

## Solución

1. Se debería sobrescribir el método pitar de la clase Auto en la clase moto, ya que como es un método aplicaría ligadura dinámica, yéndose así al tipo de la clase del objeto.  
@Override  
public void pitar(){System.out.println("Las motos no pitan")}  
O también se le podría asignar al valor de la variable pitar que se hereda de la clase Auto
2. Pasaría que si quisiera usar los atributos de la clase Moto tendría que volverlos a declarar puesto que tienen el modificador private, además de no contar con un getter o setter para poder interactuar con estos
3. Si, ya que lo hereda de la clase padre Auto, es compatible en el retorno y tiene la misma firma, además de que los dos son métodos de instancia y son públicos, así que no habría problema en sobrescribirlo
4. Se puede usar ese método puesto que la clase lo hereda de su clase padre, y al no tener este método sobrescrito lo que hace el this es verificar dentro de la clase que esta si esta ese método y sino va a buscar a la clase padre
5. Al ser público no habría problema en accesarlo directamente, además como es un atributo estático seria correcto llamarlo con el nombre de su clase
6. No es posible debido a que su modificador de acceso es de paquete, por lo que en realidad la clase Moto o el objeto moto de tipo Moto no conoce ese método puesto que no esta definido en su clase y en realidad no lo heredó
7. Ya que este método lo heredó de la clase Auto y puesto que, en este caso a diferencia del anterior, su modificador de acceso es publico
8. De hecho, no hay problema con usar ese método ya que al ser público lo hereda de la clase Auto
9. Imprime 30 ya que este método es llamado desde una instancia de la clase Moto entonces imprime con el this se hace referencia a la instancia que llamo al método y como Moto tiene su versión del atributo velocidad entonces imprime este
10. Se deberían agregar los getters para cada uno de estos atributos en su respectiva clase, ya que al ser privados no pueden ser accesados directamente desde otra clase