

## Herencia. Constructores

- Si la primera instrucción de un constructor de una subclase no es una invocación a otro constructor con **this** o **super**, Java añade de forma invisible e implícita una llamada **super()** con la que invoca al constructor por defecto de la superclase. Luego continúa con las instrucciones de inicio de las variables de la subclase y luego sigue con la ejecución normal. Si en la superclase no hay constructor por defecto (sólo hay explícitos) ocurrirá un error.
- Si se invoca a constructores de superclases mediante **super(...)** en la primera instrucción, entonces se llama al constructor seleccionado de la superclase, luego inicia las propiedades de la subclase y luego sigue con el resto de sentencias del constructor.
- Finalmente, si esa primera instrucción es una invocación a otro constructor de la clase con **this(...)**, entonces se llama al constructor seleccionado por medio de **this** y realiza sus instrucciones, y después continúa con las sentencias del constructor. La inicialización de variables la habrá realizado el constructor al que se llamó mediante **this(..)**.

El uso de **super** y **this** no puede ser simultáneo puesto que ambas tienen que ser la primera instrucción. Lo que significa que hay que elegir entre ambas.