

- A. Al eliminar la referencia de un objeto por ejemplo de tipo Persona en Java, lo que sucederá es que el Garbage Collector-GC se encarga de destruir los objetos no referenciados, liberando la memoria reservada, pero antes de esto es invocado automáticamente el método finalize()
- B. Aplicando el método getDueno() de la clase Vehiculo, y luego el método getNombre() de la clase Persona
- C. Aplicando el método setDueno(), el cual recibe un objeto de clase Persona y le asigna a "dueno" ese objeto de tipo Persona
- D. Para obtener la velocidadMáxima, primero accedemos al motor con getMotor() y luego llamamos al método getVelocidadMaxima() del enum Motor.
- E. "Matando a: Santiago" (por la línea personas[3].finalize())

El resultado de System.out.println(personas[3]), que sería "Soy Santiago", ya que personas[3] aún tiene una referencia explícita en el código en ese punto. El método finalize() imprime su mensaje, pero el objeto no será recolectado mientras existan referencias accesibles como es el caso.

- F. Previamente el objeto personas[1] se le asigna la referencia de personas[2], que en este caso es "Daniel", entonces al imprimir "personas[1]" el resultado será "Soy Daniel"
- G. Sobrescribiendo el método toString de la clase Vehiculo

```
@Override
public String toString() {
    String infoDueno = (dueno != null) ? dueno.getNombre() : "Sin
dueño";
    return "Placa: " + placa + ", Dueño: " + infoDueno;
}
```

- H. Podemos usar los métodos getDueno() y setMejorAmigo() de las clases Vehiculo y Persona respectivamente:

```
if (auto.getDueno() != null){
    auto.getDueno().setMejorAmigo(personas[2]);
}
```