

CURSO DE PROGRAMACIÓN CON JAVA

EJERCICIO

EJEMPLO DE POLIMORFISMO EN JAVA



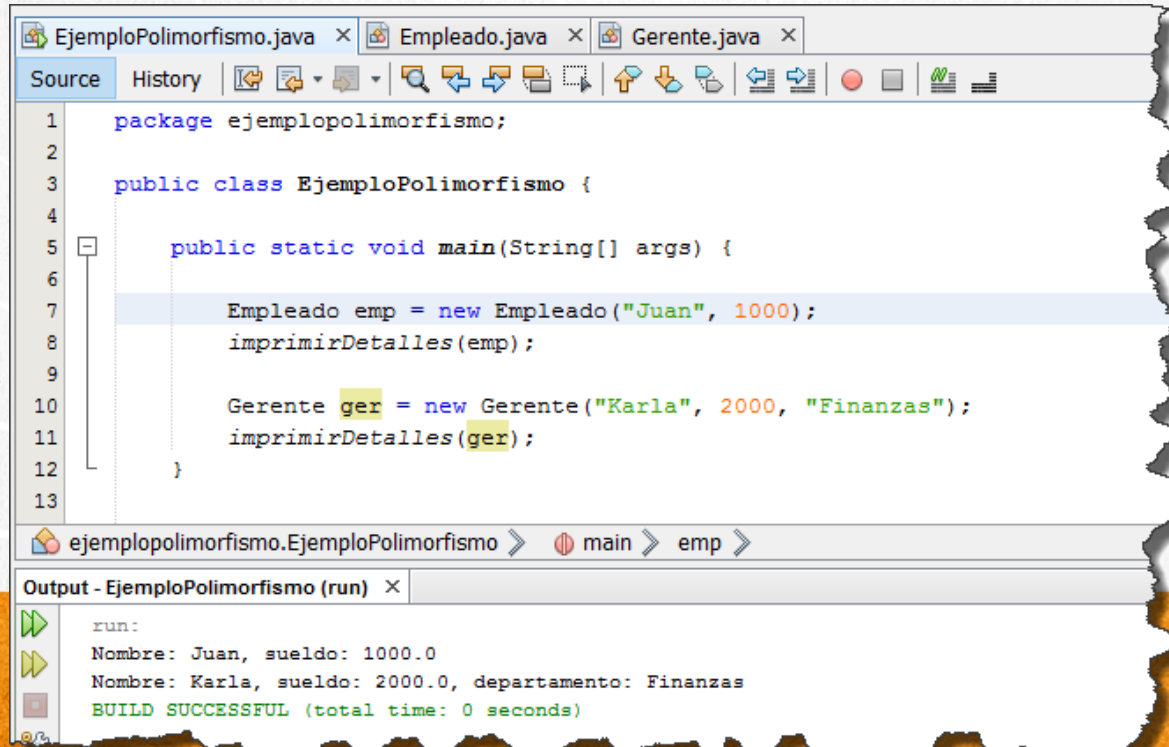
Experiencia y Conocimiento para tu vida

CURSO DE PROGRAMACIÓN CON JAVA

www.globalmentoring.com.mx

OBJETIVO DEL EJERCICIO

Crear un programa para practicar polimorfismo en Java. Al finalizar deberemos observar lo siguiente:



The screenshot shows an IDE with three tabs: EjemploPolimorfismo.java, Empleado.java, and Gerente.java. The main window displays the source code of EjemploPolimorfismo.java. The code defines a package, a public class with a main method, and instantiates objects of Empleado and Gerente classes. The output window at the bottom shows the execution results, confirming that the program runs successfully and prints the details of the created objects.

```
1 package ejemplopolimorfismo;
2
3 public class EjemploPolimorfismo {
4
5     public static void main(String[] args) {
6
7         Empleado emp = new Empleado("Juan", 1000);
8         imprimirDetalles(emp);
9
10        Gerente ger = new Gerente("Karla", 2000, "Finanzas");
11        imprimirDetalles(ger);
12    }
13 }
```

ejemplopolimorfismo.EjemploPolimorfismo > main > emp >

Output - EjemploPolimorfismo (run) x

```
run:
Nombre: Juan, sueldo: 1000.0
Nombre: Karla, sueldo: 2000.0, departamento: Finanzas
BUILD SUCCESSFUL (total time: 0 seconds)
```

PASO 1. CREACIÓN DEL PROYECTO

Vamos a crear el proyecto:

New Java Application

Steps

1. Choose Project
2. **Name and Location**

Name and Location

Project Name: EjemploPolimorfismo

Project Location: C:\Cursos\ProgramacionJava\Leccion11 Browse...

Project Folder: ProgramacionJava\Leccion11\EjemploPolimorfismo

☐ Use Dedicated Folder for Storing Libraries

Libraries Folder: Browse...

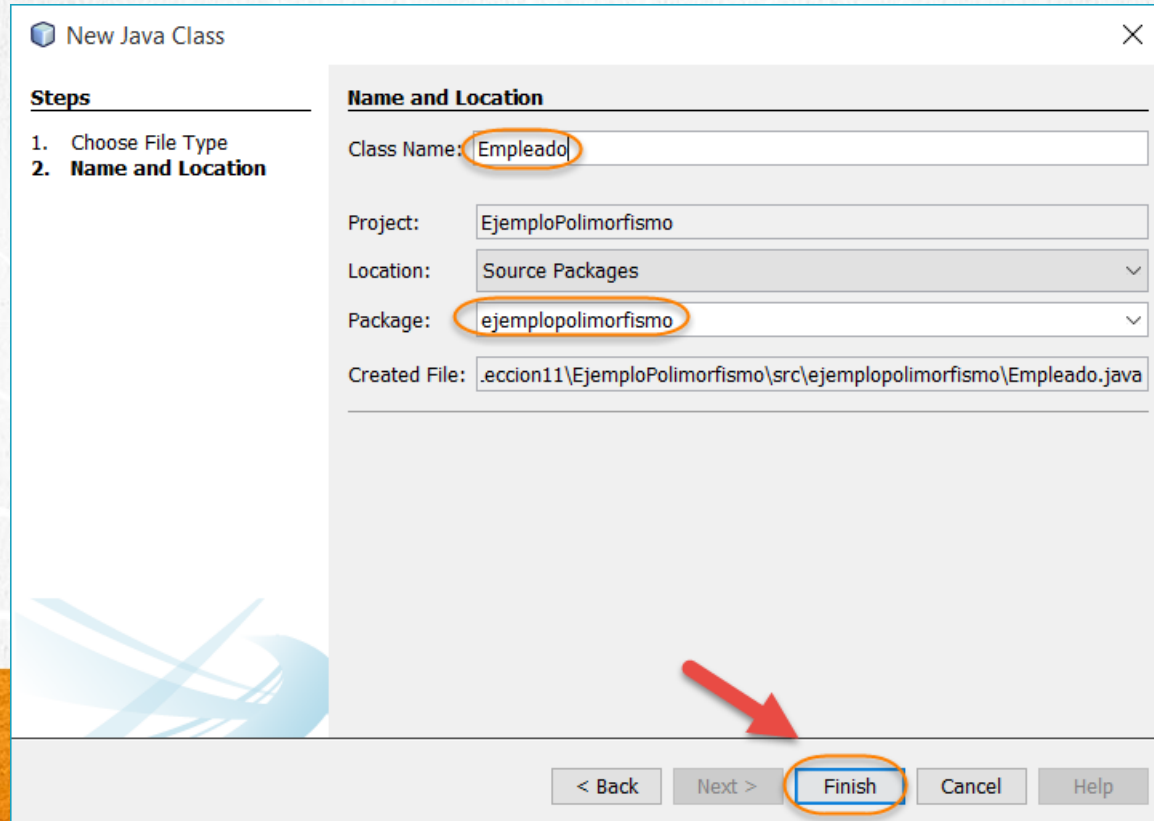
Different users and projects can share the same compilation libraries (see Help for details).

☒ Create Main Class ejemplopolimorfismo.EjemploPolimorfismo

< Back Next > **Finish** Cancel Help

PASO 2. AGREGAMOS UNA CLASE

Agregamos una clase al proyecto:



New Java Class

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

Class Name:

Project:

Location:

Package:

Created File:

< Back Next > **Finish** Cancel Help

PASO 3. AGREGAMOS UNA CLASE

Agregamos una clase al proyecto:

New Java Class

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

Class Name:

Project:

Location:

Package:

Created File:

< Back Next > **Finish** Cancel Help

PASO 4. MODIFICAMOS EL CÓDIGO

Archivo Empleado.java:

```
package ejemplopolimorfismo;

public class Empleado {
    protected String nombre;
    protected double sueldo;

    protected Empleado(String nombre, double sueldo){
        this.nombre = nombre;
        this.sueldo = sueldo;
    }

    public String obtenerDetalles(){
        return "Nombre: " + nombre +
            ", sueldo: " + sueldo;
    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public double getSueldo() {
        return sueldo;
    }

    public void setSueldo(double sueldo) {
        this.sueldo = sueldo;
    }
}
```

PASO 5. MODIFICAMOS EL CÓDIGO

Archivo Gerente.java:

```
package ejemplopolimorfismo;

public class Gerente extends Empleado{

    private String departamento;

    public Gerente(String nombre, double sueldo, String departamento) {
        super(nombre, sueldo);
        this.departamento = departamento;
    }

    //Sobreescribimos el metodo padre heredado
    public String obtenerDetalles(){
        //Observamos que para no repetir codigo, podemos utilizar
        //el metodo del padre y solo agregar a este metodo de la clase hija
        //esto es invocar un metodo sobreescrito
        return super.obtenerDetalles() + ", departamento: " + departamento;
    }

    public String getDepartamento() {
        return departamento;
    }

    public void setDepartamento(String departamento) {
        this.departamento = departamento;
    }
}
```

PASO 6. MODIFICAMOS EL CÓDIGO

Archivo EjemploPolimorfismo.java:

```
package ejemplopolimorfismo;

public class EjemploPolimorfismo {

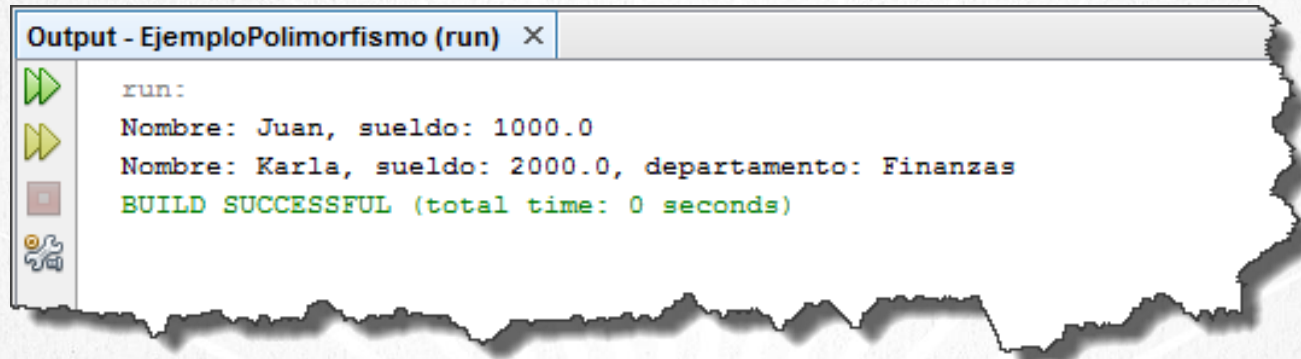
    public static void main(String[] args) {

        Empleado emp = new Empleado("Juan", 1000);
        imprimirDetalles(emp);

        Gerente ger = new Gerente("Karla", 2000, "Finanzas");
        imprimirDetalles(ger);
    }

    //Observamos que el tipo que recibe el metodo es de tipo padre (Empleado)
    //Sin embargo al momento de ejecutar el metodo, se ejecuta el del hijo (Gerente)
    //eso es el polimorfismo, multiples formas pero en tiempo de ejecucion
    public static void imprimirDetalles(Empleado emp) {
        System.out.println(emp.obtenerDetalles());
    }
}
```


PASO 7. EJECUTAMOS EL PROYECTO



CONCLUSIÓN DEL EJERCICIO

Hemos puesto en práctica el concepto de polimorfismo, así como la sobreescritura de métodos y la llamada a métodos ocultos debido a la sobreescritura utilizando la palabra super.

Vimos que para aplicar el concepto de polimorfismo es necesario apoyarnos de una jerarquía de clases, y utilizar un tipo de datos que pueda almacenar referencias de objetos en la jerarquía de clases deseada.

CURSO ONLINE

PROGRAMACIÓN CON JAVA

Por: Ing. Ubaldo Acosta



Experiencia y Conocimiento para tu vida

www.globalmentoring.com.mx