

# EJERCICIO 2-OPERACIONES

---

Este archivo lleva la estructura de los 4 pilares en POO aplicados a un ejercicio de Operaciones, esto con el fin de ser más claro al momento de presentar el ejercicio, este md solo lleva la estructura del ejercicio 2, los demás están en su respectiva carpeta

## 1. Herencia

---

### Operaciones

```
package Modelo;

public class Operaciones {
    private String nombre;
    private String operador;
    private double num1;
    private double num2;
    private double resultado;

    //Implementación del método
    public void Visualizar(){
        this.nombre = "Suma";
        this.operador = "+";
        this.num1 = 8;
        this.num2 = 2;
        this.resultado = num1+num2;
        System.out.println("Nombre de la operación: "+this.nombre+"\n" +
            "Operador de la operación: "+this.operador+"\n"+
            "Primer número elegido: "+this.num1+"\n"+
            "Segundo número elegido: "+this.num2+"\n"+
            "Resultado = "+this.resultado+"\n");
    }
}
```

### Resta

```
package Modelo;

public class Resta extends Operaciones {

}
```

### Multiplicacion

```
package Modelo;

public class Multiplicacion extends Operaciones{

}
```

## Division

```
package Modelo;

public class Division extends Operaciones{

}
```

## Ejecutar

```
package Vista;
import Modelo.Operaciones;
import Modelo.Resta;
import Modelo.Multiplicacion;
import Modelo.Division;

public class Ejecutar {
    public static void main(String[] args) {
//Mostrar datos
        System.out.println(" Probando Herencia");
        Operaciones suma = new Operaciones();
        suma.Visualizar();

        Resta resta = new Resta();
        resta.Visualizar();

        Multiplicacion multiplicacion = new Multiplicacion();
        multiplicacion.Visualizar();

        Division division = new Division();
        division.Visualizar();
    }
}
```

## 2. Encapsulamiento

---

### Operaciones

```
package Modelo;

public class Operaciones {
    private String nombre;
    private String operador;
    private double num1;
    private double num2;
    private double resultado;

    //Set
    public void setNombre(String nombre){
        this.nombre = nombre;
    }
    public void setOperador(String operador){
        this.operador = operador;
    }
    public void setNum1(double num1){
        this.num1 = num1;
    }
    public void setNum2(double num2){
        this.num2 = num2;
    }
    public void setResultado(double resultado){
        this.resultado = resultado;
    }
    //Get
    public String getNombre(){
        return this.nombre;
    }
    public String getOperador(){
        return this.operador;
    }
    public double getNum1(){
        return this.num1;
    }
    public double getNum2(){
        return this.num2;
    }
    public double getResultado(){
        return this.resultado;
    }
    //Implementación del método
    public void Visualizar(){
        System.out.println("Nombre de la operación: "+this.getNombre()+"\n" +
            "Operador de la operación: "+this.getOperador()+"\n"+
            "Primer número elegido: "+this.getNum1()+"\n"+
            "Segundo número elegido: "+this.getNum2()+"\n"+
            "Resultado = "+this.getResultado()+"\n");
    }
}
```

```
package Modelo;

public class Resta extends Operaciones {

}
```

## Multiplificacion

```
package Modelo;

public class Multiplicacion extends Operaciones{

}
```

## Division

```
package Modelo;

public class Division extends Operaciones{

}
```

## Ejecutar

```
package Vista;
import Modelo.Operaciones;
import Modelo.Resta;
import Modelo.Multiplicacion;
import Modelo.Division;

import java.util.Scanner;

public class Ejecutar {
    public static void main(String[] args) {
        Scanner leer = new Scanner(System.in);
        //Mostrar datos
        System.out.println("Clase Padre");

        Operaciones suma = new Operaciones();
        suma.setNombre("Suma");
        suma.setOperador("+");
        System.out.println("Son dos números para hacer la operación"+
            "\nIngrese el primero: ");
        double num1Suma = leer.nextDouble();
        suma.setNum1(num1Suma);
```

```
System.out.println("Ingrese el segundo número");
double num2Suma = leer.nextDouble();
suma.setNum2(num2Suma);
suma.setResultado(num1Suma+num2Suma);
suma.Visualizar();

System.out.println("Clase Hija Resta");

Resta resta = new Resta();
resta.setNombre("Resta");
resta.setOperador("-");
System.out.println("Son dos números para hacer la operación"+
    "\nIngrese el primero: ");
double num1Resta = leer.nextDouble();
resta.setNum1(num1Resta);
System.out.println("Ingrese el segundo número");
double num2Resta = leer.nextDouble();
resta.setNum2(num2Resta);
resta.setResultado(num1Resta-(num2Resta));
resta.Visualizar();

System.out.println("Clase Hija Multiplicación");

Multiplicacion multiplicacion = new Multiplicacion();
multiplicacion.setNombre("Multiplicación");
multiplicacion.setOperador("*");
System.out.println("Son dos números para hacer la operación"+
    "\nIngrese el primero: ");
double num1Multiplicacion = leer.nextDouble();
multiplicacion.setNum1(num1Multiplicacion);
System.out.println("Ingrese el segundo número");
double num2Multiplicacion = leer.nextDouble();
multiplicacion.setNum2(num2Multiplicacion);
multiplicacion.setResultado(num1Multiplicacion*num2Multiplicacion);
multiplicacion.Visualizar();

System.out.println("Clase Hija Division");

Division division = new Division();
division.setNombre("División");
division.setOperador("/");
System.out.println("Son dos números para hacer la operación"+
    "\nIngrese el primero: ");
double num1Division = leer.nextDouble();
division.setNum1(num1Division);
System.out.println("Ingrese el segundo número (debe ser mayor a 0)");
double num2Division = leer.nextDouble();
division.setNum2(num2Division);
division.setResultado(num1Division/num2Division);
division.Visualizar();
    }
}
```

## 3. Polimorfismo

---

### Operaciones

```
package Modelo;

public class Operaciones {
    private String nombre;
    private String operador;
    private double num1;
    private double num2;
    private double resultado;

    //Set
    public void setNombre(String nombre){
        this.nombre = nombre;
    }
    public void setOperador(String operador){
        this.operador = operador;
    }
    public void setNum1(double num1){
        this.num1 = num1;
    }
    public void setNum2(double num2){
        this.num2 = num2;
    }
    public void setResultado(double resultado){
        this.resultado = resultado;
    }
    //Get
    public String getNombre(){
        return this.nombre;
    }
    public String getOperador(){
        return this.operador;
    }
    public double getNum1(){
        return this.num1;
    }
    public double getNum2(){
        return this.num2;
    }
    public double getResultado(){
        return this.resultado;
    }
    //Implementación del método
    public void Visualizar(){
        System.out.println("Nombre de la operación: "+this.getNombre()+"\n" +
            "Operador de la operación: "+this.getOperador()+"\n"+
            "Primer número elegido: "+this.getNum1()+"\n"+
            "Segundo número elegido: "+this.getNum2()+"\n"+
```

```
        "Resultado = "+this.getResultado()+"\n");
    }
}
```

## Resta

```
package Modelo;

public class Resta extends Operaciones {

    @Override
    public void Visualizar(){
        System.out.println("La operación es la "+this.getNombre()+", su operador
es (" +this.getOperador()+")."+
        "\nEl primer número elegido: "+this.getNum1()+"\n"+
        "Y este es el segundo: "+this.getNum2()+"\n"+
        "Resultado = "+this.getResultado()+"\n");
    }
}
```

## Multiplacion

```
package Modelo;

public class Multiplicacion extends Operaciones{

    @Override
    public void Visualizar(){
        System.out.println("La "+this.getNombre()+" es una operación
aritmética."+"\n"+
        "Su operador es: "+this.getOperador()+"\n"+
        "Digitaste como primer número: "+this.getNum1()+" y como segundo
el: "+this.getNum2()+"\n"+
        "Resultado = "+this.getResultado()+"\n");
    }
}
```

## Division

```
package Modelo;

public class Division extends Operaciones{

    @Override
    public void Visualizar(){
        System.out.println(this.getNombre()+" termino donde dividir entre 0 no es
```

```

    posible."+"\n"+
        "Tiene como operador: "+this.getOperador()+"\n"+
        "----Su primer número: "+this.getNum1()+"----+"\n"+
        "----Su segundo número: "+this.getNum2()+"----+"\n"+
        "Resultado = "+this.getResultado()+":")+"\n");
    }
}

```

## Ejecutar

```

package Vista;
import Modelo.Operaciones;
import Modelo.Resta;
import Modelo.Multiplicacion;
import Modelo.Division;

import java.util.Scanner;

public class Ejecutar {
    public static void main(String[] args) {
        Scanner leer = new Scanner(System.in);
//Mostrar datos
        System.out.println("Clase Padre");

        Operaciones suma = new Operaciones();
        suma.setNombre("Suma");
        suma.setOperador("+");
        System.out.println("Son dos números para hacer la operación"+
            "\nIngrese el primero: ");
        double num1Suma = leer.nextDouble();
        suma.setNum1(num1Suma);
        System.out.println("Ingrese el segundo número");
        double num2Suma = leer.nextDouble();
        suma.setNum2(num2Suma);
        suma.setResultado(num1Suma+num2Suma);
        suma.Visualizar();

        System.out.println("Clase Hija Resta");

        Resta resta = new Resta();
        resta.setNombre("Resta");
        resta.setOperador("-");
        System.out.println("Son dos números para hacer la operación"+
            "\nIngrese el primero: ");
        double num1Resta = leer.nextDouble();
        resta.setNum1(num1Resta);
        System.out.println("Ingrese el segundo número");
        double num2Resta = leer.nextDouble();
        resta.setNum2(num2Resta);
        resta.setResultado(num1Resta-(num2Resta));
        resta.Visualizar();
    }
}

```



```
System.out.println("Clase Hija Multiplicación");

Multiplicacion multiplicacion = new Multiplicacion();
multiplicacion.setNombre("Multiplicación");
multiplicacion.setOperador("*");
System.out.println("Son dos números para hacer la operación"+
    "\nIngrese el primero: ");
double num1Multiplicacion = leer.nextDouble();
multiplicacion.setNum1(num1Multiplicacion);
System.out.println("Ingrese el segundo número");
double num2Multiplicacion = leer.nextDouble();
multiplicacion.setNum2(num2Multiplicacion);
multiplicacion.setResultado(num1Multiplicacion*num2Multiplicacion);
multiplicacion.Visualizar();

System.out.println("Clase Hija Division");

Division division = new Division();
division.setNombre("División");
division.setOperador("/");
System.out.println("Son dos números para hacer la operación"+
    "\nIngrese el primero: ");
double num1Division = leer.nextDouble();
division.setNum1(num1Division);
System.out.println("Ingrese el segundo número (debe ser mayor a 0)");
double num2Division = leer.nextDouble();
division.setNum2(num2Division);
division.setResultado(num1Division/num2Division);
division.Visualizar();
    }
}
```

## 4.Abstracción

---

### Operaciones

```
package Modelo;

public abstract class Operaciones {
    private String nombre;
    private String operador;
    private double num1;
    private double num2;
    private double resultado;

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

```
public void setOperador(String operador){
    this.operador = operador;
}
public void setNum1(double num1){
    this.num1 = num1;
}
public void setNum2(double num2){
    this.num2 = num2;
}
public void setResultado(double resultado){
    this.resultado = resultado;
}
//Get
public String getNombre(){
    return this.nombre;
}
public String getOperador(){
    return this.operador;
}
public double getNum1(){
    return this.num1;
}
public double getNum2(){
    return this.num2;
}
public double getResultado(){
    return this.resultado;
}
//Implementación del método
public void Visualizar(){
    System.out.println("Nombre de la operación: "+this.getNombre()+"\n" +
        "Operador de la operación: "+this.getOperador()+"\n"+
        "Primer número elegido: "+this.getNum1()+"\n"+
        "Segundo número elegido: "+this.getNum2()+"\n"+
        "Resultado = "+this.getResultado()+"\n");
}
}
```

## Resta

```
package Modelo;

public class Resta extends Operaciones {
    private String parte;

    public String getParte(){
        return parte;
    }

    public void setParte(String parte){
        this.parte = parte;
    }
}
```

```

    }

    @Override
    public void Visualizar(){
        System.out.println("La operación es la "+this.getNombre()+", su operador
es (" +this.getOperador()+").\n"+
            "\nSabías que en la resta el "+this.getParte()+" es el número al
que se le resta otra cantidad.\n"+
            "\nEl primer número elegido: "+this.getNum1()+"\n"+
            "Y este es el segundo: "+this.getNum2()+"\n"+
            "Resultado = "+this.getResultado()+"\n");
    }
}

```

## Multiplicacion

```

package Modelo;

public class Multiplicacion extends Operaciones{
    private String propiedad;

    public String getPropiedad(){
        return propiedad;
    }

    public void setPropiedad(String propiedad){
        this.propiedad = propiedad;
    }

    @Override
    public void Visualizar(){
        System.out.println("La "+this.getNombre()+" es una operación
aritmética."+"\n"+
            "\nLa multiplicación tiene 4 propiedades.\n"+
            "Una de ellas es la propiedad: "+this.getPropiedad()+"\n"+
            "\nSu operador es: "+this.getOperador()+"\n"+
            "Digitaste como primer número: "+this.getNum1()+" y como segundo
el: "+this.getNum2()+"\n"+
            "Resultado = "+this.getResultado()+"\n");
    }
}

```

## Division

```

package Modelo;

public class Division extends Operaciones{
    private String simbolo;

```

```

    public String getSimbolo(){
        return simbolo;
    }

    public void setSimbolo(String simbolo){
        this.simbolo = simbolo;
    }

    @Override
    public void Visualizar(){
        System.out.println(this.getNombre()+" termino donde dividir entre 0 no es
posible."+"\\n"+
            "Tiene como operador: "+this.getOperador()+"\\n"+
            "----Su primer número: "+this.getNum1()+"----"+"\\n"+
            "----Su segundo número: "+this.getNum2()+"----"+"\\n"+
            "Resultado = "+this.getResultado()+":)"+"\\n"+
            "\\nSu otro signo conocido es: "+this.getSimbolo());
    }
}

```

## Ejecutar

```

package Vista;
import Modelo.Resta;
import Modelo.Multiplicacion;
import Modelo.Division;

import java.util.Scanner;

public class Ejecutar {
    public static void main(String[] args) {
        Scanner leer = new Scanner(System.in);
        //Mostrar datos
        System.out.println("Clase Hija Resta");

        Resta resta = new Resta();
        resta.setNombre("Resta");
        resta.setOperador("-");
        resta.setParte("minuendo");
        System.out.println("Son dos números para hacer la operación"+
            "\\nIngrese el primero: ");
        double num1Resta = leer.nextDouble();
        resta.setNum1(num1Resta);
        System.out.println("Ingrese el segundo número");
        double num2Resta = leer.nextDouble();
        resta.setNum2(num2Resta);
        resta.setResultado(num1Resta-(num2Resta));
        resta.Visualizar();

        System.out.println("Clase Hija Multiplicación");
    }
}

```

```
Multiplicacion multiplicacion = new Multiplicacion();
multiplicacion.setNombre("Multiplicación");
multiplicacion.setPropiedad("conmutativa");
multiplicacion.setOperador("*");
System.out.println("Son dos números para hacer la operación"+
    "\nIngrese el primero: ");
double num1Multiplicacion = leer.nextDouble();
multiplicacion.setNum1(num1Multiplicacion);
System.out.println("Ingrese el segundo número");
double num2Multiplicacion = leer.nextDouble();
multiplicacion.setNum2(num2Multiplicacion);
multiplicacion.setResultado(num1Multiplicacion*num2Multiplicacion);
multiplicacion.Visualizar();

System.out.println("Clase Hija Division");

Division division = new Division();
division.setNombre("División");
division.setOperador("/");
System.out.println("Son dos números para hacer la operación"+
    "\nIngrese el primero: ");
double num1Division = leer.nextDouble();
division.setNum1(num1Division);
System.out.println("Ingrese el segundo número (debe ser mayor a 0)");
double num2Division = leer.nextDouble();
division.setNum2(num2Division);
division.setResultado(num1Division/num2Division);
division.setSimbolo("÷");
division.Visualizar();
    }
}
```