

Práctica Nro 12 de SIS103

Nombre: Gonzales Franz Reinaldo

C.U. 111-500

Carrera: Ing. en Ciencias de la Computación

Polimorfismo:

Clase Profesor:

```
package polimorfismo;

/**
 * Esta clase denominada Profesor es una superclase que representa un
 * profesor genérico.
 *
 * @version 1.2/2020
 */
public class Profesor {
    /**
     * Método que imprime en pantalla un texto específico identificando
     * que el objeto es un Profesor
     */
    protected void imprimir() {
        System.out.println("Es un profesor.");
    }
}
```

Clase ProfesorTitular

```
package polimorfismo;

/**
 * Esta clase denominada Profesor es una superclase que representa un
 * profesor genérico.
 *
 * @version 1.2/2020
 */
public class Profesor {
    /**
     * Método que imprime en pantalla un texto específico identificando
     * que el objeto es un Profesor
     */
    protected void imprimir() {
        System.out.println("Es un profesor.");
    }
}
```

Clase Prueba:

```
package polimorfismo;

/**
 * Esta clase prueba las clase Profesor y ProfesorTitular utilizando el
 * polimorfismo
 *
 * @version 1.2/2020
 */
public class Prueba {
    /**
     * Método main que crea un Profesor pero instanciando la clase
     * ProfesorTitular. ¿Qué se imprimirá en pantalla?
     */
    public static void main(String[] args) {
        Profesor profesor1 = new ProfesorTitular();
        profesor1.imprimir();
    }
}
```

Prueba de ejecución:

```
PS C:\GONZALES\SIS103> java polimorfismo.Prueba
Es un profesor titular.
PS C:\GONZALES\SIS103> █
```

Ejercicios Propuestos:

Clase prueba:

```
public class Prueba {
    public static void main(String[] args) {
        Profesor profesor1 = new ProfesorTitular();
        Profesor profesor2 = (Profesor) profesor1;
        profesor2.imprimir();
    }
}
```

Prueba de ejecución:

```
PS C:\GONZALES\SIS103> java polimorfismo.Prueba
Es un profesor titular.
PS C:\GONZALES\SIS103> java polimorfismo.Prueba
Es un profesor titular.
PS C:\GONZALES\SIS103> █
```

Ejercicio 2 de Ejercicios de propuesta

Clase Profesor:

```
package polimorfismo;

/**
 * Esta clase denominada Profesor es una superclase que representa un
 * profesor genérico.
 *
 * @version 1.2/2020
 */
public class Profesor {
    /**
     * Método que imprime en pantalla un texto específico identificando
     * que el objeto es un Profesor
     */
    protected void imprimir() {
        System.out.println("Es un profesor.");
    }
}
```

Clase ProfesorTitular

```
package polimorfismo;

/**
 * Esta clase denominada ProfesorTitular es una subclase de Profesor
 *
 * @version 1.2/2020
 */
public class ProfesorTitular extends Profesor {
    /**
     * Método que sobrescribe el método imprimir heredado de la clase
     * padre Profesor
     */
    protected void imprimir() {
        System.out.println("Es un profesor titular.");
    }
}
```

Clase Prueba2

```
package Polimorfismo;

/**
 * Esta clase prueba las clases Profesor y ProfesorTitular utilizando la
 * conversión descendente
 *
 * * @version 1.2/2020
 */
public class Prueba2 {
    /**
     * Método main que crea un ProfesorTitular pero instanciando la
     * clase Profesor. ¿Qué se imprimirá en pantalla? ¿Compilará el
     * programa?
     */
    public static void main(String[] args) {
        ProfesorTitular objeto = new Profesor();
        objeto.imprimir();
    }
}
```

Prueba de ejecución:

```
PS C:\GONZALES\SIS103> java polimorfismo.Prueba2
Error: Could not find or load main class polimorfismo.Prueba2
Caused by: java.lang.ClassNotFoundException: polimorfismo.Prueba2
PS C:\GONZALES\SIS103> █
```

Como podemos ver en la ejecución no es posible de ejecutar en asignar un objeto a la instancia de la variable Profesor titular.

Ejercicios de propuesta #2

Clase Prueba

```
package Polimorfismo;

public class Prueba {
    public static void main(String[] args) {
        Profesor profesor1 = new ProfesorTitular();
        ProfesorTitular profesor2 = (Profesor) profesor1;
        profesor2.imprimir();
    }
}
```

Prueba de ejecución

```
PS C:\GONZALESGIS103> java polimorfismo.Prueba2
Error: Could not find or load main class polimorfismo.Prueba2
Caused by: java.lang.ClassNotFoundException: polimorfismo.Prueba2
PS C:\GONZALESGIS103>
```

Nos permite la implementación de un objeto ProfesorTitular a una variable Profesor, pero al contrario vemos que no nos permite la asignación de un objeto Profesor a una variable ProfesorTitular.

Ejercicio Propuestos #3 Polimorfismo

Clase Profesor:

```
public class Profesor {
    /**
     * Método que imprime en pantalla un texto específico identificando
     * que el objeto es un Profesor
     */
    protected void imprimir() {
        System.out.println("Es un profesor.");
    }
}
```

Clase ProfesorTitular:

```
public class ProfesorTitular extends Profesor {

    int anos = 0;
    protected void imprimir() {
        System.out.println("Es un profesor titular.");
    }

    protected void imprimirAnos() {
        System.out.println("Años = " + anos);
    }
}
```

Clase Prueba3

```
package Polimorfismo;

/**
 * Esta clase prueba las clases Profesor y ProfesorTitular utilizando
 * métodos polimórficos
 *
 * @version 1.2/2020
 */
public class Prueba3 {
    /**
     * Método main que crea un ProfesorTitular pero en un objeto tipo
     * Profesor.
     */
}
```

```

    * ¿Qué se imprimirá en pantalla? ¿Compilará el programa?
    */
    public static void main(String[] args) {
        Profesor profesor1 = new ProfesorTitular();
        profesor1.imprimirAños();
    }
}

```

Prueba de ejecución:

```

PS C:\GONZALES\SIS103> java polimorfismo.Prueba3
Error: Could not find or load main class polimorfismo.Prueba3
Caused by: java.lang.ClassNotFoundException: polimorfismo.Prueba3
PS C:\GONZALES\SIS103>

```

Como se ve en la ejecución no permite implementar ProfesorTitular a una variable Profesor pero asignar la variable de tipo Profesor no contiene la función imprimirAnos() dentro de la clase.

Ejercicio de Propuesto del #3

Clase Prueba:

```

import java.util.*;

public class Prueba {

    Vector profesores;
    public static void main(String[] args) {
        Prueba prueba = new Prueba();
        prueba.profesores = new Vector();
        Profesor profesor1 = new Profesor();
        ProfesorTitular profesor2 = new ProfesorTitular();
        prueba.profesores.add(profesor1);
        prueba.profesores.add(profesor2);

        for(int i = 0; i < prueba.profesores.size(); i++) {
            Profesor p = (Profesor) prueba.profesores.elementAt(i);
            p.imprimir();
        }
    }
}

```

Prueba de ejecución

```

PS C:\GONZALES\SIS103> java polimorfismo.Prueba
Es un profesor titular.
PS C:\GONZALES\SIS103>

```