

Taller 4 Java Ejercicio 1

A. ¿Qué pasaría si se pierde la referencia de un objeto de tipo Persona?

Si se pierde la referencia a un objeto de tipo Persona, el objeto quedará pendiente para ser recolectado por el Garbage Collector de Java. Esto significa que, si no hay más referencias apuntando a ese objeto, el recolector de basura puede liberar la memoria ocupada por él.

B. ¿Cómo podríamos conocer el nombre del dueño de la variable auto de la línea 11 del método main?

Para conocer el nombre del dueño de auto, primero necesitamos acceder al objeto dueño del Vehículo y luego obtener el nombre de la persona asociada a ese vehículo. Esto se puede hacer de la siguiente manera:

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

C. ¿De qué manera podemos agregar un dueño al Vehículo de la línea 13 del método main?

El Vehículo de la línea 13 (auto2) se crea sin un dueño asignado, por lo que para asignar un dueño, podemos usar el método setDueño(). Por ejemplo:

```
auto2.setDueño(personas[1]);
```

D. Usando la variable auto2 de la línea 13 del método main, obtenga el valor del atributo velocidadMáxima del motor del vehículo. Adjunte su propuesta.

Para obtener la velocidadMaxima del motor de auto2, primero necesitamos acceder al motor del vehículo y luego obtener el valor del atributo velocidadMaxima. Esto se puede hacer así:

```
System.out.println(auto2.getMotor().getVelocidadMaxima());
```

Este código accede al motor de auto2 y luego llama al método getVelocidadMaxima() para obtener el valor de la velocidad máxima.

E. Suponga que, al momento de perder la referencia al objeto, se borra del sistema, es decir el garbage collector es muy eficiente, ¿Qué imprimiría al ejecutar el método main por consola?

Si el **Garbage Collector** es muy eficiente y elimina los objetos cuando se pierde la referencia, esto afectaría el comportamiento del programa. Sin embargo, en el código, el único método que se invoca cuando un objeto es recolectado por la basura es el método finalize() de la clase Persona, que imprime:

Matando a: <nombre>

Así que, si el Garbage Collector borra objetos como personas[0] o personas[4], se imprimirían mensajes de "matando a..." para esos objetos antes de ser eliminados.

F. ¿Qué ocurre al momento de ejecutar la siguiente línea después de la línea 16 `System.out.println(personas[1])` ? Explique

En este caso, después de la línea:

`personas[1] = personas[2];`

La referencia `personas[1]` se cambia para apuntar al mismo objeto que `personas[2]`. Es decir, ahora tanto `personas[1]` como `personas[2]` apuntan al objeto `Persona` correspondiente a "Daniel". Por lo tanto, al ejecutar:

`System.out.println(personas[1]);`

Imprimiría:

Soy Daniel

Esto ocurre porque `personas[1]` ahora es una referencia a "Daniel", después de la asignación en la línea 16.

G. ¿Qué modificación al código debo hacer para que al momento de ejecutar `System.out.println(auto2)`, me aparezca la placa del vehículo y el dueño del vehículo?

Para lograr esto, necesitamos sobrescribir el método `toString()` en la clase `Vehiculo` para que devuelva una representación en cadena que incluya tanto la placa del vehículo como el nombre del dueño.

H. Usando la variable `auto` de la línea 11 del método `main`, y usando el atributo `dueno`, asigne de `mejorAmigo` al tercer elemento del listado `personas`. Adjunte su propuesta.

Para asignar al tercer elemento del arreglo `personas` como el mejor amigo del dueño de `auto`, podemos hacer lo siguiente:

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

Este código establece a la persona en `personas[2]` (que es "Daniel") como el mejor amigo del dueño de `auto` (que es "Alexander").