

Solución Ejercicio 1.

- A.** Si se pierde la referencia a un objeto de tipo Persona, si el objeto no está referenciado por nada más se ejecuta su `finalize()` para cuando el garbage collection vaya a eliminarlo.
- B.** Con: `auto.getDueno().getNombre()`
- C.** Haciendo uso de: `auto2.setDueno(personas[0])`, le pasamos el objeto de tipo Persona que será el dueño.
- D.** Mi propuesta es: `auto2.getMotor().getVelocidadMaxima()`
- E.** Si el garbage collector fuera eficiente, esta sería la salida del programa:
Matando a: Alejandro
Matando a: Jaime
Matando a: Santiago
Soy Santiago
- F.** Lo que ocurriría es que imprime: Soy Daniel, porque una línea de código antes el segundo índice de la lista se re-assigna al objeto Persona cuyo nombre es Daniel.
- G.** Definir el método `toString()` en la clase Vehículo, pidiendo que retorne la placa del vehículo y su dueño.
- H.** `auto.getDueno().setMejorAmigo(personas[2]);`

Solución Ejercicio 2.

- A.** La salida es la siguiente:
char: entra a int
short: entra a int
byte: entra a int
long: entra a double
integer: entra a int
double: entra a double

float: entra a double

Esto se produce porque en la gran mayoría de casos, el método no cuenta con una definición para el tipo de atributo que se pasa, así que se decide por la definición del método inmediatamente superior.

- B.** Si se activa la función que recibe un short: El byte y el short entran a esta definición y ya no a la de int. Por ser del mismo tipo y el tipo superior más cercano.
- Si se activa la función que recibe un float: el long y el float entran a esta definición. Por ser del mismo tipo y el tipo superior más cercano.
- Si se comenta la función que recibe un double y se activa la del float, el código lanza error, porque no hay definición para el dato de tipo double.
- Si se comentan todas las funciones y solo se deja la del double, entonces todas entran a la del double por ser la única que hay.