

Práctica Nº 5

Clases, instancias. Parámetros del `main()`

1. Analice las siguientes declaraciones y asignaciones y marque las que están correctamente definidas:

☐ `int _animal;`
☐ `String nom-per;`
☐ `Long nro!grande;`
☐ `char goto='a';`

☐ `Persona p=0;`
☐ `char marca='VMW'`
☐ `float n=12f;`
☐ `double pi=3,14;`

☐ `String nom_per;`
☐ `int v2=20;`
☐ `boolean for=(v2>10);`
☐ `boolean for1=(v2>10);`

2. Considere y analice la siguiente clase ClaseA:

```
1. public class ClaseA{
2.     public String nombre;
3.     public String apellido;
4.     public void setValores(String unValor, String nombre){
5.         String esteValor = "nada";
6.         esteValor = unValor;
7.         esteValor = nombre;
8.         esteValor = this.nombre;
9.         this.nombre = nombre;
10.        System.out.println("Contenido del atributo nombre: "+ this.nombre);
11.        System.out.println("Contenido del parámetro nombre: "+nombre);
12.    }
13. }
```

- a. ¿Son correctas las asignaciones realizadas en las líneas 6, 7, 8 y 9?
b. ¿Con qué valor quedan las variables `esteValor` y `nombre` después de la operación de asignación?
c. ¿Las impresiones en las líneas 10 y 11 muestran correctamente los valores del atributo y del parámetro? ¿Qué sucede?
3. Ejecute la siguiente clase:

```
package curso;

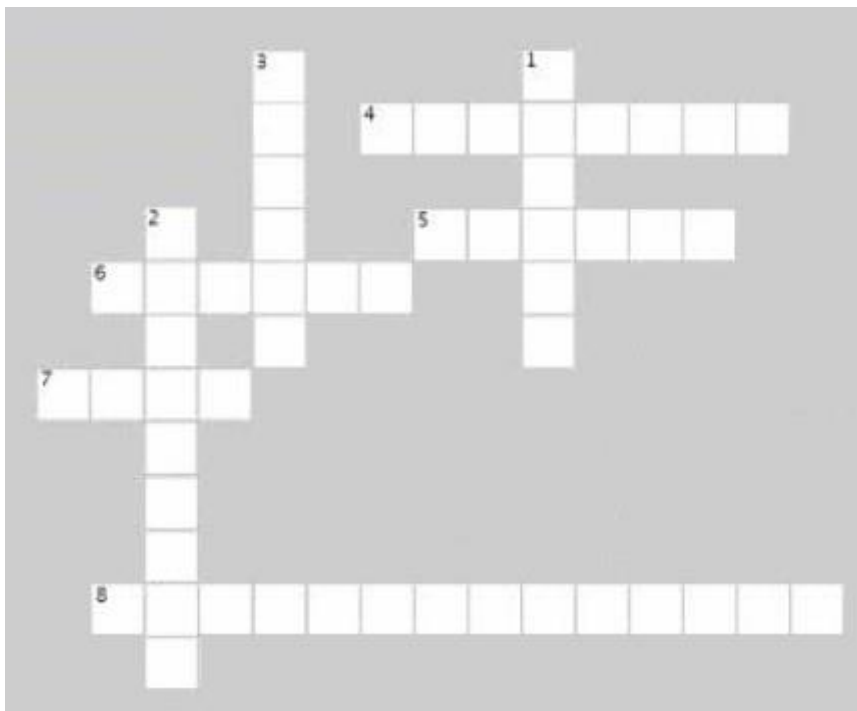
public class Matematica{

    public int promedio(int a, int b){
        return ((a+b)/2);
    }

    public static void main(String args[]){
        int a =10;
        int b =20;
        Matematica m = new Matematica();
        System.out.println("Valor a:" + a);
        System.out.println("Valor b:" + b);
        System.out.println("El promedio es: " + m.promedio(a, b));
    }
}
```

- Modifique el código de la clase para que los valores a y b sean recibidos por parámetro. Ejecute la clase utilizando diferentes valores para a y b.
- Agregue un método promedio que reciba 3 valores de tipo int y retorne el promedio de ellos.
- Agregue una variable de instancia de tipo float llamada PI y no la inicialice. Imprima su valor en el método main? ¿Qué valor tiene PI?
- Inicialice PI con el siguiente valor `private float PI = 3.1416F;`
- Agregue un método area, que reciba 1 parámetro de tipo entero con el radio del círculo y retorne el valor del área del mismo. Ejecute y pruebe el método.
- Agregue una **variable local** en el método area de tipo float llamada PI y no la inicialice. Use este valor para hacer el cálculo del área. ¿Qué imprime? ¿Por qué?

4. Complete el siguiente crucigrama



Horizontales

- Mecanismo usado en diseños orientados a objetos para expresar similitudes entre objetos.
- Nombre genérico dado a los métodos que recuperan los valores de los atributos de una clase.
- Comportamiento de un objeto.
- Palabra clave usada para referenciar a la instancia actual
- Conjunto de métodos a los que responden los objetos de una clase.

Verticales

- Nombre genérico dado a los métodos usados para setear valores en los atributos de una clase.
- Miembros de una clase que mantienen los valores de los objetos de la misma.
- Datos de un objeto.

5. Considere y analice la siguiente clase:

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

- ¿Cuáles son los atributos de clase?
- ¿Cuáles son los atributos de instancia?
- ¿Cuál es la salida del siguiente código?

```
IdentificandoMisPartes a = new IdentificandoMisPartes ();
IdentificandoMisPartes b = new IdentificandoMisPartes ();
a.setY(5);
b.setY(6);
a.setX(1);
b.setX(2);
System.out.println("a.getY() = " + a.getY());
System.out.println("b.getY() = " + b.getY());
System.out.println("a.x = " + a.x);
System.out.println("b.x = " + b.x);
System.out.println("IdentificandoMisPartes.x = " + IdentificandoMisPartes.x);
```

6. ¿Qué sucede cuando se intenta compilar y ejecutar el siguiente programa? Seleccione la respuesta correcta.

```
public class MyStatic {

    static int x = 6;

    MyStatic() {
        x ++ ;
    }

    public void metodo(){
        System.out.print("Valor de x " + x);
    }

    public static void main(String[] args){
        MyStatic mc1,mc2,mc3,mc4;
        MyStatic mc5 = new MyStatic();
        MyStatic mc6 = new MyStatic();
        MyStatic mc7 = new MyStatic();
        mc7. metodo ();
    }

}
```

- El programa no ejecuta.
- El programa escribe en la pantalla "Valor de x 6".
- El programa escribe en la pantalla "Valor de x 7".
- El programa escribe en la pantalla "Valor de x 8".
- El programa escribe en la pantalla "Valor de x 9".
- El programa no escribe nada en la pantalla.
- El programa no compila.

En caso de haber determinado que el programa no compila, ¿Por qué? ¿Qué modificaciones habría que hacer para que compile?

7. ¿Qué sucede cuando se intenta compilar y ejecutar el siguiente programa? Seleccione la respuesta correcta.

```
public class AtributoStatic{

    static int x;
    int y;

    AtributoStatic () {
        x += 2;
        y ++ ;
    }

    static int getCuadrado(){
        return x * x;
    }

    public int getY(){
        return y;
    }

    public static void main(String[] args){
        AtributoStatic sm1 = new AtributoStatic();
        AtributoStatic sm2 = new AtributoStatic();
        int z = sm1.getCuadrado();
        System.out.print("Cuadrado: " + z + " Valor de y: " + sm2.getY());
    }
}
```

- a. Seleccione la respuesta correcta:

- a. El programa no ejecuta.
- b. El programa no compila
- c. El programa escribe en la pantalla "Cuadrado: 16 Valor de y: 1".
- d. El programa escribe en la pantalla "Cuadrado: 4 Valor de y: 1".
- e. El programa escribe en la pantalla "Cuadrado: 4 Valor de y: 2".
- f. El programa escribe en la pantalla "Cuadrado: 2 Valor de y: 2".
- g. El programa escribe en la pantalla "Cuadrado: 16 Valor de y: 2".
- h. El programa no escribe nada en la pantalla.

En caso de haber determinado que el programa no compila, ¿Por qué? ¿Qué modificaciones habría que hacer para que compile?

- b. ¿Es correcta la siguiente invocación?

```
int z = sm1.getCuadrado();
```

8. Analice el código siguiente. ¿Compila? Si es así, de las opciones presentadas ¿Qué se imprimiría por pantalla?

```

public class StaticSuper{

    static{
        System.out.println("super static block");
    }

}

public class StaticTests extends StaticSuper{
    static int rand;

    static{
        rand = (int)(Math.random()*6);
        System.out.println("static block "+rand);
    }

    public StaticTests{
        System.out.println("constructor");
    }

    public static void main(String [] args){
        System.out.println("En el main");
        StaticTests st = new StaticTests();
    }
}

```

Posible salida 1:

```

static block 4
En el main
super static block
constructor

```

Posible salida 2:

```

super static block
static block 3
En el main
constructor

```