

UNIDAD TEMÁTICA 1: Introducción a JAVA

PRACTICOS DOMICILIARIOS INDIVIDUALES 4

EJERCICIO #1 (EN PAPEL)

Dado el siguiente código fuente:

```
public class IdentifyMyParts {  
    public static int x = 7;  
    public int y = 3;  
}
```

- a) ¿Cuáles son las variables de clase?
- b) ¿Cuáles son las variables de instancia?

¿Cuál es la salida que produce el siguiente código?

```
IdentifyMyParts a = new IdentifyMyParts();  
IdentifyMyParts b = new IdentifyMyParts();  
a.y = 5;  
b.y = 6;  
a.x = 1;  
b.x = 2;  
System.out.println("a.y = " + a.y);  
System.out.println("b.y = " + b.y);  
System.out.println("a.x = " + a.x);  
System.out.println("b.x = " + b.x);  
System.out.println("IdentifyMyParts.x = " + IdentifyMyParts.x);
```

Salida:

```
a.y = 5  
b.y = 6  
a.x = 2  
b.x = 2  
IdentifyMyParts.x = 2
```

EJERCICIO #2

- 1) Indica qué es lo que está mal en el siguiente programa:

```
public class SomethingIsWrong {  
    public static void main(String[] args) {  
        Rectangle myRect;  
        myRect.width = 40;  
        myRect.height = 50;  
        System.out.println("myRect's area is " + myRect.area());  
    }  
}
```

Faltó "= new Rectangle();" para instanciarlo

- 2) Repara el error, ejecuta el programa y verifica que la salida es correcta.
(Resuelto en código en PD4-Ej2)

EJERCICIO #3

- 1) El siguiente código crea un array y una string. ¿Cuántas referencias a estos objetos existen luego de que el código se ha ejecutado? ¿Es alguno de los objetos candidato a ser eliminado por el garbage collector?

```
String[] students = new String[10];  
String studentName = "Peter Parker";  
students[0] = studentName;  
studentName = null;  
...
```

La referencia students queda apuntando al array, por ende, este no se va a eliminar. El string "Peter Parker" si se podría eliminar porque studentName ya no apunta a él.

- 2) Cómo hace un programa para destruir un objeto que ha creado?
3) Dada la siguiente clase, llamada "ContenedorDeNumeros", escribe un programa que cree una instancia de la clase, inicialice sus dos variables miembro y luego muestre el valor de cada una de ellas.

```
public class NumberHolder {  
    public int anInt;  
    public float aFloat;  
}
```

2)

Los objetos se destruyen solos cuando el programa necesita memoria y un espacio en la memoria carece de referencia