

Problema 4. Cuestiones: objetos, referencias y herencia

Objetivos

- Afianzar los conceptos de objeto y referencia y la relación entre ellos.
- Afianzar los conceptos relacionados con la herencia.

Desarrollo

Responder a las siguientes cuestiones de teoría.

Cuestión 1

1. Encontrar las líneas incorrectas en el siguiente código
2. Suponiendo que las líneas incorrectas están comentadas, indicar en que momento(s) se convierte algún objeto en basura

```
public static void main(String[] args) {
    UnaClase obj1;
    UnaClase obj2=new UnaClase();
    UnaClase obj3=obj2;

    obj1.cambiaNúmero(3);
    obj3.cambiaNúmero(2);
    obj3=null;
    obj2.cambiaNúmero(4);
    obj2=obj3;
    obj3.cambiaTexto("hola");
}
```

Donde la clase UnaClase es la mostrada a continuación:

```
public class UnaClase {

    private static int contadorCambios=0;
    private String texto;
    private int número;

    public void cambiaNúmero(int n) {
        contadorCambios++; número = n;
    }

    public void cambiaTexto(String txt) {
        contadorCambios++; texto = txt;
    }

    public static int contadorCambios() {
        return contadorCambios;
    }

    @Override
    public String toString() {
        return "[texto:" + texto + ", num:" + número +
            ", contador:" + contadorCambios + "]";
    }
}
```

Solución Cuestión 1

```
public static void main() {  
    UnaClase obj1;  
    UnaClase obj2=new UnaClase();  
    UnaClase obj3=obj2;  
  
    obj1.cambiaNúmero(3);      <- error1  
    obj3.cambiaNúmero(2);  
    obj3=null;  
    obj2.cambiaNúmero(4);  
    obj2=obj3;                 <- basura  
    obj3.cambiaTexto("hola"); <- error 2  
}
```

- error 1: obj1 no ha sido inicializado
- basura: el objeto con número=2 deja de estar apuntado por alguna referencia, por lo que se convierte en basura
- error 2: obj3 es una referencia que apunta a null (no apunta a ningún objeto), por lo que es incorrecto tratar de ejecutar uno de sus métodos

Cuestión 2

1. Encontrar las líneas incorrectas en el siguiente código
2. Suponiendo que las líneas incorrectas están comentadas:
 - indicar cuál sería la salida del programa
 - indicar en que momento(s) se convierte algún objeto en basura

```
public static void main(String[] args) {
    System.out.println(UnaClase.contadorCambios());

    UnaClase[] a;
    UnaClase obj1=new UnaClase();
    obj1.cambiaNúmero(3);

    System.out.println("long. array:"+a.length);

    a = new UnaClase[2];
    System.out.println("long. array:"+a.length);
    a[0]=obj1;
    a[1]=new UnaClase();

    for(int i=0; i<a.length; i++) {
        System.out.println(a[i]);
    }

    a[0].cambiaTexto("hola");
    obj1.cambiaTexto("adiós");
    for(int i=0; i<a.length; i++) {
        System.out.println(a[i]);
    }

    obj1=null;
    a[1]=null;
    System.out.println("long. array:"+a.length);
    System.out.println(UnaClase.contadorCambios());
}
}
```

Solución Cuestión 2

La línea

```
System.out.println("long array:"+a.length);
```

es un error puesto que a es una referencia que todavía no apunta a ningún array

Salida por consola:

```
0
long array:2
[texto:null, num:3, contador:1]
[texto:null, num:0, contador:1]
[texto:adiós, num:3, contador:3]
[texto:null, num:0, contador:3]
long array:2
3
```

Cuestión 3

1. Indicar cual sería la salida del siguiente método (las clases Abuelo, Padre e Hijo se muestran a continuación)

```
public static void main(String[] args) {
    Hijo h = new Hijo();

    System.out.println("\nm1 retorna:" + h.m1());

    System.out.println("\nm2:");
    h.m2();

    System.out.println("\nm3:");
    h.m3();

    System.out.println("\nm4:");
    h.m4();
}

public class Abuelo {

    public int m1() {
        return 0;
    }

    public void m2() {
        System.out.println("Abuelo.m2");
    }

    public void m3() {
        System.out.println("Abuelo.m3");
    }

    public void m4() {
        System.out.println("Abuelo.m4");
    }
}

public class Padre extends Abuelo {

    public int m1() {
        return 1;
    }

    public void m2() {
        super.m2();
        System.out.println("Padre.m2");
    }

    public void m4(int i) {
        super.m4();
        System.out.println("Padre.m4");
    }
}
```

```
public class Hijo extends Padre {  
    public int m1() {  
        return super.m1();  
    }  
  
    public void m2() {  
        super.m2();  
        System.out.println("Hijo.m2");  
    }  
  
    public void m3() {  
        super.m3();  
        System.out.println("Hijo.m3");  
    }  
}
```

Solución Cuestión 3

Salida por consola:

m1 retorna:1

m2:
Abuelo.m2
Padre.m2
Hijo.m2

m3:
Abuelo.m3
Hijo.m3

m4:
Abuelo.m4