



## **MODIFICADORES DE ACCESO**

**Alumno:**

Gómez Faustino Paola | 01296824

**Profesor:**

M.C. Claudia Tona

**Grupo:**

544

**Materia:**

Programación orientada a objetos

## Investiga sobre los cuatro modificadores de acceso disponibles en Java:

Los modificadores de acceso son palabras clave que se utilizan para controlar el alcance de una clase, un método, una variable o un constructor en Java. En Java, existen cuatro modificadores de acceso:

**public:** permite el acceso a los miembros desde cualquier otra clase o paquete.

**private:** restringe el acceso a los miembros solo dentro de la misma clase.

**protected:** permite el acceso a los miembros desde la misma clase, clases del mismo paquete y subclases

**Default:** (sin modificador o private-package): es el modificador de acceso predeterminado cuando no se especifica ninguno

Elabora un cuadro comparativo donde resumas el comportamiento de cada modificador de acceso en cuanto a su visibilidad:



Modificador	Misma Clase	Mismo paquete	Subclase	Subclase (distinto paquete)	otra clase
Publico	✓	✓	✓	✓	✓
Protegido	✓	✓	✓	✓	✗
Por defecto	✓	✓	✓	✗	✗
Privado	✓	✗	✗	✗	✗

## Ejemplos imples

```
J Ejemplos.java > ...  
1 public class Ejemplos {  
2     public String mensaje = "¡Accesible desde cualquier lugar!";  
3 }  
4 |
```

```
J Ejemplos.java > Language Support for Java(TM) by Red Hat > Ejemplos  
1 public class Ejemplos {  
2     private String secreto = "Solo accesible dentro de esta clase.";  
3 }  
4 |  
5 }
```

```
J Ejemplos.java > ...  
1 public class Ejemplos {  
2     protected String mensajeProtegido = "Accesible en el mismo paquete y subclases.";  
3 }  
4 |  
5 }
```

```
J Ejemplos.java > ...  
1 public class Ejemplos {  
2     String mensajePorDefecto = "Accesible solo dentro del mismo paquete.";  
3 }  
4 |  
5 }  
6 |
```

### Código de uso de modificador de acceso.

```
public class ClaseA {  
  
    public String variablePublica = "Variable Pública";  
  
    private String variablePrivada = "Variable Privada";  
  
    protected String variableProtegida = "Variable Protegida";  
  
    String variablePorDefecto = "Variable por Defecto";  
  
    public void mostrarAcceso() {  
  
        System.out.println("Pública: " + variablePublica);  
  
        System.out.println("Privada: " + variablePrivada);  
  
        System.out.println("Protegida: " + variableProtegida);  
  
        System.out.println("Por Defecto: " + variablePorDefecto);  
  
    }  
  
}
```

```
public class ClaseB extends ClaseA {  
  
    public void probarAcceso() {  
  
        System.out.println("Pública: " + variablePublica); // Accesible  
  
        // System.out.println("Privada: " + variablePrivada); // No accesible  
  
        System.out.println("Protegida: " + variableProtegida); // Accesible por  
herencia  
  
        // System.out.println("Por Defecto: " + variablePorDefecto);  
  
    }  
  
}
```

```

public class Principal {

    public static void main(String[] args) {

        // Instancia de ClaseA

        ClaseA claseA = new ClaseA();

        System.out.println("Acceso desde ClaseA:");

        claseA.mostrarAcceso();


        // Instancia de ClaseB

        ClaseB claseB = new ClaseB();

        System.out.println("\nAcceso desde ClaseB:");

        claseB.probarAcceso();

    }

}

```

```

Acceso desde ClaseA:
Pública: Variable Pública
Privada: Variable Privada
Protegida: Variable Protegida
Por Defecto: Variable por Defecto

```

```

Acceso desde ClaseB:
Pública: Variable Pública
Protegida: Variable Protegida

```

## **Conclusión:**

Los modificadores de acceso son claves para organizar el código y proteger los datos. Usar `private` ayuda a evitar que alguien modifique más información que no debería, mientras que `protected` permite compartir lo necesario con clases relacionadas sin exponer todo.

Saber cuándo usar cada modificador hace que el código sea más claro y fácil de mantener, evitando errores y problemas de seguridad. En general, son una herramienta que ayuda a que el desarrollo de programas sea más ordenado y seguro.

## **Bibliografía:**

Oracle . (nd). Control de acceso ( Tutoriales de Java™) . Oracle . Recuperado de <https://docs.oracle.com/javase/tutorial/java/javaOO/accesscontrol.html>

Programación Fácil. (Dakota del Norte). *Modificadores de acceso en Java: Público, Privado y Predeterminado* . Recuperado [de https://programacionfacil.org/blog/modificadores-de-acceso-en-java-public-private-y-default/](https://programacionfacil.org/blog/modificadores-de-acceso-en-java-public-private-y-default/)