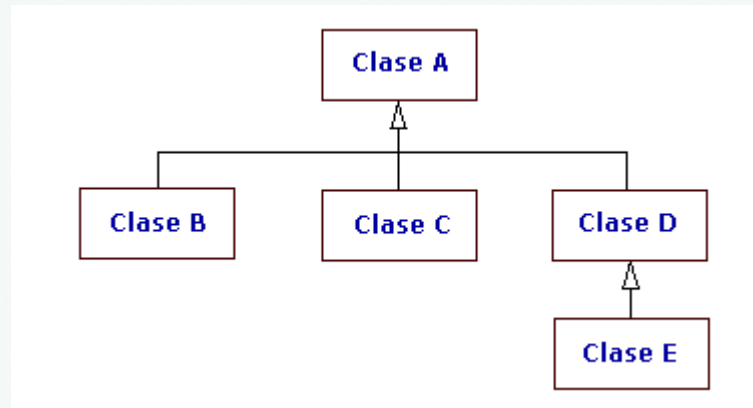
# Herencia

Superclase y  
subclase

Java Básico, enero 2019

# Superclase y subclase

- El concepto de herencia implica a una estructura jerárquica de clases:





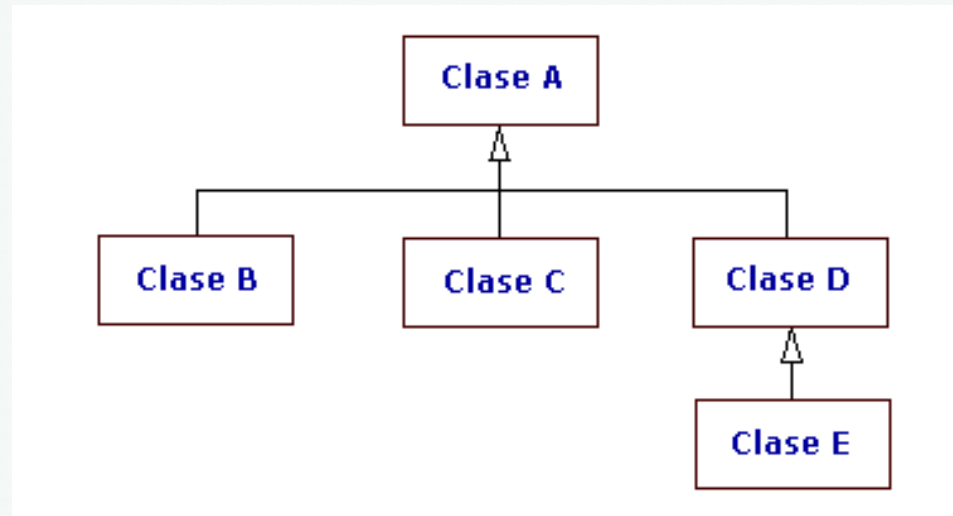
# Superclase y subclase

- - La clase padre de cualquier clase es conocida como su superclase.
- - La clase hija de una superclase es llamada una subclase.
- 
- Una superclase puede tener cualquier número de subclases.
- Una subclase puede tener sólo una superclase.
- 
- 



# Superclase y subclase

- **Ejercicio:** Establecer la relación existente entre las clases mostradas según la jerarquía indicada.



# Superclase y subclase

Respuesta:

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



PROTECO

# Superclase y subclase

Cuando una clase hereda de otra se usa la palabra reservada `extends`, al hacer esto se están "copiando" los métodos y las variables de la clase padre, la palabra `extends` se usa después de declarar el nombre de la clase, por ejemplo:

```
<tipo de clase> <nombre clase> extends <clase padre> {  
  
... resto de código ...  
  
}
```





# Superclase y subclase

Así como utilizamos la palabra reservada **this** para referirnos a elementos del mismo ámbito del objeto, existe algo que nos puede ayudar a acceder a los atributos y métodos de las clases padre o superclases, y es la palabra reservada **super**, con ella podemos invocar ya sea desde una instancia o un método de la subclase algo que venga de la clase desde la que hereda.



# Superclase y subclase

Uno de los usos más frecuentes que le damos a la palabra super, es poder invocar al constructor de la superclase, con ello podemos reducir la cantidad de código que debemos escribir en nuestras clases hijas

```
public Circulo(double radio, String color, boolean lleno) {  
    super(color, lleno);  
    this.radio = radio;  
}
```



# Sobrecarga de métodos

Una subclase hereda todos los métodos de su superclase que son accesibles a dicha subclase a menos que la subclase sobrescriba los métodos.

Una subclase sobrescribe un método de su superclase cuando define un método con las mismas características ( nombre, número y tipo de argumentos) que el método de la superclase.

Las subclases emplean la sobreescritura de métodos la mayoría de las veces para agregar o modificar la funcionalidad del método heredado de la clase padre.



# Referencias

- - Herencia, tomado de:  
[http://profesores.fi-b.unam.mx/carlos/java/java\\_basico3\\_4.html](http://profesores.fi-b.unam.mx/carlos/java/java_basico3_4.html)
- -Herencia, palabras reservadas, tomado de:  
<http://analista1.foroactivo.com/t21-herencia-palabras-reservadas-super-extends-implements>
- - Sobreescritura de métodos, tomado de:  
[http://profesores.fi-b.unam.mx/carlos/java/java\\_basico4\\_7.html](http://profesores.fi-b.unam.mx/carlos/java/java_basico4_7.html)
- 

