

✓ 1) ¿Qué palabra se usa para generar la herencia entre clases? \* 10/10

- ☐ implements
- ☐ super
- ☐ this
- ☒ extends ✓

✓ 2) ¿Qué código de los siguientes tiene que ver con la herencia? \* 10/10

- ☒ public class Componente extends Producto ✓
- ☐ public class Componente inherit Producto
- ☐ public class Componente implements Producto
- ☐ public class Componente belong to Producto

✓ 3) ¿La superclase es la clase? \* 10/10

- ☒ Madre o Padre ✓
- ☐ Hija o Hijo
- ☐ Nieto o Nieto
- ☐ Es una interfaz

✓ 4) Cual de estos componentes de la superclase no hereda la subclase \* 10/10

- ☐ Atributos
- ☐ Métodos
- ☐ Getter/Setters
- ☒ Constructores ✓

✓ 5) La superclase debe ser abstracta \* 10/10

- ☐ Siempre
- ☐ Nunca
- ☒ Depende de la situación ✓
- ☐ Cuando implementa una interfaz

✓ 6) Podemos acceder a los atributos de una superclase desde una subclase gracias al modificador de acceso \* 10/10

- ☐ public
- ☐ private
- ☒ protected ✓
- ☐ final

✓ 7) ¿De cuántas clases se puede derivar/heredar en Java? \* 10/10

- ☐ Tres clases
- ☐ Dos clases
- ☒ Una clase ✓
- ☐ Cinco clases

✓ 8) Una clase que termina la cadena de una herencia se la declara como: \* 10/10

- ☒ final ✓
- ☐ abstract
- ☐ public
- ☐ Ninguna de las anteriores

✓ 9) ¿Qué palabra se usa para implementar una interfaz? \* 10/10

- ☒ implements ✓
- ☐ super
- ☐ this
- ☐ extends

✓ 10) ¿Qué código asociarías a una Interfaz en Java? \* 10/10

- ☐ public class Componente interface Product
- ☐ Componente cp = new Componente (interfaz)
- ☒ public class Componente implements Producto ✓
- ☐ Componente cp = new Componente.interfaz