

Samuel Botero Rivera  
cc: 1000654007

A) Si se pierde la referencia a un objeto de tipo Persona, el objeto quedará inaccesible y será elegible para ser recolectado por el garbage collector. Cuando esto suceda, el método `finalize()` se ejecutará antes de que la memoria ocupada por ese objeto sea liberada.

B) usando el método `getDueno()` de la clase Vehiculo, luego el método `getNombre()` de la clase Persona.

```
System.out.println(auto.getDueno().getNombre());
```

C) Podemos usar el método `setDueno()` de la clase Vehiculo para asignar un dueño al vehículo `auto2`

```
auto2.setDueno(personas[0]);
```

D) primero accedemos al motor con `getMotor()` y luego llamamos al método `getVelocidadMaxima()` del Motor

```
int velocidadMax = auto2.getMotor().getVelocidadMaxima();  
System.out.println(velocidadMax);
```

E) 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.

F) El objeto referenciado por `personas[1]` apunta al mismo objeto que `personas[2]`. Por lo tanto, el resultado será:

"Soy Daniel".

G) Debemos sobrescribir el método `toString()` en la clase Vehiculo.

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

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

```
auto.getDueno().setMejorAmigo(personas[2]);
```