

## ACTIVIDADES TEMA 4. OPTIMIZACIÓN Y DOCUMENTACIÓN.

---

*Actividad 1. Refactorizar el siguiente código, indicando los pasos realizados para refactorizar:*

- Gastar el mínimo número de variables.
- Evitar repetir código.
- Intentar gastar la mejor estructura posible.
- Nombres de variables descriptivos.
- Hacer solo las operaciones necesarias.
- Usar tipos primitivos.

```
package catlaboral;
import java.util.Scanner;
public class CatLaboral {
    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);
        String c;
        double s;

        System.out.println("¿Introduce Categoría Laboral?");
        c = teclado.nextLine();

        System.out.println("¿Introduce Sueldo?");
        s = teclado.nextDouble();

        double r1;
        double r2;
        double r3;
        double r4;
        double r5;

        r1 = s * 0.10;
        r2 = s * 0.15;
        r3 = s * 0.20;
        r4 = s * 0.25;
        r5 = s * 0.30;

        if (c.equals("1")) {
            System.out.println("La retención es " + r1);
        } else if (c.equals("2")) {
            System.out.println("La retención es " + r2);
        } else if (c.equals("3")) {
            System.out.println("La retención es " + r3);
        } else if (c.equals("4")) {
            System.out.println("La retención es " + r4);
        } else if (c.equals("5")) {
            System.out.println("La retención es " + r5);
        }
    }
}
```

**Actividad 2.** Refactorizar todo lo posible el siguiente código para hacerlo más legible y eficiente. Con esto conseguiremos que sea más fácil de mantener en un futuro y que funcione más rápido.

```
import java.util.Scanner;

public class Actividad2 {

    public static void main(String[] args) {

        Scanner tec = new Scanner(System.in);
        int num1;
        int num2;
        String opcion;
        int resSuma;
        int resResta;

        System.out.print("Introduce número 1: ");
        num1 = tec.nextInt();
        tec.nextLine();
        System.out.print("Introduce número 2: ");
        num2 = tec.nextInt();
        tec.nextLine();

        resSuma = num1 + num2;
        resResta = num1 - num2;

        System.out.println("¿Qué desea realizar? (1 = Sumar, 2 = Restar)");
        opcion = tec.nextLine();

        if(opcion.equals("1")){
            System.out.println(resSuma);
        }else if(opcion.equals("2")){
            System.out.println(resResta);
        }
    }
}
```

*Actividad 3. Refactorizar todo lo posible el siguiente código para hacerlo más legible y eficiente. Con esto conseguiremos que sea más fácil de mantener en un futuro y que funcione más rápido.*

```
import java.util.Scanner;

public class Actividad3 {

    public static void main(String[] args) {

        Scanner tec = new Scanner(System.in);
        int[] numeros;

        numeros = new int[10];

        for(int i = 0; i < 10; i++){
            System.out.println("Introduce número " + (i+1));
            numeros[i] = tec.nextInt();
        }

        for(int i = 0; i < 10; i++){
            numeros[i] *= 2;
        }

        System.out.print("[");
        for(int i = 0; i < 10; i++){
            System.out.print(numeros[i] + " ");
        }
        System.out.println("]");

    }

}
```

**Actividad 4.** Refactorizar todo lo posible el siguiente código para hacerlo más legible y eficiente. Con esto conseguiremos que sea más fácil de mantener en un futuro y que funcione más rápido.

```
import java.util.Scanner;

public class Actividad4 {

    public static void main(String[] args) {

        Scanner tec = new Scanner(System.in);
        int a;

        System.out.println("Introduce el día de la semana en formato numérico (1-7)");
        a = tec.nextInt();

        if(a == 1){
            System.out.println("Día impar");
        }else if(a == 3){
            System.out.println("Día impar");
        }else if(a == 5){
            System.out.println("Día impar");
        }else if(a == 7){
            System.out.println("Día impar");
        }else if(a == 2){
            System.out.println("Día par");
        }else if(a == 4){
            System.out.println("Día par");
        }else if(a == 6){
            System.out.println("Día par");
        }else{
            System.out.println("Día incorrecto");
        }

    }

}
```

*Actividad 5. Refactorizar todo lo posible el siguiente código para hacerlo más legible y eficiente. Con esto conseguiremos que sea más fácil de mantener en un futuro y que funcione más rápido.*

```
import java.util.Scanner;

public class Actividad5 {

    public static void main(String[] args) {

        Scanner tec = new Scanner(System.in);
        String a;
        int b;
        double c;

        System.out.println("Introduce tu nombre:");
        a = tec.nextLine();

        System.out.println("Introduce tu edad:");
        b = tec.nextInt();

        System.out.println("Introduce tu altura:");
        c = tec.nextDouble();

        System.out.println("Tu nombre es " + a);
        System.out.println("Tu edad es " + b);
        System.out.println("Tu altura es " + c);

    }
}
```

**Actividad 6. Refactorizar todo lo posible el siguiente código para hacerlo más legible y eficiente. Consiguiendo que sea más fácil de mantener y que funcione más rápido.**

```
import java.util.Scanner;
public class Actividad6 {
    public static void main(String[] args) {
        Scanner tec = new Scanner(System.in);
        int num1;
        int num2;
        int suma;
        int resta;
        int multiplicacion;
        int division;
        int opcion;
        do{
            System.out.println("*****Menú principal*****");
            System.out.println("1. Sumar");
            System.out.println("2. Restar");
            System.out.println("3. Multiplicar");
            System.out.println("4. Dividir");
            System.out.println("5. Salir");
            System.out.print("Seleccione una opción: ");
            opcion = tec.nextInt();

            if(opcion == 1){
                System.out.print("Introduce número 1: ");
                num1 = tec.nextInt();
                System.out.print("Introduce número 2: ");
                num2 = tec.nextInt();
                suma = num1 + num2;
                System.out.println("El resultado es " + suma);
            }else if(opcion == 2){
                System.out.print("Introduce número 1: ");
                num1 = tec.nextInt();
                System.out.print("Introduce número 2: ");
                num2 = tec.nextInt();
                resta = num1 - num2;
                System.out.println("El resultado es " + resta);
            }else if(opcion == 3){
                System.out.print("Introduce número 1: ");
                num1 = tec.nextInt();
                System.out.print("Introduce número 2: ");
                num2 = tec.nextInt();
                multiplicacion = num1 * num2;
                System.out.println("El resultado es " + multiplicacion);
            }else if(opcion == 4){
                System.out.print("Introduce número 1: ");
                num1 = tec.nextInt();
                System.out.print("Introduce número 2: ");
                num2 = tec.nextInt();
                division = num1 / num2;
                System.out.println("El resultado es " + division);
            }else if(opcion == 5){
            }
        }while(opcion != 5);
    }
}
```

**Actividad 7.-** Crear la clase calculadora con los métodos suma, resta, multiplicacion, division, potencia, factorial y sumatorio que reciben uno o dos parámetros de tipo double, dependiendo del método y devuelven el resultado de tipo double. El método división estará deprecated (obsoleto) existiendo una nueva versión del mismo llamado dividir que controla con Excepciones la división por 0. Crear los comentarios necesarios para documentar la clase y los métodos y generar con Javadoc la documentación navegable HTML.

**Actividad 8.- Realizar las siguientes acciones de control de versiones sobre un directorio Git.**

**1.- Abrir cmd y en el disco c: crear una carpeta con nuestro nombre. Identificarse mediante nombre:**

**2.- Identificarse mediante email:**

**3.- Consultar el nombre de Identificación:**

**4.- Consultar el email de Identificación:**

**5.- Configurar colores para la consola de salida:**

**6.- Listar la Configuración:**

**7.- Crear directorio de trabajo git llamado ActividadGit:**

**8.- Entrar en la carpeta ActividadGit:**

**9.- Listar los ficheros del directorio ActividadGit:**

**10.- Mostrar el estado del proyecto:**

**11.- Crear un fichero llamado alumnos.txt con el nombre de 3 alumnos:**

**12.- Mostrar el estado del proyecto ActividadGit:**

**13.- Añadir los ficheros del Directorio de trabajo al Área Preparada:**

**14.- Mostrar el estado del proyecto ActividadGit:**

**15.- Crear una versión del proyecto ActividadGit en el repositorio llamado "Listado Alumnos":**

**16.- Mostrar el estado del proyecto ActividadGit:**

**17.- Editar el fichero alumnos.txt quitando el nombre de alumno 3 y añadiendo 2 alumnos más:**

**18.- Mostrar las diferencias entre el directorio de trabajo y el área preparada:**

**19.- Mostrar las diferencias entre el directorio de trabajo y el último commit:**

**20.- Mostrar las diferencias entre el área preparada y el último commit:**

**21.- Mostrar los commits realizados:**

**22.- Añadir el fichero alumnos.txt al área preparada:**

**23.- Mostrar las diferencias entre el área preparada y el último commit:**

**24.- Crear una versión del proyecto en el repositorio llamado "Listado de alumnos actualizado":**

**25.- Mostrar los commits realizados con un commit por línea:**

**26.- Mostrar los commits realizados en modo gráfico:**

**27.- Cambiar el comentario identificador del último commit por "Listado de alumnos Final"**

**28.- Crear el fichero profesores.txt con las siguientes líneas Marta, Nacho, Noelia y Rafael:**

**29.- Mostrar los ficheros de la carpeta:**

**30.- Añadir el fichero profesores.txt del directorio de trabajo al área preparada:**

**31.- Mostrar el estado del proyecto:**



**32.- Eliminar el fichero profesores.txt del área preparada:**

**33.- Mostrar el estado del proyecto:**

**34.- Añadir el fichero profesores.txt del directorio de trabajo al área preparada:**

**35.- Crear una versión del proyecto en el repositorio llamado "Profesores al repositorio":**

**36.- Mostrar los commits realizados:**

**37.- Modificar el fichero profesores.txt quitando los nombres Nacho y Noelia:**

**38.- Volver el fichero profesores.txt a la versión del último commit:**

**39.- Mostrar el fichero profesores.txt y comprobar que vuelven a estar los profesores Nacho y Noelia:**

**40.- Muestra Información y diferencias del directorio de trabajo respecto al commit llamado "Listado Alumnos":**

**41.- Muestra Información y diferencias del directorio de trabajo respecto al último commit:**

**42.- Crear una rama llamada Conselleria:**

**43.- Mostrar ramas. Se muestra \* en la rama activa:**

**44.- Cambiar a la rama Conselleria:**

**45.- Crear el fichero notas.txt con las líneas 1ª Evaluación, 2ª Evaluación, 3ª Evaluación y Evaluación Final:**

**46.- Añadir los ficheros del Directorio de trabajo al Área Preparada:**

**47.- Crear una versión en el repositorio llamada "Introducir Notas":**

**48.- Muestra los ficheros de la carpeta:**

**49.- Cambiar a la rama master:**

**50.- Mostrar los ficheros de la carpeta. Comprobar que no aparece el fichero notas.txt:**

**51.- Mostrar las ramas:**

**52.- Unir las rama Conselleria con master:**

**53.- Mostrar las ramas:**

**54.- Mostrar los ficheros de la carpeta. Comprobar que existe el fichero notas.txt al haber fusionado la rama master con Conselleria:**

**55.- Eliminar la rama Conselleria:**

**56.- Mostrar las ramas. Solo aparece la rama master:**

**57.- Creamos en la cuenta gratuita de GitHub en <https://github.com/> el repositorio remoto AulaCampus:**

**58.- Crear enlace llamado accesoAula al repositorio remoto <https://github.com/xxxxxxxx/AulaCampus>, siendo xxxxxxxx tu usuario:**

**59.- Mostrar los enlaces a repositorios remotos:**

**60.- Mostrar los enlaces a repositorios remotos y sus url's:**

**61.- Subir los ficheros de la rama local master al repositorio remoto con enlace AulaCampus:**

**62.- Comprobar que el repositorio remoto contiene los ficheros del directorio de trabajo local:**

**63.- Eliminar el fichero profesores.txt**

**64.- Eliminar el fichero notas.txt**

**65.- Mostrar los ficheros de la carpeta:**

**66.- Mostrar los enlaces a repositorios remotos:**

**67.- Mostrar el estado del proyecto ActividadGit:**

**68.- Añadir los cambios del directorio de trabajo al área preparada:**

**69.- Crear una versión llamada "Eliminamos profesores y notas" en el repositorio local:**

**70.- Mostrar los ficheros de la carpeta:**

**71.- Volver el proyecto a la versión del repositorio remoto, con lo que se recuperan los ficheros profesores.txt y notas.txt:**

**72.- Mostrar los ficheros de la carpeta:**

**73.- Mostrar el estado del proyecto:**