



**UNRN**

Universidad Nacional  
de Río Negro



| unrionegro

---

# Programación 2

## V

**UNRN**

Universidad Nacional  
de **Río Negro**

Martín René Vilugrón  
[mrvilugron@unrn.edu.ar](mailto:mrvilugron@unrn.edu.ar)

# IntelliJ IDEA

Capable and Ergonomic Java IDE

**JET  
BRAINS**







**¿Preguntas?**

# Pasaje de argumentos

# Por un lado tenemos

```
public static int suma(int n, int m){  
    n += m;  
    return n;  
}
```

## ¿Cómo quedan las variables en 'test'?

```
public static int suma(int n, int m){  
    n += m;  
    return n;  
}
```

```
public static void test(){  
    int a = 10;  
    int b = 20;  
    int resultado = suma(a, b);  
    System.out.println(resultado);  
}
```



# **En Java, todo es por copia**

---

**Esto es mucho muy  
importante**



**¿Preguntas?**

# Y ahora

# La suma de elementos de un arreglo

# Suma de los elementos del arreglo

```
public static int suma(int[] arreglo){  
    for(int i = arreglo.length-1; i>0; i--){  
        arreglo[0] += arreglo[i];  
    }  
    return arreglo[0];  
}
```



**¿Es legal eso?**

## ¿Se obtiene el resultado?

```
public static int suma(int[] arreglo){
    for(int i = arreglo.length-1; i>0; i--){
        arreglo[0] += arreglo[i];
    }
    return arreglo[0];
}

public static void test(){
    int[] areglo = {1,2,3,4}; // la suma debiera dar 10
    int resultado = suma(areglo);
    System.out.println(resultado);
}
```





**¿Preguntas?**

---

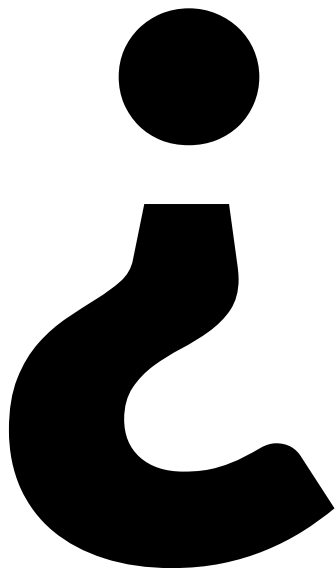
# ¿Cuales son los efectos secundarios?

# Suma de los elementos del arreglo

```
public static int suma(int[] arreglo){  
    for(int i = arreglo.length-1; i>0; i--){  
        arreglo[0] += arreglo[i];  
    }  
    return arreglo[0];  
}
```

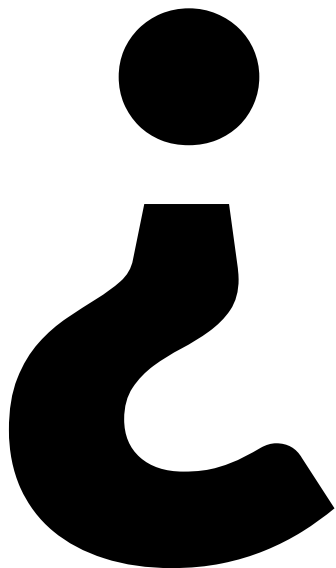
# Suma de los elementos del arreglo

```
/**
 * Devuelve la suma de los elementos del arreglo
 * @param arreglo contiendo los valores que deseamos sumar
 * @return la suma de los elementos del arreglo
 * #PRE: el arreglo debe ser válido y contener por lo menos un valor
 * #POST: el arreglo debe quedar exactamente igual a como entró
 */
public static int suma(int[] arreglo){
    for(int i = arreglo.length-1; i>0; i--){
        arreglo[0] += arreglo[i];
    }
    return arreglo[0];
}
```



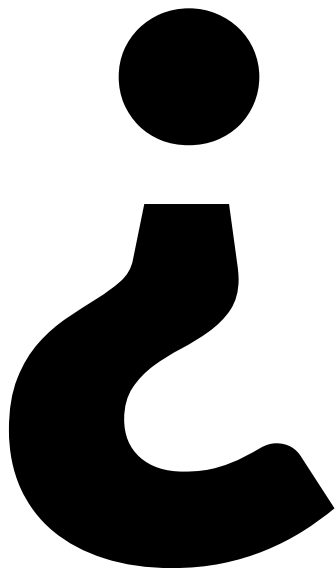
**por que sucede  
esto**





**punteros**





**punteros**



**¿punteros?**



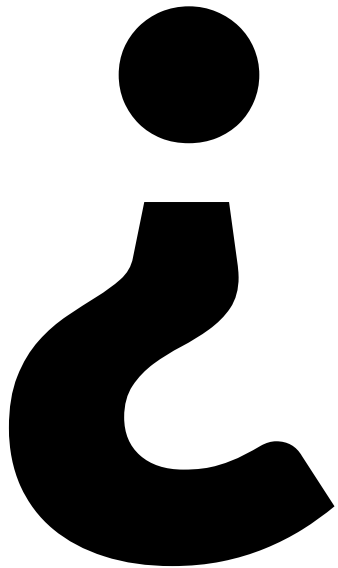
**NOPE!**



**Chuck Testa**

tenemos  
**referencias**

**Que son parecidos pero no iguales**



**Qué operaciones**  
**definen** a los  
**punteros**

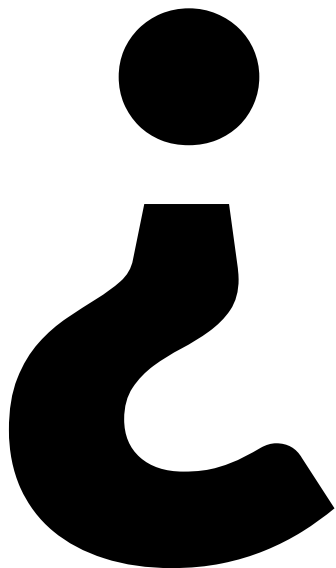


**en punteros**

**aritmética**

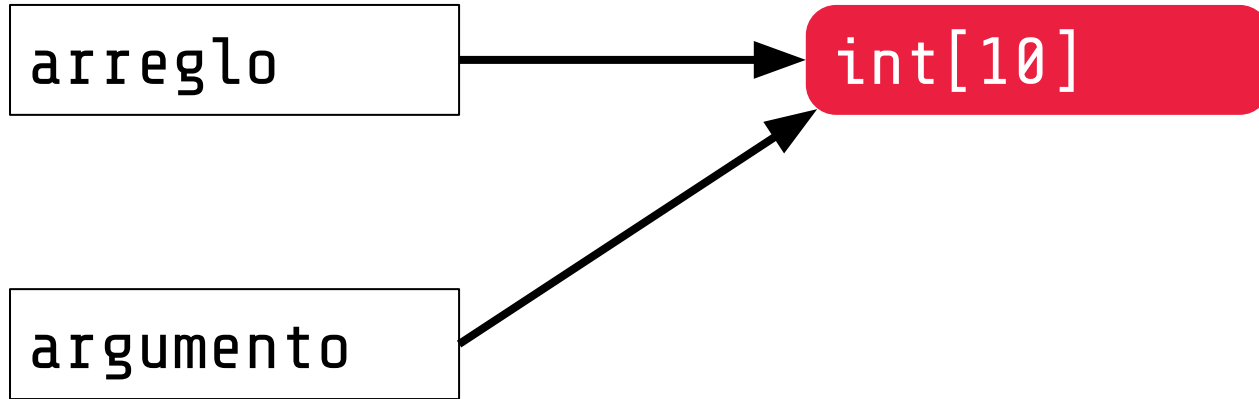
**en referencias**

**~~aritmética~~**

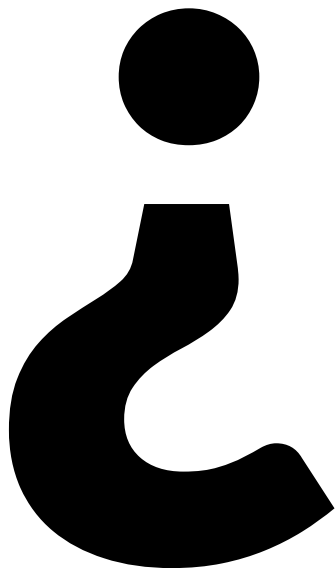


**Que quiere decir  
esto**





**Ambos van a parar al mismo arreglo**



**Como lo podemos  
evitar**





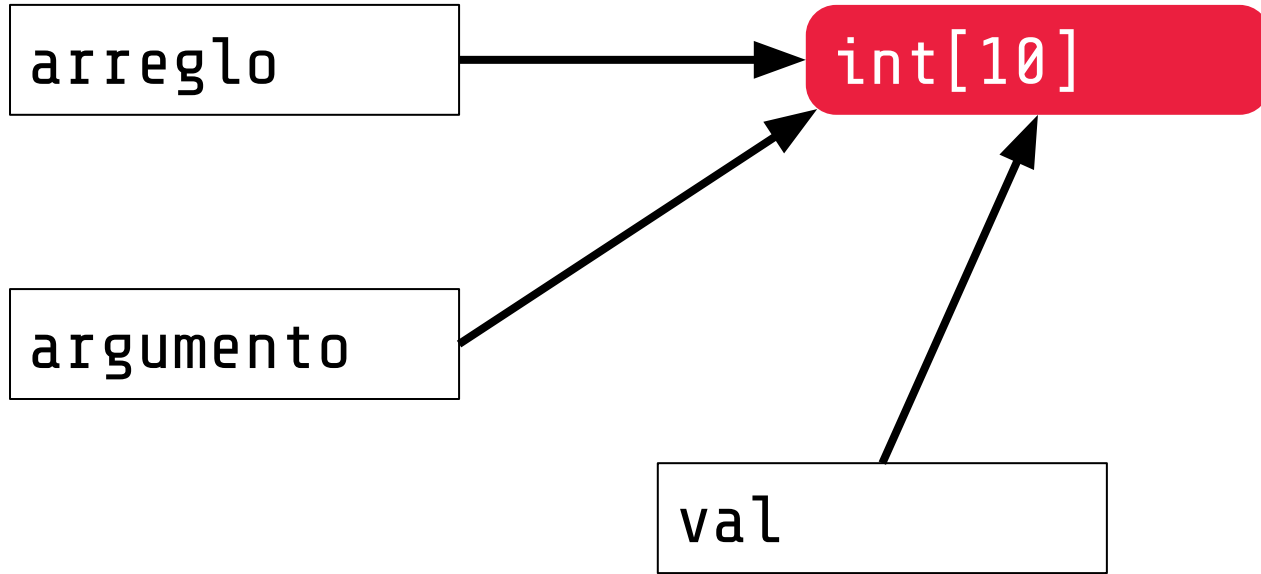
# Y ahora, ¿como queda el arreglo fuera?

```
public static int suma(int[] arreglo){  
    int[] val = arreglo;  
    for(int i = val.length - 1; i>0; i--){  
        val[0] += val[i];  
    }  
    return val[0];  
}
```

**¿ Funciona ?**

---

# nope



**Los tres van a parar al mismo arreglo**

**Solo creamos una  
nueva referencia al  
arreglo**



**¿Preguntas?**

—

¿Y entonces?  
¿Qué tenemos que  
hacer?

```
int[] arr2 = new  
int[arr.length]  
for (int i = 0;  
i<arr.length; i++){  
    arr2[i] = arr[i]  
}
```

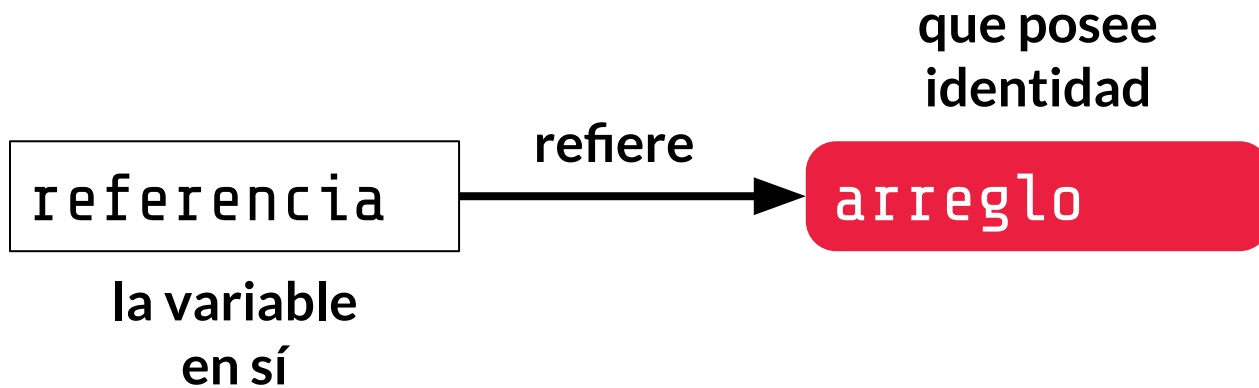
**Copiar los elementos en un arreglo nuevo**



**0 usar**  
**Arrays.copyOf**

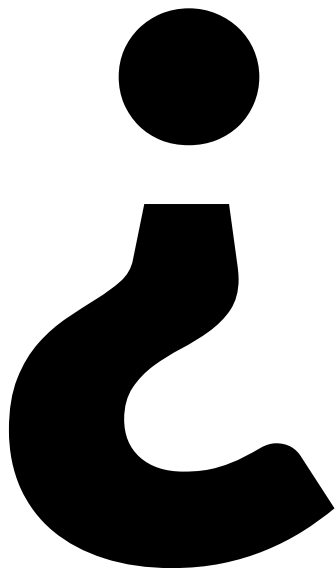


**¿Preguntas?**





**¿Preguntas?**



**Y String**



# La tercera via

```
public static String concatenar(String dst, String src){  
    dst = dst + src;  
    return dst;  
}
```

# La tercera via

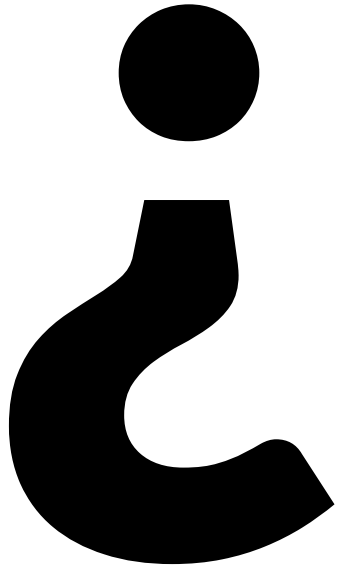
```
public static String concatenar(String dst, String src){  
    dst = dst + src;  
    return dst;  
}
```

```
public static void testCadenas(){  
    String uno = "CADENA";  
    String dos = "cadena";  
    String resultado = concatenar(uno, dos);  
    System.out.println(resultado);  
}
```

**Los String  
son inmutables**



**La identidad de cada  
cadena es diferente**



**cuando tenemos  
que tener en  
cuenta esto**



**cuando cambien  
valores que no  
debieran de  
cambiar**



**¿Preguntas?**

# Solo nos queda

---

# null

**La referencia no válida  
y su valor por defecto**

# ¿Que contiene arreglo?

```
int[] arreglo;  
for(int i = 0; i < arreglo.length; i++){  
    System.out.print(arreglo[i]);  
}
```

**¡No hay nada!**





**¿Preguntas?**



**Ya que estamos**

# for-each

```
for (int item: arr){  
    //for each  
}
```



**¿Preguntas?**

**unrn.edu.ar**

