



Universidad Autónoma de Baja California

FACULTAD DE CIENCIAS QUÍMICAS E INGENIERÍA



Reporte de actividad

Materia:

Sistemas de Control

Tema:

Práctica No. 2: Inicialización de objetos precisando los atributos e invocación de métodos.

Alumno:

Joshua Osorio Osor...

Matrícula:

1293271

Docente:

Itzel Barriba Cazares

Fecha:

23/02/2025

ÍNDICE

Objetivo.....	2
Introducción.....	2
Lista de materiales.....	2
Desarrollo.....	2
Conclusiones.....	2
Bibliografía.....	3

Objetivo

El alumno diseñará clases para modelar el estado y comportamiento de entidades del mundo real, a través de atributos y métodos, respectivamente, respetando los principios de la orientación objetos y las capacidades del lenguaje de programación con una actitud crítica, propositiva y creativa.

Introducción

Para realizar esta práctica debes haber estudiado lo visto en clases anteriores, solo podrás consultar apuntes vistos en clases, no puedes hacer uso de internet.

Lista de materiales

- Computadora Personal (PC)
- Ambiente de desarrollo o IDE (VSCODE)
- JDK Java Development Kit

Desarrollo

Programa 1:

Haz el modelado UML de la siguiente clase que represente un libro en una

UABC_FCQI_Cuarto_Programación_Orientada_A_Objeto.

Biblioteca en donde contemple las siguientes características:

La clase Libro debe contener los siguientes atributos:

- Título del libro
- Año de publicación
- Editorial
- Géneros
- Número de páginas
- Autores

El libro debe tener un método por cada atributo para obtener los valores (getter) y un método por cada atributo para editar los valores (setter). Crear una clase principal donde el usuario pueda ingresar cada uno de los atributos mediante selección de opciones de un menú (opciones de captura para cada uno de los atributos) agregando una opción de imprimir los datos del libro. El usuario podrá repetir el proceso de modificar y mostrar datos hasta que seleccione salir del programa.

Menú de opciones:

- Modificar título del libro:
- Modificar año del libro de publicación
- Modificar Editorial
- Modificar Géneros
- Modificar Número de páginas
- Modificar Autores
- Mostrar datos completos del libro (deberá ser un método que junte todos los atributos).

El programa debe mostrar el dato anterior del libro y el dato modificado, cuando el usuario modifique los atributos del libro.

Crear una clase principal donde el usuario pueda ingresar cada uno de los atributos mediante selección de opciones de un menú (opciones de captura para cada uno de los atributos) agregando una opción de imprimir los datos del libro. Los datos deben ser capturados por el usuario. En el main solo debe hacer las capturas y puras llamadas o invocaciones a métodos.

Programa 2:

Cree una clase Empleado, en donde se contemplen los siguientes atributos:

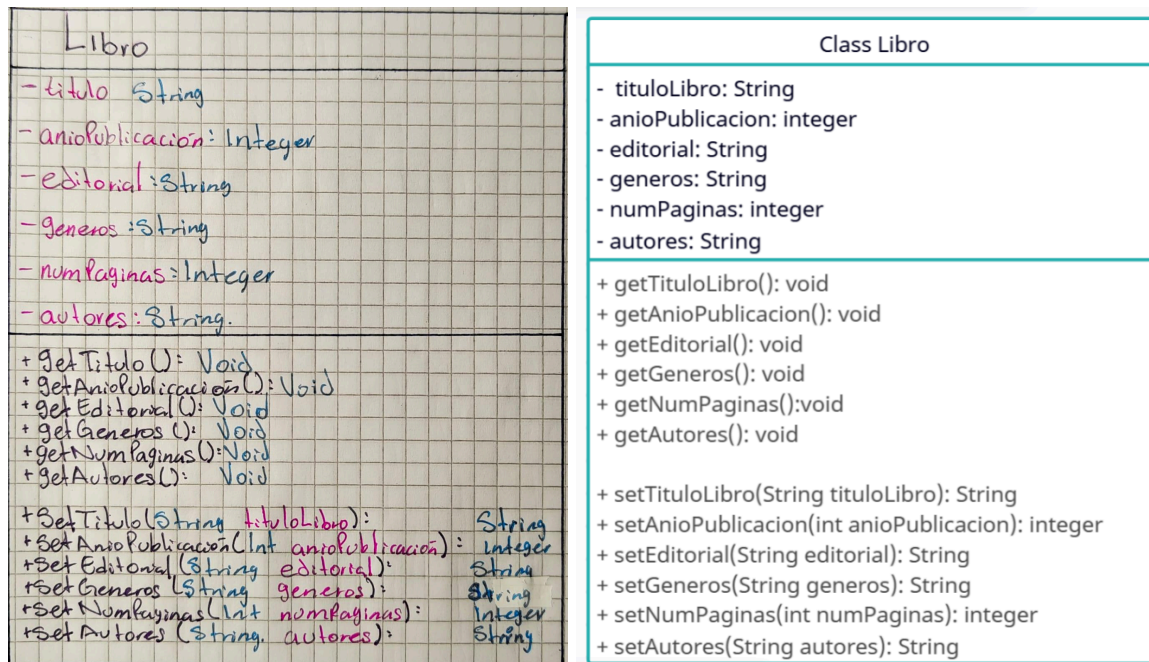
Nombre completo, fecha de nacimiento, salario semanal. Deberá pedir al usuario separados Nombres por separado (nombre1 y nombre 2), apellido paterno, apellido materno, el cual deberá concatenar y almacenarlos en el atributo nombre completo, y los datos de nacimiento (día, mes y año), considere el caso en el que un empleado tiene solo un nombre y no dos.

Crear una clase principal donde el usuario pueda ingresar cada uno de los atributos mediante selección de opciones de un menú (opciones de captura para cada uno de los atributos) agregando una opción de imprimir los datos del empleado y el salario mensual (crear un método para el cálculo). En el main solo debe hacer las capturas de datos y puras llamadas o invocaciones a métodos.

Análisis de Resultados

Programa 1:

Diagrama UML:



Libro.java

Java

```
public class Libro{
    // Atributos privados de la clase.
    private String tituloLibro;
    private int anioPublicacion;
    private String editorial;
    private String generos;
    private int numPaginas;
    private String autores;

    // Metodos get para traer valores de la clase.
    public String getTituloLibro(){return tituloLibro;}
    public int getAnioPublicacion(){return anioPublicacion;}
    public String getEditorial(){return editorial;}
    public String getGeneros(){return generos;}
    public int getNumPaginas(){return numPaginas;}
    public String getAutores(){return autores;}

    // Metodos set de la case, se utilizan para declarar los atributos.
    public void setTituloLibro(String tituloLibro){this.tituloLibro =
tituloLibro;}
    public void setAnioPublicacion(int anioPublicacion){this.anioPublicacion
= anioPublicacion;}
    public void setEditorial(String editorial){this.editorial = editorial;}
    public void setGeneros(String generos){this.generos = generos;}
    public void setNumPaginas(int numPaginas){this.numPaginas = numPaginas;}
    public void setAutores(String autores){this.autores = autores;}
}
```

Principal.java

Java

```
/*-----
Practica: 3 Inicialización de objetos precisando los atributos e invocación de
métodos.
Nombre: Joshua Osorio
Materia: Programación Orientada a Objetos
Fecha: Febrero/25/2025
```

```

-----*/

import java.util.Scanner;

public class Principal{
    public static void main(String [] args){
        Scanner input = new Scanner(System.in);

        // Creacion de un objeto de la clase Libro.
        Libro generico = new Libro();

        // Declaracion de las variables utilizadas en el programa y
        asignarles valores null a los String para que no falle el metodo de la clase
        Libro.
        String titulo = null, editorial = null, genero = null, autores =
        null;
        int opcion, anioPublicacion, numpaginas;

        System.out.printf("Sistema para capturar informacion de libro\n");

        // Inicio el ciclo do while para estar mostrando siempre las
        opciones.
        do{
            // Se despliega información para mostrar las opciones.
            System.out.printf("\nMenú\n");
            System.out.printf("\t1 - Modificar título del libro\n");
            System.out.printf("\t2 - Modificar año del libro de
publicación\n");
            System.out.printf("\t3 - Modificar Editorial\n");
            System.out.printf("\t4 - Modificar Géneros\n");
            System.out.printf("\t5 - Modificar Número de páginas\n");
            System.out.printf("\t6 - Modificar Autores\n");
            System.out.printf("\t7 - Mostrar información\n");
            System.out.printf("\t0 - Salir\n");
            // Se captura la opcion que desea.
            System.out.printf("\tOpción:\t");
            opcion = input.nextInt();

            // Limpia el bufer despues de ingresar un numero y asi
            capturar correctamente cadenas de texto.
            input.nextLine();
            switch (opcion){
                case 1:

```

```

        System.out.printf("\tModificar título del libro\t");
        System.out.printf("\n\tTitulo anterior:\t%s\n\t",
generico.getTituloLibro());
        titulo = input.nextLine();
        generico.setTituloLibro(titulo);
        System.out.printf("\tTitulo nuevo:\t%s",
generico.getTituloLibro());
        break;
    case 2:
        System.out.printf("\tModificar año del libro de
publicación:\t");
        System.out.printf("\n\tAño
anterior:\t%d\n\t", generico.getAnioPublicacion());
        anioPublicacion = input.nextInt();
        generico.setAnioPublicacion(anioPublicacion);
        System.out.printf("\tAño
nuevo:\t%d\n", generico.getAnioPublicacion());
        break;
    case 3:
        System.out.printf("\tModificar Editorial\t");
        System.out.printf("\n\tEditorial anterior:\t%s\n\t",
generico.getEditorial());
        editorial = input.nextLine();
        generico.setEditorial(editorial);
        System.out.printf("\tEditorial nueva:\t%s",
generico.getEditorial());
        break;
    case 4:
        System.out.printf("\tModificar Géneros\t");
        System.out.printf("\n\tGeneros anterior:\t%s\n\t",
generico.getGeneros());
        genero = input.nextLine();
        generico.setGeneros(genero);
        System.out.printf("\tTitulo nuevo:\t%s",
generico.getGeneros());
        break;
    case 5:
        System.out.printf("\tModificar Número de
páginas:\t");
        System.out.printf("\n\tPaginas
anterior:\t%d\n\t", generico.getNumPaginas());
        numpaginas = input.nextInt();
        generico.setNumPaginas(numpaginas);

```

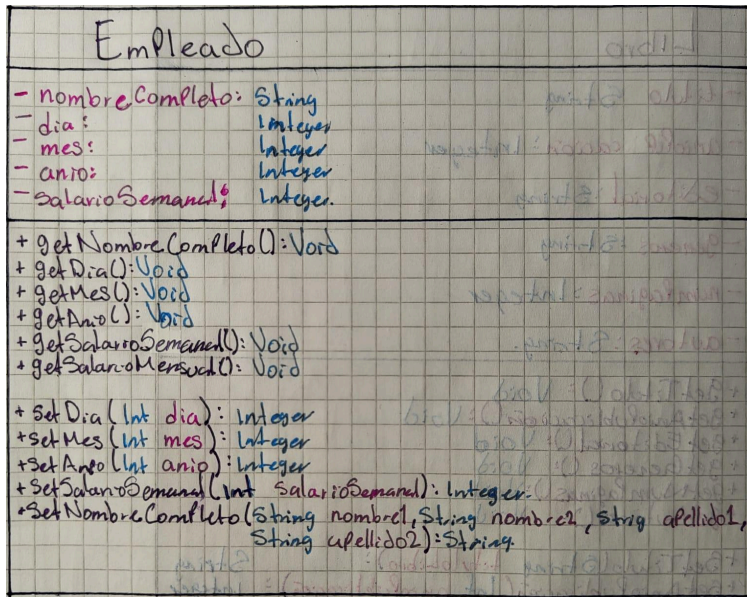
```

        System.out.printf("\tpaginas
nuevas:\t%d\n", generico.getNumPaginas());
        break;
    case 6:
        System.out.printf("\tModificar Autores\t");
        System.out.printf("\n\tAutores anterior:\t%s\n\t",
generico.getAutores());
        autores = input.nextLine();
        generico.setAutores(autores);
        System.out.printf("\tAutores nuevo:\t%s",
generico.getAutores());
        break;
    case 7:
        System.out.printf("\tMostrar Informacion\t");
        System.out.printf("\tTitulo anterior:\t%s",
generico.getTituloLibro());
        System.out.printf("\tAño
anterior:\t%d\n", generico.getAnioPublicacion());
        System.out.printf("\tEditorial anterior:\t%s",
generico.getEditorial());
        System.out.printf("\tGeneros anterior:\t%s",
generico.getGeneros());
        System.out.printf("\tPaginas
anterior:\t%d\n", generico.getNumPaginas());
        System.out.printf("\tAutores anterior:\t%s",
generico.getAutores());
        break;
    case 0:
        System.out.printf("\tSaliendo\t");
        break;
    default:
        System.out.printf("\tOpción no válida, intente
nuevamente.:\t");
        break;
    }
}while(opcion != 0);
}
}

```


Programa 2:

Diagrama UML:



Empleado.java

Java

```

public class Empleado {
    private String nombreCompleto;
    private int dia;
    private int mes;
    private int anio;
    private int salarioSemanal;

    // Métodos getter para obtener los atributos del empleado
    public String getNombreCompleto() { return nombreCompleto; }
    public int getDia() { return dia; }
    public int getMes() { return mes; }
    public int getAnio() { return anio; }
    public int getSalarioSemanal() { return salarioSemanal; }

    // Método para calcular el salario mensual basado en el salario semanal
    public int getSalarioMensual() { return salarioSemanal * 4; }

    // Métodos setter para modificar los atributos del empleado
    public void setDia(int dia) { this.dia = dia; }
    public void setMes(int mes) { this.mes = mes; }
  
```

```

    public void setAnio(int anio) { this.anio = anio; }
    public void setSalarioSemanal(int salarioSemanal) { this.salarioSemanal =
salarioSemanal; }

    public void setNombreCompleto(String nombre1, String nombre2, String
apellido1, String apellido2) {
        /*
        - Método para establecer el nombre completo del empleado.
        - Se concatena dinámicamente según los valores proporcionados.
        */
        if (nombre1 != null) { this.nombreCompleto = nombre1 + " "; }
        if (nombre2 != null) { this.nombreCompleto += nombre2 + " "; }
        if (apellido1 != null) { this.nombreCompleto += apellido1 + " "; }
        if (apellido2 != null) { this.nombreCompleto += apellido2; }

        // En caso de que no se ingrese ningún nombre ni apellido, se asigna
        "No capturado".
        if (nombre1 == null && nombre2 == null && apellido1 == null &&
apellido2 == null) {
            this.nombreCompleto = "No capturado";
        }
    }
}

```

Principal.java

```

Java
/*-----
Practica: 3 Inicialización de objetos precisando los atributos e invocación de
métodos.
Nombre: Joshua Osorio
Materia: Programación Orientada a Objetos
Fecha: Febrero/25/2025
-----*/

import java.util.Scanner;

public class Principal{
    public static void main(String [] args){

```

```

Scanner input = new Scanner(System.in);
Empleado generico = new Empleado();

// Se declaran las variables tipo string en null para poder
realizar operaciones.
String primerNombre = null, segundoNombre = null, primerApellido =
null, segundoApellido = null;
int dia, mes, anio, salarioSemanal, opcion;

// Inicio el ciclo do while para estar mostrando siempre las
opciones.
do{
    // Se despliega información para mostrar las opciones.
    System.out.flush();
    System.out.printf("\n\n\tSistema para capturar informacion
de un Empleado\n");
    System.out.println("\n\t-----Menú-----\n");
    System.out.println("\t1 - Modificar primer nombre");
    System.out.println("\t2 - Modificar segundo nombre");
    System.out.println("\t3 - Modificar apellido paterno");
    System.out.println("\t4 - Modificar apellido materno");
    System.out.println("\t5 - Modificar fecha de nacimiento");
    System.out.println("\t6 - Modificar salario semanal");
    System.out.println("\t7 - Mostrar sueldo mensual");
    System.out.println("\t8 - Mostrar información");
    System.out.println("\t0 - Salir\n");

    // Se captura la opción que desea.
    System.out.printf("\tOpción:\t");
    opcion = input.nextInt();

    // Evaluar la opción para ingresar a la opción deseada.
    input.nextLine();
    switch (opcion){
        case 1:
            System.out.println("\n\n\t\tModificar primer
nombre\t");
            System.out.printf("\t\tAnterior:\t%s\n",
primerNombre);

            System.out.printf("\t\tNombre:\t");
            primerNombre = input.nextLine();
            generico.setNombreCompleto(primerNombre,
segundoNombre, primerApellido, segundoApellido);

```

```

        System.out.printf("\t\tCambio a:\t%s",
primerNombre);

        System.out.printf("\n\t\tNombre completo
cambio a:\t%s", generico.getNombreCompleto());
        break;
    case 2:

        System.out.println("\n\n\t\tModificar segundo
nombre\t");

        System.out.printf("\t\tAnterior:\t%s\n",
segundoNombre);

        System.out.printf("\t\tNombre:\t");
        segundoNombre = input.nextLine();
        generico.setNombreCompleto(primerNombre,
segundoNombre, primerApellido,segundoApellido);
        System.out.printf("\t\tCambio a:\t%s",
segundoNombre);

        System.out.printf("\n\t\tNombre completo
cambio a:\t%s", generico.getNombreCompleto());
        break;
    case 3:
        System.out.println("\n\n\t\tModificar apellido
paterno\t");

        System.out.printf("\n\t\tAnterior:\t%s\n",
primerApellido);

        System.out.printf("\t\tApellido:\t");
        primerApellido = input.nextLine();
        generico.setNombreCompleto(primerNombre,
segundoNombre, primerApellido,segundoApellido);
        System.out.printf("\t\tCambio a:\t%s",
primerApellido);

        System.out.printf("\n\t\tNombre completo
cambio a:\t%s", generico.getNombreCompleto());
        break;
    case 4:
        System.out.println("\n\n\t\tModificar apellido
materno\t");

        System.out.printf("\n\t\tAnterior:\t%s\n",
segundoApellido);

        System.out.printf("\t\tApellido:\t");
        segundoApellido = input.nextLine();
        generico.setNombreCompleto(primerNombre,
segundoNombre, primerApellido,segundoApellido);

```

```

        System.out.printf("\t\tCambio a:\t%s",
segundoApellido);

        System.out.printf("\n\t\tNombre completo
cambio a:\t%s", generico.getNombreCompleto());
        break;
    case 5:
        System.out.println("\n\n\t\tModificar fecha de
nacimiento:\t");

        System.out.printf("\n\t\tAnterior:\t%d / %d /
%d\n\t", generico.getDia(), generico.getMes(), generico.getMes());

        System.out.printf("\tDia:\t");
        dia = input.nextInt();
        System.out.printf("\t\tMes:\t");
        mes = input.nextInt();
        System.out.printf("\t\tAño:\t");
        anio = input.nextInt();

        generico.setDia(dia);
        generico.setMes(mes);
        generico.setAnio(anio);

        System.out.printf("\t\tNueva:\t%d / %d /
%d\n", generico.getDia(), generico.getMes(), generico.getAnio());
        break;
    case 6:
        System.out.println("\n\n\t\tModificar salario
semanal:\t");

        System.out.printf("\t\tAnterior:\t%d
\n\t", generico.getSalarioSemanal());
        System.out.printf("\tSalario:\t");
        salarioSemanal = input.nextInt();
        generico.setSalarioSemanal(salarioSemanal);

        System.out.printf("\t\tNuevo:\t%d\n", generico.getSalarioSemanal());
        break;
    case 7:
        System.out.println("\n\n\t\tMostrar salario
mensual\t");

        if(generico.getSalarioSemanal() != 0){
            System.out.printf("\t\tSalario
mensual:\t%d", generico.getSalarioMensual());
        }else{

```

```

                                System.out.println("\t\tFavor de
capturar antes salario semanal\t");
                                }
                                break;
                                case 8:
                                    System.out.printf("\t\tMostrar
Informacion\t\n");
                                    System.out.printf("\t\tNombre
completo:\t%s\n", generico.getNombreCompleto());
                                    System.out.printf("\t\tFecha de
nacimiento:\t%d / %d / %d\n", generico.getDia(), generico.getMes(),
generico.getMes());
                                    System.out.printf("\t\tSalario
semanal:\t%d\n", generico.getSalarioSemanal());
                                    System.out.printf("\t\tSalario
mensual:\t%d\n", generico.getSalarioMensual());
                                    break;
                                case 0:
                                    // Opcion para salir del menu muestra mensaje y
finaliza programa.
                                    System.out.printf("\tSaliendo\t");
                                    break;
                                default:
                                    // Solo sale del ciclo si se ingresa un 0.
                                    System.out.printf("\tOpción no válida, intente
nuevamente.\t");
                                    break;
                                }
                                }while(opcion != 0);
                            }
                        }

```

Conclusiones

Durante la práctica se realizaron sistemas para gestionar información de libros y empleados, empleado clases y modulando lo que facilitó la organización y manipulación de los datos, permitiendo estructuras más claras, reutilizables y escalables.

Durante el desarrollo, surgieron problemas con la captura de cadenas de texto después de ingresar un número. Este inconveniente se debió a que `nextInt()` no realiza el salto de línea, lo cual genera errores en capturas posteriores. Para solucionarlo, se implementó `nextLine()` antes

del switch case, asegurando una captura correcta y evitando inconsistencias en la entrada del usuario.

Además en la clase empleado, se implementó un método para concatenar el nombre, lo que permitió mejorar la presentación de la información y su manejo dentro del sistema.

En general, la práctica permitió reforzar conceptos clave vistos en clase, como la encapsulación y el uso de métodos para modificar y acceder a los atributos de una clase, mejorando la comprensión sobre cómo estructurar programas más organizados y eficientes.

Bibliografía

- [1] «1.Introduccion a POO.pdf», *Google Docs*.
https://drive.google.com/file/d/1xYBDI61nS_FjrNAyBf3jpeDaRw19A2ix/view
- [2] «2. Sintaxis de Java.pdf», *Google Docs*.
<https://drive.google.com/file/d/1-mdalYxfs4frHhzLQEUmowy4ARSe5rPj/view>
- [3] «Unidad 1. Apuntes clase creacion de objetos.pdf», *Google Docs*.
https://drive.google.com/file/d/1O0bdZqxIk6VxgvwcteewPfvsk2Lyx3b_/view