

1. Ejercicio de análisis

a) Explique por qué la clase Instrumento debe definirse como abstracta y qué la

la clase instrumento se usa como una base para crear clases con características comunes, pero como clase por si misma no tiene sentido, así que las clases hijas que heredan a partir de la clase instrumento, son quienes redefinen los metodos abstractos de tocar y afinar, manteniendo la clase no abstracta get tipo, y la diferencia de una clase normal es que no se pueden crear instancias de instrumento unicamente, sino que de sus clase hijas.

b) Elabore una clase denominada Piano heredada de Instrumento, añada un constructor y redefina para ella los métodos Tocar y Afinar.

```
public class Piano extends Instrumento {  
    public Piano(String tipo) {  
        super(tipo);  
    }  
  
    public void Tocar() {  
        System.out.println("Tocando Piano");  
    }  
  
    public void Afinar() {  
        System.out.println("Afinando Piano");  
    }  
}
```

c) ¿Señale cuál de las siguientes líneas no se pueden ejecutar y por qué?

la 3, ya que se intenta crear una inteancia de instrumento, la cual es una clase abstracta

d) ¿Qué imprime el siguiente programa?
error, ya que no compila la línea 3

2. Ejercicio de código en el repositorio

a. ¿Qué sucede si se define el método explotar() de la clase Estrella como se indica a continuación? Explique su respuesta.

esta mal implementada ya que es un metodo abstracto, el cual no deberia tener una implementacion dentro de el programa, pero como se muestra en la imagen, se usa un print, asi que no podria ser un metodo abstracto.

b. ¿Qué significa que los métodos tipoCuerpo2() y getID() de la clase ObjetoAstronomicoExtraSolar, no se definan como abstract? ¿Podría considerarse esta situación un error? Explique.

no, estan bien implementados, ya que aunque sea una clase abstracta, sus metodos no tienen porque ser abstractos,

en este caso esos 2 metodos se les da una implementacion en la definicion de la clase, y si fueran abstractos, no se les podria dar.

c. Si se define como abstracta la clase ObjetoAstronomicoExtraSolar, como se indica a continuación, ¿puede considerarse un error definir una clase abstracta sin métodos abstractos? Explique.

no necesariamente, aunque una clase abstracta la mayoría del tiempo cuenta con metodos abstractos, estos metodos no son obligatorios a la hora de crear una clase abstracta, en este caso usando unicamente metodos no abstractos, También funcionaria la implementacion

d. Explique por qué el arreglo oa (línea 19) hace referencia a una clase abstracta y sin embargo, en la línea 25 se invoca el método correspondiente a cada clase derivada

en la linea 19 se esta creando un array de 3 elementos, pertenecientes a la clase objetoastronomicoextrasolar, al llamar al metodo descripcion en la linea 25, al esta clase padre tener el metodo descripcion como un metodo abstracto, llama a la definicion de este metodo sobreescrito por las clases hijas

e. ¿Por qué la línea 29 imprime "Soy una Super Nova" sabiendo que el arreglo oa en esa posición fue inicializado con un objeto de tipo Galaxia?

ya que en la linea 28, justo antes de ejecutarse, se define que el apuntador del objeto 0, ahora sera el mismo apuntador del objeto 2, que es una supernova

f. ¿Por qué en la clase Estrella no se define el método descripcion() si la superclase lo está solicitando, ya que en este método descripción en abstracto?

las clase hijas tienen la posibilidad de definir los metodos abstractos, mas no es obligatorio, de esta forma las clases hijas de estrella, pueden tambien definir este metodo descripcion()

g. ¿Qué sucede si el método tipoCuerpo1() de la clase Galaxia se define como privado?
¿Por qué se genera error?

porque en la linea 8 donde se llama a tipocuerpo1 de el objeto g que es una galaxia, se generaria un error al no poder acceder a este metodo

h. ¿Por qué la clase Nova no define el método tipoCuerpo1()? ¿Se podría definir? Si lo define, ¿qué interpreta de esta situación?

no lo define ya que su clase padre, estrella ya lo define con una implementacion, asi que la clase nova, hereda este metodo, si se puede definir pero se sobrescribiria la definicion ya planteada

i. ¿Qué imprime la línea 9? ¿Por qué se puede llamar al método toString() si la clase Galaxia no lo define y su papá ObjetoAstronomicoExtraSolar tampoco?

el metodo toString es un metodo ya implementado en java, que no hay que definir, este imprime el nombre y id del objeto Galaxia@63961c42

j. ¿Por qué en la línea 11 se puede crear un puntero obN de tipo ObjetoAstronomicoExtraSolar si esta es una clase abstracta?

en esta linea lo que se dice es que todos los objetos nova, tambien pueden ser tratados como objetos ObjetoAstronomicoExtraSolar, esto es posible gracias al polimorfismo, ya que al la clase nova ser una clase hija, tambien puede ser tratada como su clase padre

k. ¿Las siguientes instrucciones (instrucciones en las líneas B y C) son válidas? Explique.

la linea b no, ya que se esta intentando crear una instancia de la clase abstracta, en la linea c lo que se dice es que se crea una variable de tipo objetoastronomicoextrasolar, y se le asigna el valor de nova, lo cual si se puede hacer ya que nova es una clase hija.

l. Explique por qué (ver código a continuación) la siguiente instrucción en la línea B es correcta y la instrucción en la línea C es incorrecta. Omitiendo la instrucción en la línea C, ¿qué se imprime por pantalla? Explique su respuesta.

aunque el objeto oa se le asigne el valor del objeto nova, para el compilador va a seguir siendo de objetoastronomicoextrasolar, asi que al este no contar con este metodo, se genera un error, en la ultima linea, al especificar el metodo de la clase nova, donde si esta definido, si se ejecuta e imprime Boom!

m. ¿Por qué la línea 15 imprime true? ¿Para cualquier objeto que se cree siempre imprimirá lo mismo? ¿Qué imprimen las siguientes líneas? ¿Por qué?

Esa linea imprime true ya que la funcion isinstance verifica si un objeto pertenece a cierta clase, y todos los objetos pertenecen a la clase object, y los siguientes imprimen false y false ya que el objeto obN es una Nova, asi que no pertenece a las clases supernova ni galaxia