



SOLUCIÓN TALLER # _ (PYTHON/JAVA)

ASIGNATURA

Programación Orientada a Objetos

PROFESOR

[Jaime Alberto Guzmán Luna](#)

ESTUDIANTE

Sebastián Martínez Sequeira

NÚMERO DE IDENTIFICACIÓN

1065593542

Universidad Nacional de Colombia

Sede Medellín

2024

RESPUESTA PREGUNTAS



¿Cuántas clases se están definiendo en este ejercicio?

Se están definiendo tres clases: La clase Animal, la clase Familia y la clase FamiliaAnimales.

¿En cuál clase se define el programa principal?

La clase principal del programa es FamiliaAnimales.

¿Cuántos objetos de la clase Animal se están creando en la clase que define el programa principal?

Se están creando dos nuevos objetos en la clase animal con respecto a la clase principal.

¿Cuáles objetos se están creando de la clase Animal en la clase que define el programa principal?

Los objetos que se han creado han sido Animal1 y Animal2.

¿Cuáles atributos tiene la clase Animal?

Los atributos de la clase Animal son los siguientes: nombre, genero, peso, pareja y el estático que es totalanimales.

¿Cuáles atributos de la clase Animal están haciendo referencia a tipos primitivos?

Existen dos atributos que son de tipo primitivo: “Double peso” y “int totalanimales”

¿Cuáles atributos de la clase Animal están haciendo referencia a objetos?

Existen tres tipos de atributos de tipo referencia en la clase animal y son: “string nombre”, “String género” y “Animal pareja”.

¿Con qué valor se inicializa el atributo pareja de la clase Animal?

El atributo de la clase Animal “pareja” se inicializa con el valor de null.

¿Cuál es el nombre que tienen los objetos animal1 y animal2 antes de la línea 7 en la clase FamiliaAnimales?

En ese instante estos se llamaban “new animal” porque es el proceso en el cual se crearon nuevos objetos para la clase “FamiliaAnimales”.

¿Cuál es el peso de animal1 en la clase FamiliaAnimal?

El peso del animal1 es 1.0 debido a que el peso asignado al hijo es de 1.0 también y se puede inferir que como animal1 es la madre del nuevo animal en este caso hijo entonces hijo se quedó con el mismo peso que su madre.

Dibuje el estado de memoria luego de establecer como pareja del caballo a la cebra y viceversa.

¿Cuál es el género del animal2 en la clase FamiliaAnimales?

El género del animal2 es masculino

Qué sucede si...

- ...se comenta la línea 15 de la clase FamiliaAnimales
- ...se comenta la línea 16 de la clase FamiliaAnimales
- ...se comentan las líneas 15 y 16 de la clase FamiliaAnimales



Si se comentarán estas líneas se estaría diciendo con quién va a quedar emparejado cada animal, en este caso sería Animal1 va a quedar emparejada con Animal2 y viceversa con Animal2.

¿En el contexto de cuál objeto se está ejecutando el método procrear cuando es invocado en la línea 18 de la clase FamiliaAnimales?

Según el contexto se está llamando al método “procrear” en el objeto Animal1, debido a que en la línea 18 se está llamando esa instancia y se está ejecutando en el objeto del Animal1. Y según la lógica se llama a que se ejecute en el objeto Animal1 porque como se ve esta es de género femenino y de ella sale el nuevo animal en este caso “hijo”.

¿Qué sucede si al atributo pareja de la clase Animal se le coloca el modificador final?

Que luego el valor de este atributo no va a cambiar en todo el programa, en pocas palabras el valor que se le asignó al atributo “pareja” va a ser inmutable.

¿Se puede eliminar el modificador static del método morir de la clase Animal sin que se afecte el funcionamiento del programa?

Si se afecta pues este está integrado para que al final del programa el Animal2 en este caso caballo siempre muera, si se llegara a quitar el modificador static, nos enviará un problema el cual dirá que no se puede crear una referencia estática para un método no estático y el programa no se lanzaría.

¿Qué sucede si modifica la línea 3 de la clase Animal como se indica a continuación?

Línea original: String

genero = "M";

Nueva línea: static final String

genero = "M";

Lo que ocurre aquí es que se está asignando un atributo estático al método de género, por lo que siempre el valor de este no va a cambiar al momento de que se inicialice el programa, esto puede generar una incompatibilidad en todas las instancias en la cuales el atributo género sea utilizado.

¿Por qué no es necesario asignarle el valor inicial al atributo totalAnimales de la clase Animal? Explique.

No es necesario asignarle un valor a totalAnimales debido a que este va a iniciar con un valor de cero y a medida que se ejecute el código este valor se va a ajustar según sea necesario durante la ejecución del programa.

¿Por qué razón, si se reemplaza la línea 28 de la clase FamiliaAnimales por la línea indicada a continuación, el resultado se mantiene igual?

Línea original:

System.out.println("Nuevo total de animales: " + Animal.totalAnimales);

Nueva línea:

System.out.println("Nuevo total de animales: " + animal1.totalAnimales);



El resultado se mantiene igual debido a que “totalAnimales” es un atributo estático de la clase Animal, por lo que pertenece en su totalidad a dicha clase, por eso una instancia de dicha clase en este caso animal1 no va a cambiar el resultado final de este atributo.

¿Por qué razón, no se afecta el resultado, si se reemplaza la línea 27 de la clase FamiliaAnimales por la indicada a continuación?

Línea original:

Animal.morir(animal2);

Nueva línea:

animal1.morir(animal2);

Esto no está afectando al resultado debido a que el método morir es estático y este está tomando como argumento a animal2, por lo que el animal que va a morir en estos casos va a ser animal2 y no animal1, a menos que se le cambie el argumento.

¿Cuántos métodos tiene la clase Animal?

La clase Animal tiene cuatro métodos que son los siguientes: setPareja, tenerHijo, Procrear y morir.

¿Cuál es el tipo de retorno de los métodos procrear() y tenerHijo() de la clase Animal?

El método procrear regresa un objeto del tipo Familia y el método tenerHijo regresa un objeto del tipo Animal.

¿Al método setPareja() de la clase Animal se le está pasando el parámetro por valor o por referencia?

El parámetro que se le da es por referencia, debido a que estamos dando como argumento un objeto al método setPareja y esto se hace es por referencia.

¿A quién está haciendo referencia la variable this de la línea 29 de la clase Animal cuando se ejecuta el programa principal? ¿Podría omitirse el uso de la variable this en este caso?

En la línea 29 cuando se usa la variable “this.” se está haciendo referencia a un objeto del tipo animal en este caso será a animal1 o animal2 y en el caso de que puede omitir “this.” por supuesto que sí, pero este se usa para que no ocurra alguna ambigüedad en el código porque puede haber otra variable que comparta el nombre y no queremos problemas.

¿Por qué no se afecta la ejecución del programa si se reemplaza la línea 38 de la clase Animal por la indicada a continuación?

Línea original:

System.out.println(nombre + " y " + pareja.nombre + " van a tener un hijo");

Nueva línea:

System.out.println(this.nombre + " y " + this.pareja.nombre + " van a tener un hijo");

En este caso no va a afectar en nada si usamos la variable “this.” debido a que ya estamos llamando a los objetos en este caso dos objetos del tipo Animal y para



agregar que no hace falta utilizarlo debido que no presentamos ambigüedades que requieran de su uso.

¿Podría eliminarse el modificador static del método morir() de la clase Animal sin que se afecte el programa?

No se puede, debido a que el método morir al ser un método estático este tiene que acceder a más métodos estáticos para el buen funcionamiento del programa, por ejemplo este tiene que acceder a miembros estadísticos de la clase Animal, por lo tanto es necesario y no se puede eliminar.

¿A quién hace referencia la variable this de las líneas 44 y 46 de la clase Animal cuando se ejecuta el programa principal? ¿Por qué es necesario?

Esas líneas de código en donde la variable “this” se está usando para hacer referencia al objeto actual del tipo animal y este es necesario porque se está creando una nueva instancia en el código y dice que “this” es el animal actual por lo que este será la mamá del nuevo animal y la pareja de éste será el papá del nuevo animal.

Modifique el método imprimir de la clase Familia para que sea un método de clase.

Ya se modificó y se cambió su instancia en la clase FamiliaAnimales, pero al convertir este en un método de clase, ha provocado que el género de hijo no se determine por completo e impide que se compile el código hasta el final.

¿Por qué razón es útil que el atributo totalAnimales sea un atributo de clase y no un atributo de ejemplar?

Es útil porque necesitamos que totalAnimales sea de clase ya que este va a mantener su valor hasta que en un momento determinado este tenga que ajustar su valor a medida que se compila el código, si fuera ejemplar entonces ya se tendría que asignar para que fuera llamado por otras instancias para que vaya obteniendo su valor en cada momentos que el programa lo solicite.

¿Se puede colocar el modificador static al método tenerHijo() de la clase Animal sin que se afecte el funcionamiento del programa? Explique.

Claro que se puede colocar sin que afecte el funcionamiento del programa, por la simple razón de que el método no tiene un valor en específico, lo que está haciendo este método es encargarse de darle sus atributos al nuevo objeto que se ha creado, como por ejemplo su peso, su nombre y su género, entonces es por eso que el modificador “static” no le hace efecto ni produce una ambigüedad en el programa.

¿Qué hace el método toString() de la línea 59 de la clase Animal?

Lo que hace este método es enviarme una cadena de texto en la que me especifica el nombre del nuevo objeto en este caso hablo de “hijo” y en la cuál esta línea me va a regresar cuál es su nombre y su peso.

