



**CENTRO UNIVERSITARIO UAEM
ATLACOMULCO**

MATERIA

PARADIGMAS DE LA PROGRAMACIÓN

DOCENTE

JULIO ALBERTO DE LA TEJA LÓPEZ

ALUMNO

ALEXIS VALENCIA MARTÍNEZ

CARRERA

**LICENCIATURA EN INGENIERÍA EN
SISTEMAS COMPUTACIONALES**

ICO-27

FECHA DE ENTREGA

16/AGOSTO/2023

- **INTRODUCCIÓN**

El encapsulamiento y la herencia son dos conceptos fundamentales en la programación orientada a objetos. El encapsulamiento se refiere a la práctica de ocultar y proteger información dentro de un objeto, mientras que la herencia permite a los programadores reutilizar código y extender funcionalidades a través de relaciones entre clases. Estos conceptos son esenciales para crear programas bien estructurados y fáciles de mantener

- **CÓDIGO**

```
class Libro {  
  
    private String titulo;  
  
    private String autor;  
  
    private int anioPublicacion;  
  
  
    public Libro(String titulo, String autor, int anioPublicacion) {  
  
        this.titulo = titulo;  
  
        this.autor = autor;  
  
        this.anioPublicacion = anioPublicacion;  
    }  
  
  
    public String getTitulo() {  
  
        return titulo;  
    }  
  
  
    public void setTitulo(String titulo) {  
  
        this.titulo = titulo;  
    }  
  
  
    public String getAutor() {  
  
        return autor;  
    }  
  
  
    public void setAutor(String autor) {  
  
        this.autor = autor;  
    }  
}
```

```
public int getAnioPublicacion() {  
    return anioPublicacion;  
}  
  
public void setAnioPublicacion(int anioPublicacion) {  
    this.anioPublicacion = anioPublicacion;  
}  
}  
  
class Prestamo extends Libro {  
    private String fechaPrestamo;  
    private String fechaDevolucion;  
  
    public Prestamo(String titulo, String autor, int anioPublicacion, String fechaPrestamo) {  
        super(titulo, autor, anioPublicacion);  
        this.fechaPrestamo = fechaPrestamo;  
    }  
  
    public String getFechaPrestamo() {  
        return fechaPrestamo;  
    }  
  
    public void setFechaPrestamo(String fechaPrestamo) {  
        this.fechaPrestamo = fechaPrestamo;  
    }  
  
    public String getFechaDevolucion() {  
        return fechaDevolucion;  
    }  
  
    public void setFechaDevolucion(String fechaDevolucion) {  
        this.fechaDevolucion = fechaDevolucion;  
    }  
}
```

```

class Main {

    public static void main(String[] args) {

        Libro libro = new Libro("Don Quijote de la Mancha", "Miguel de Cervantes", 1605);

        System.out.println("Titulo: " + libro.getTitulo());

        System.out.println("Autor: " + libro.getAutor());

        System.out.println("Año de Publicación: " + libro.getAnioPublicacion());


        Prestamo prestamo = new Prestamo("Don Quijote de la Mancha", "Miguel de Cervantes", 1605, "16/08/2023");

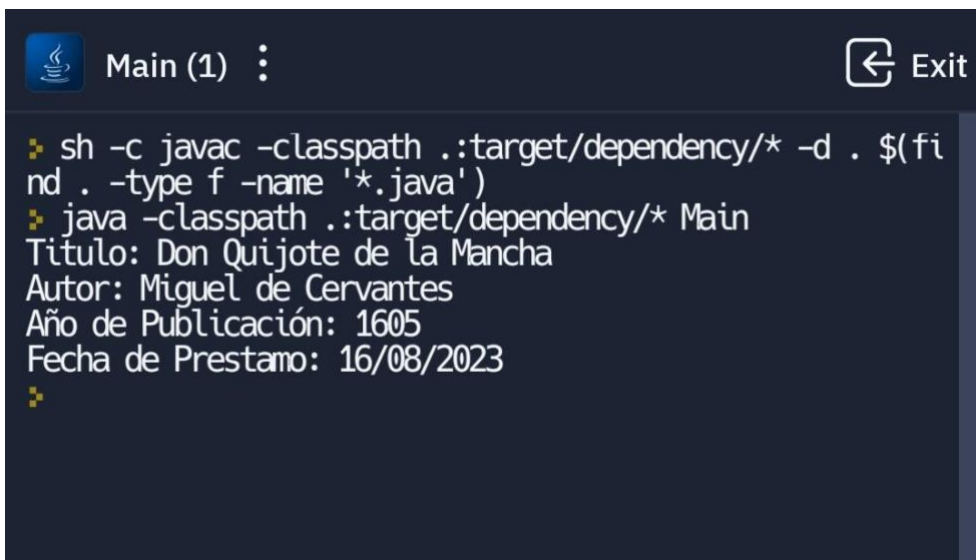
        System.out.println("Fecha de Prestamo: " + prestamo.getFechaPrestamo());

    }

}

```

- **SALIDA**



```

Main (1)  ⌵
❏ sh -c javac -classpath .:target/dependency/* -d . $(fi
❏ nd . -type f -name '*.java')
❏ java -classpath .:target/dependency/* Main
Titulo: Don Quijote de la Mancha
Autor: Miguel de Cervantes
Año de Publicación: 1605
Fecha de Prestamo: 16/08/2023
❏

```

- **CONCLUSIÓN**

Tanto el encapsulamiento como la herencia tienen rasgos/características un tanto similares, así como lo es su uso para el <<almacenamiento>> de los datos y el como se muestra a el usuario, así como su facilidad de uso para el programador y el fácil entendimiento para el usuario

- **FUENTES ADICIONALES DE APOYO**

Herencia

<https://es.ccm.net/aplicaciones-e-internet/museo-de-internet/enciclopedia/10700-que-es-la-herencia-en-programacion/>

La herencia en programación

<https://es.ccm.net/aplicaciones-e-internet/museo-de-internet/enciclopedia/10694-que-es-el-encapsulamiento-en-programacion/>

Herencia en programación: características, tipos, ejemplos

<https://www.lifeder.com/herencia-programacion/>