

# Programación en Lenguaje Java

## Problema 6.2. Cuestiones sobre referencias y objetos



**Michael González Harbour**

**Mario Aldea Rivas**

Departamento de Matemáticas,  
Estadística y Computación

Este tema se publica bajo Licencia:

[Creative Commons BY-NC-SA 4.0](https://creativecommons.org/licenses/by-nc-sa/4.0/)

## Problema 6.2. Cuestiones sobre referencias y objetos.

### Objetivos

- Comprender la diferencia entre referencias y objetos en Java.
- Saber identificar las instrucciones que causan la creación de objetos y cuales la generación de basura.

Este problema es un compendio de cuestiones (algunas aparecidas en exámenes escritos de la asignatura).

La resolución de las cuestiones debería realizarse sin compilar ni ejecutar el código.

Se recomienda ir dibujando un diagrama con referencias y objetos que reproduzca la situación de la memoria del computador a medida que se va ejecutando el programa.

### Cuestión 1.

Responder a las siguientes cuestiones:

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 class Cuestion1{
    public static void main(String[] args) {
        UnaClase obj1;
        UnaClase obj2=new UnaClase();
        UnaClase obj3=obj2;

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

El código de la clase UnaClase se muestra a continuación:

```
public class UnaClase {
    private static int contadorCambios=0;
    private String texto;
    private int numero;

    public void cambiaNumero(int n) {
        contadorCambios++;
    }
}
```

```

        numero = n;
    }

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

    public static int contadorCambios() {
        return contadorCambios;
    }

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

```

## Cuestión 2 (Examen septiembre 2012)

Responde a las siguientes cuestiones relativas al código mostrado a continuación:

- Indica la **salida por consola** que produciría la ejecución del programa.
- Indica los objetos que se crean y en qué línea se produce la **creación**, así como aquellas líneas en las que un objeto se convierte en **basura**.

```

public class Cosa {
    public int num = 0;

    public Cosa() {
        num = 1;
    }

    @Override
    public String toString() {
        return "(" + num + ")";
    }
}

```

- - - - -

```

public class CuestRefObjArray {

    public static void main(String[] args) {

        Cosa[] cosas = new Cosa[3];

        if (cosas[1] == null) {
            System.out.println("Es null");
        }
    }
}

```

```

    cosas[0] = new Cosa();
    cosas[1] = new Cosa();
    cosas[2] = cosas[1];
    cosas[1].num = 2;

    for(Cosa c: cosas) {
        System.out.println(c);
    }

    cosas[2].num = 3;

    for(Cosa c: cosas) {
        System.out.println(c);
    }

    cosas[1] = null;
    cosas[2] = null;
}
}

```

### Cuestión 3 (Examen septiembre 2013)

Responde a las siguientes cuestiones relativas al código mostrado a continuación:

- Indica la **salida por consola** que produciría la ejecución del programa.
- Indica los objetos que se crean y en qué línea se produce la **creación**, así como aquellas líneas en las que un objeto se convierte en **basura**.

```

public class CuestionRefsObjs {
    public static void main(String[] args) {
        Cosa c1;
        Cosa c2 = null;
        c1 = new Cosa("A");
        Cosa[] cosas = {c1, c2, new Cosa("B")};
        c2 = new Cosa("A");

        if (c1 == cosas[0]) {
            System.out.println("c1 == cosas[0] => TRUE");
        }
        if (c2 == cosas[0]) {
            System.out.println("c2 == cosas[0] => TRUE");
        }
        if (c2.equals(cosas[0])) {
            System.out.println("c2.equals(cosas[0]) => TRUE");
        }
        if (c2.equals(cosas[2])) {
            System.out.println("c2.equals(cosas[0]) => TRUE");
        }
    }
}

-----

public class Cosa {
    public String nombre;
}

```

```
public Cosa(String nombre) {
    this.nombre = nombre;
}

@Override
public boolean equals(Object obj) {
    Cosa c = (Cosa) obj;
    return nombre.equals(c.nombre);
}

@Override
public String toString() {
    return "(" + nombre + ")";
}
}
```

## Cuestión 4

Responder a las siguientes cuestiones:

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

(La clase UnaClase es la mostrada en la cuestión 1)

```
public class Cuestion4
{
    public static void main(String[] args) {
        System.out.println("Contador Cambios: " +
            UnaClase.contadorCambios());

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

        System.out.println("Longitud 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]);
        }
        System.out.println("Contador Cambios: " +
            UnaClase.contadorCambios());

        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("Contador Cambios: " +
            UnaClase.contadorCambios());
    }
}
```

### Cuestión 5 (Examen junio 2014)

Dadas las clases:

```
public class Dato {
    public int num;

    public Dato() {
    }

    public Dato(int i) {
        num = i;
    }
}

public class Cosa {
    public String str;
    public Dato dato;

    public Cosa(String str, Dato dato) {
        super();
        this.str = str;
        this.dato = dato;
    }

    @Override
    public String toString() {
        return "(" + str + ", " + dato.num + ")";
    }
}
```

Responde las cuestiones relativas a cada uno de los programas que aparecen a continuación:

#### 3.a)

```
public class CuestionA {
    public static void main(String[] args) {
        Dato d = new Dato();
        Cosa c1;
        Cosa c2 = new Cosa("C2", d);
        c1 = new Cosa("C1", d);
        System.out.printf("%s\n%s\n", c1, c2);
        c1.dato.num = 3;
        System.out.printf("%s\n%s\n", c1, c2);
    }
}
```

```
}
```

- Indica la **salida por consola** que produciría la ejecución del programa.
- Indica los objetos que se crean y en qué línea se produce la **creación**.

### 3.b)

```
public class CuestionB {
    public static void main(String[] args) {
        Dato d = new Dato(1);
        Cosa c1 = new Cosa("C1", d);
        Cosa c2 = new Cosa("C2", new Dato(1));
        metodo(c2, 3);
        System.out.println(c1 + " " + c2 + " d:" + d.num);
        metodo(c1, 4);
        System.out.println(c1 + " " + c2 + " d:" + d.num);
    }

    public static void metodo(Cosa c1, int i) {
        c1.dato = new Dato(i);
    }
}
```

- Indica la **salida por consola** que produciría la ejecución del programa.
- Indica las líneas en las que un objeto se convierte en **basura** y di cual es ese objeto.

## Cuestión 6 (Examen septiembre 2014)

Dada la clase:

```
public class Cosa {
    public Character letra;
    private static int numero;

    public Cosa(Character letra) {
        this.letra = letra;
    }

    public static void setNumero(int n) {
        numero = n;
    }

    @Override
    public String toString() {
        return "(" + letra + ", " + numero + ")";
    }
}
```

Responde las cuestiones relativas a cada uno de los programas que aparecen a continuación:

### 3.a)

```
public static void main(String[] args) {
    Cosa c = new Cosa('A');
    Cosa c1 = new Cosa('A');

    if (c == c1) {
        System.out.println("c == c1 es verdadero");
    }
}
```

```
    }

    if (c.equals(c1)) {
        System.out.println("c.equals(c1) es verdadero");
    }

    System.out.println("c:" + c);

    c.setNumero(2);
    Cosa c2 = c1;
    c2.letra = 'B';

    System.out.println("c:" + c);
    System.out.println("c1:" + c1);
    System.out.println("c2:" + c2);
}
```

- Indica la **salida por consola** que produciría la ejecución del programa.
- Hay una forma más correcta de escribir la línea “`c.setNumero(2);`”. ¿Cuál?

### 3.b)

```
public static void main(String[] args) {
    Cosa[] a;
    a = new Cosa[2];

    a[0] = new Cosa('A');

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

    a[1] = a[0];
    for(Cosa c: a) {
        System.out.println(c);
    }

    a[1] = null;
    a[0] = new Cosa('B');
}
```

- Indica la **salida por consola** que produciría la ejecución del programa.
- Indica los objetos que se crean y en qué línea se produce la **creación**.
- Indica las líneas en las que un objeto se convierte en **basura** y di cual es ese objeto.