

Taller 4 Java Ejercicio 2 Andrés Jacobo Leal Aguirre

Ejercicio 1

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

En algún momento pasaría el garbage collector a eliminarlo y en ese momento se imprimiría "Matando a" + el nombre del objeto

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

El dueño es la persona que esté en el índice 4 de la lista personas, y esta lista está conformada por personas con los nombres que están en la lista nombres. La persona en el 4to índice es "Alexander"

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

Pasándole como tercer parámetro un objeto de tipo Persona, o si el objeto ya está creado, usando el método setDueño() y pasándole como parámetro un objeto tipo Persona

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

```
auto2.getMotor().getVelocidadMaxima()
```

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?

Imprimiría:

Matando a: Alejandro

Matando a: Jaime

Matando a: Alexander

Matando a: Santiago

Soy Santiago

F. ¿Qué ocurre al momento de ejecutar la siguiente línea después de la línea 16

`System.out.println(personas[1])` ? Explique

Imprime "Soy Daniel" porque al índice 1 de la lista personas le estamos asignando ahora el valor de el índice 2 que es el objeto tipo Persona con nombre Daniel. Al llamarlo para imprimirlo, en el método toString() retorna, "Soy"+ el nombre de si mismo, que en este caso es Daniel

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?

Agregar en la clase Vehiculo el método toString() que retorne la placa del vehiculo y el dueño concatenados

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

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

Ejercicio 2

A. Según el siguiente código, indique qué se imprime por consola y explique el porqué de cada línea donde se imprime.

Se imprime:

char : Entra a int: 103

Se imprime esto por consola ya que se llama el método funcion(c) donde c es un dato de tipo char el cual toma el método donde la firma del método tiene como parámetro un dato de tipo int, por jerarquía el dato de tipo char tomaría el método del dato que le siga en la jerarquía. Se imprime lo que hay en el método y se imprime 103 ya que este es el número de el carácter 'g' en código ASCII.

short : Entra a int: 2

Se imprime esto por consola ya que se llama el método funcion(s) donde s es un dato de tipo short el cual toma el método donde la firma del método tiene como parámetro un dato de tipo int, por jerarquía el dato de tipo short tomaría el método del dato que le siga en la jerarquía.

byte : Entra a int: 1

Se imprime esto por consola ya que se llama el método funcion(b) donde b es un dato de tipo byte el toma el método donde la firma del método tiene como parámetro un dato de tipo int, por jerarquía el dato de tipo byte tomaría el método del dato que le siga en la jerarquía.

long : Entra a double: 9.99999999E8

Se imprime esto por consola ya que se llama el método funcion(l) donde l es un dato de tipo long el toma el método donde la firma del método tiene como parámetro un dato de tipo double, por jerarquía el dato de tipo long tomaría el método del dato que le siga en la jerarquía.

integer : Entra a int: 51232

Se imprime esto por consola ya que se llama el método funcion(i) donde i es un dato de tipo int el cual toma el método donde la firma del método tiene como parámetro un dato de tipo int.

double : Entra a double: 12.4

Se imprime esto por consola ya que se llama el método funcion(d) donde d es un dato de tipo double el cual toma el método donde la firma del método tiene como parámetro un dato de tipo double.

float : Entra a double: 5.650000095367432

Se imprime esto por consola ya que se llama el método funcion(f) donde f es un dato de tipo float el cual toma el método donde la firma del método tiene como parámetro un dato de tipo double, por jerarquía el dato de tipo float tomaría el método del dato que le siga en la jerarquía.

B. Realice los siguientes cambios, teniendo siempre como referencia el código inicial.

Explique cómo y por qué cambia lo que se imprime por pantalla.

- **Active la función que recibe un short.**

Cuando se desea imprimir el método funcion() que toma un tipo de dato de jerarquía menor igual que el short(en este caso byte y short) ya no se pasa por jerarquía al método con parámetro de tipo int que estaba preestablecido en el código original, sino que ahora pasa a tomar el método con el parámetro de tipo short, así cambiarían algunos outputs:

short : Entra a short: 2

byte : Entra a short: 1

- **Active la función que recibe un float.**

Cuando se desea imprimir el método funcion() que toma un tipo de dato de jerarquía menor igual que el float (en este caso long y float) ya no se pasa por jerarquía al método con parámetro de

tipo double que estaba preestablecido en el código original, sino que ahora pasa a tomar el método con el parámetro de tipo float, así cambiarían algunos outputs:

long : Entra a float: 1.0E9

float : Entra a float: 5.65

- **Comente la función que recibe un double y active la que recibe un float.**

Imprimiría error ya que cuando se quiera llamar el método funcion(d) con un parámetro d de tipo double por jerarquía no habría un método para este tipo de dato y habría una ambigüedad.

- **Comente todas las funciones, excepto la que recibe un double**

Se imprimiría todo correctamente de la siguiente manera:

char : Entra a double: 103.0

short : Entra a double: 2.0

byte : Entra a double: 1.0

long : Entra a double: 9.99999999E8

integer : Entra a double: 51232.0

double : Entra a double: 12.4

float : Entra a double: 5.650000095367432

Esto porque cuando se llama en estas líneas el método funcion() el parámetro es de tipo double y este tipo de dato es el más alto en la jerarquía por lo que todos los tipos de datos primitivos incluyendo el double pueden entrar a este método sin que haya ambigüedad.