

Introducción a la Programación

Clase 02



INTRODUCCIÓN A LA PROGRAMACIÓN



Agenda

- Repaso de la lección anterior.
- Revisión de los ejercicios.
- Estructuras de decisión



1

Escriba un programa que **lea** el nombre del trabajador y su salario semanal, con esta información **calcule** su salario mensual sin deducciones (salario bruto), el monto de las deducciones (que corresponden a un 9.34%) y el salario después de aplicadas las deducciones (salario neto), finalmente debe **mostrar** los datos al usuario de la siguiente manera.

Estimado <<nombre>>, el salario de este mes se desglosa de la siguiente manera.

Salario bruto: #####.##

Deducciones: #####.##

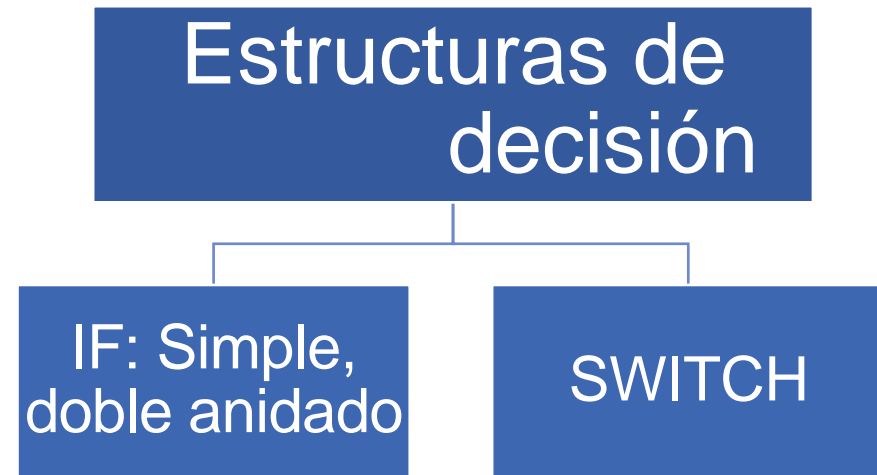
Salario Neto: #####.##

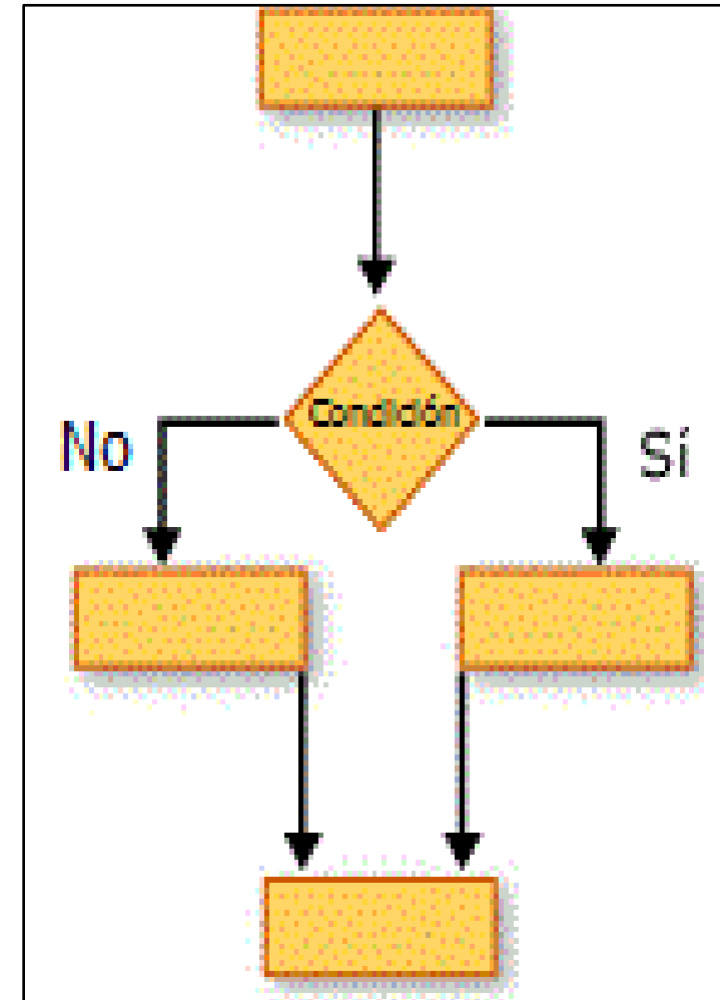
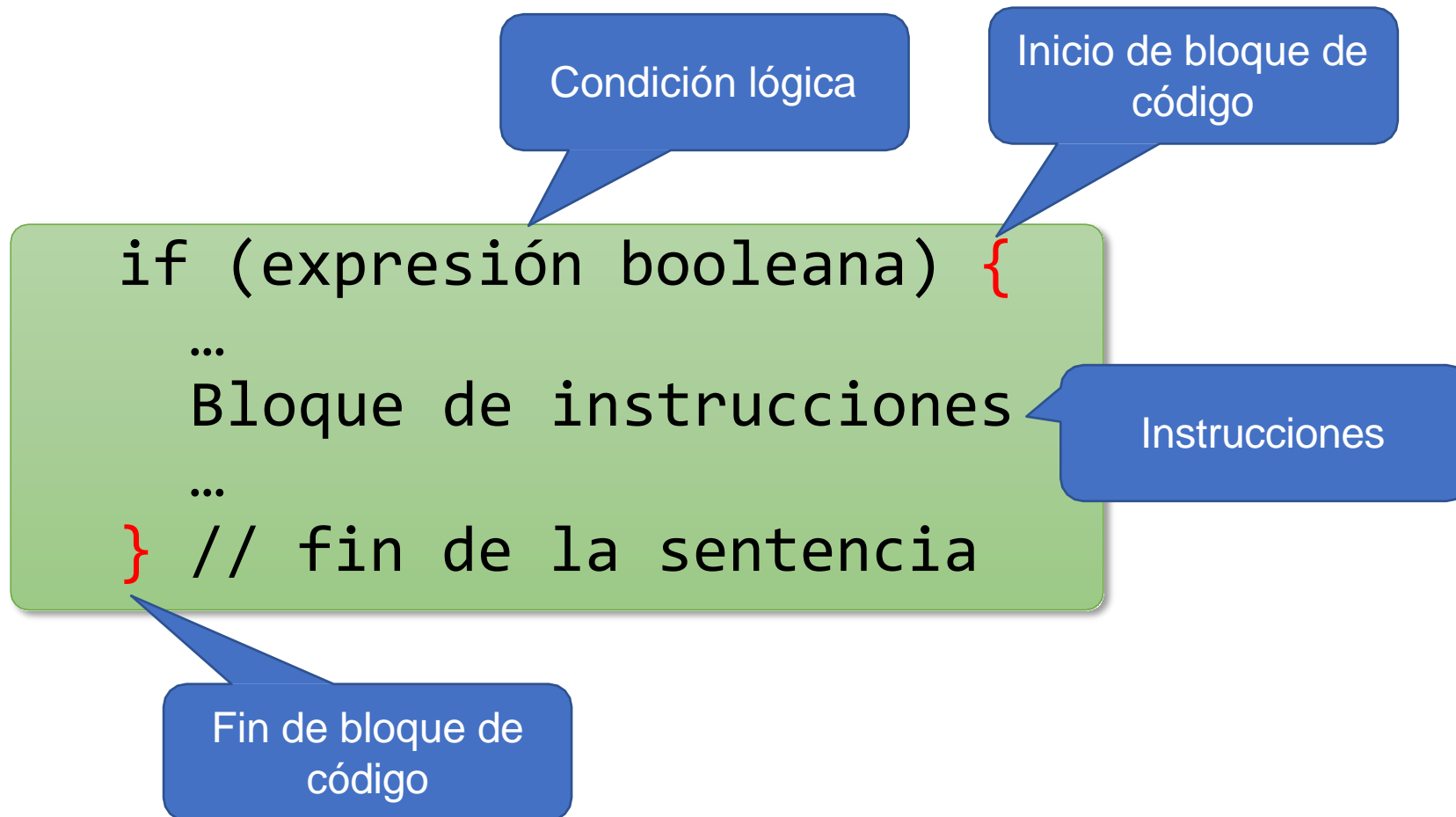
2

1. Escriba un programa que calcule la suma y el promedio de cuatro números de tipo int. Muestre los resultados. Usted le asigna los valores a las variables.
2. Modifique el programa que realizó en el punto anterior, de modo que lea los valores que se van a sumar.

INTRODUCCIÓN A LA PROGRAMACIÓN

Las estructuras de decisión son sentencias que nos permiten tomar decisiones con base en el valor de una condición o en el valor de una variable.





OPERADORES RELACIONALES

- La expresión booleana que necesita el **if** puede ser una sola variable de tipo boolean o una expresión en la cual se utilizan operadores relacionales.
- Los operadores relacionales son utilizados para comparar dos valores y determinar su relación.
- El resultado de todos los operadores relacionales es booleano (falso o verdadero).

Condición	Operador	Ejemplo
Es igual a	==	int i=1; (i == 1)
No es igual a	!=	int i=2; (i != 1)
Es menor a	<	int i=0; (i < 1)
Es menor o igual a	<=	int i=1; (i <= 1)
Es mayor a	>	int i=2; (i > 1)
Es mayor o igual a	>=	int i=1; (i >= 1)

OPERADORES LÓGICOS

- En muchas ocasiones una decisión de programación se debe tomar evaluando más de una comparación booleana, para esto utilizamos los operadores lógicos **o**, **y**, **negación**.
- Los operadores lógicos en Java son los siguientes:
 - **||** indica o
 - **&&** para el y
 - **!** para la negación

1

- Escriba un nuevo programa en JAVA que le solicite al usuario su edad y le indique si puede o no puede votar.
- *Considere que en nuestro país, solo las personas con 18 años o más pueden votar.*

2

- Escriba un nuevo programa en JAVA que le solicite al usuario la nota final del curso anterior y le indique si aprobó o no.
- *Considere que para aprobar debe obtener una nota igual o superior a 70.*

ANIDACIÓN DE if

Dentro de las instrucciones de un bloque de código de un **if** puede haber otro **if**.

```
if (expresión booleana) {  
    ...  
    if(expresión booleana) {  
        ...  
        Bloque de instrucciones  
        ...  
    }  
}
```

La ejecución del código correspondiente al IF interno solo se ejecutará si se cumplen las condiciones de los dos IF, por lo tanto podría no votar y tampoco ser candidato, podría votar y no poder ser candidato y podría votar y ser candidato.

```
if (edad >= 18) {  
    System.out.println("Puede votar");  
    if (edad >= 31) {  
        System.out.println("Puede ser presidente");  
    }  
}
```

Identifique 3 edades con las cuales podríamos probar todas las posibilidades

- Eventualmente en un programa se requiere hacer una cosa si la condición es verdadera y otra si no lo es, por eso se creó la sentencia
- **if else.**

```
• if (expresión booleana)
{
    ...
    Bloque de instrucciones
    ...
}
else{
    ...
    Bloque de instrucciones
    ...
}
```

La cláusula ELSE adicional nos permite realizar acciones en el caso inverso al cumplimiento de la condición.

```
if (edad >= 18) {  
    System.out.println("Puede votar");  
} else {  
    System.out.println("No puede votar");  
}
```

El ELSE no necesita una condición porque se ejecutará en la condición inversa del IF

El **IF** compuesto es utilizado cuando tenemos múltiples opciones con múltiples condiciones.

```
if (mes == 2) {  
    System.out.println("Este mes tiene 28 / 29 días");  
} else if (mes == 4 || mes == 6 || mes == 9 || mes == 11) {  
    System.out.println("Este mes tiene 30 días");  
} else {  
    System.out.println("Este mes tiene 31 días");  
}
```

Hay 3 posibles opciones, la primera opción ocupa una condición, de igual manera la segunda, la tercera se basa en la exclusión de las anteriores.

1

- Escriba un programa que le solicite al usuario el número de día y le indique el nombre del día.
- *Por ejemplo 1 = Domingo, 2 = Lunes, 3 = Martes, 4 = Miércoles, 5 = Jueves, 6 = Viernes, 7 =*
- *Sábado.*

2

- Modifique el programa para que muestre cuáles días son laborales y cuales son libres.
- *A pesar de que esto está en función de horarios muy variados, considere que de Lunes a*
- *Viernes es laborable y los sábados y domingos son libres.*

3

- Escriba un programa que le solicite al usuario 4 números y le muestre cuál es el mayor.

switch

- La sentencia **SWITCH** nos permite gestionar las mismas funciones de un **IF compuesto**, su ejecución es más eficiente que el IF, pero solamente puede ser utilizado para comparaciones de igualdad.

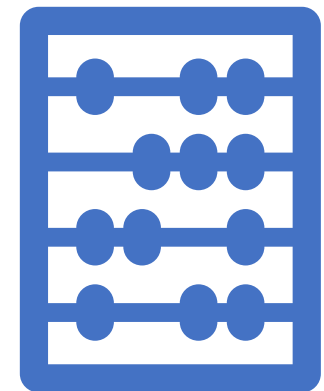
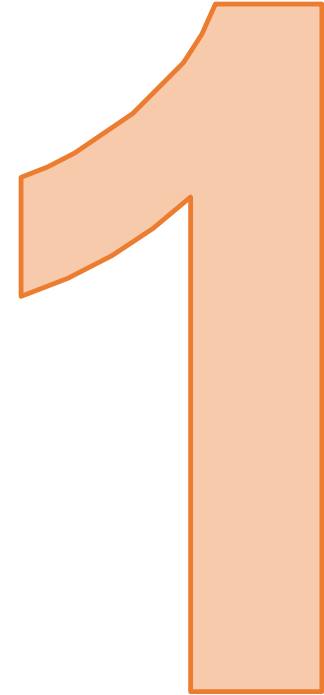
```
        System.out.println("Lunes");
        break;
case 3:
        System.out.println("Martes");
        break;
case 4:
        System.out.println("Miércoles");
        break;
case 5:
        System.out.println("Jueves");
        break;
case 6:
        System.out.println("Viernes");
        break;
default:
        System.out.println("Sábado");
        break;
```

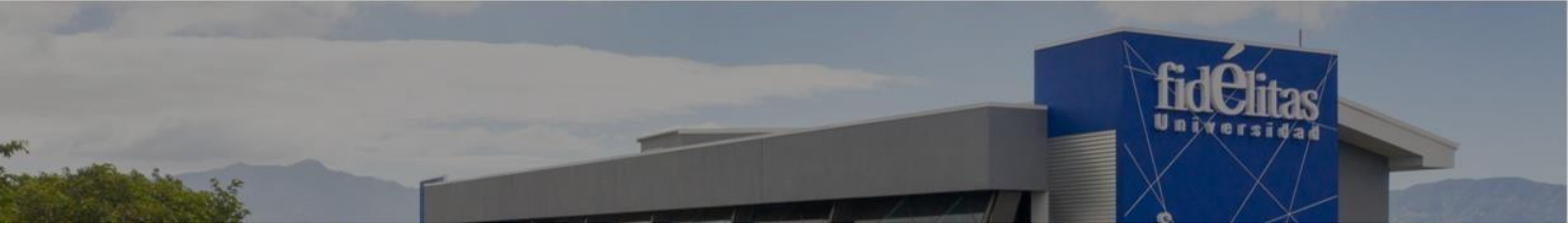
Otro ejemplo del **SWITCH**, en este caso remplazando el ejercicio de los meses, desarrollado anteriormente.

```
switch (mes) {  
    case 2:  
        System.out.println("Este mes tiene 28 / 29 días");  
        break;  
    case 4:  
    case 6:  
    case 9:  
    case 11:  
        System.out.println("Este mes tiene 30 días");  
        break;  
    default:  
        System.out.println("Este mes tiene 31 días");  
        break;  
}
```

TRABAJO EXTRACLASE

- Desarrolle un ejercicio que le solicite al usuario los años de antigüedad, la cantidad de horas laboradas por semana y el precio que se le paga por hora, con estos datos debe calcular el salario bruto.
 - Si el usuario tiene más de 12 años de trabajar para la empresa, se le dará un bono del 5.5% de su salario bruto, si este nuevo salario bruto es superior a \$2500 debe hacerse una deducción del 2.7%, pero si el salario supera los \$3500, esta deducción será del 3.8%. Muestre al usuario el salario neto final.





En los cursos de programación, es importante que el estudiante refuerce lo aprendido con mucha práctica y considerando que las dificultades y errores que se le presenten le van a ayudar a desarrollar sus habilidades de programación.

Desarrolle siempre los ejercicios de la clase buscando alternativas y probando diferentes soluciones.



“Si de todas formas vas a pensar,
mejor piensa en grande.” (D. Trump)

¡Nos vemos la próxima semana!