

# UT 1. POO, EDA y API JAVA.

Sentencias de Control de Flujo del Programa

# Sentencias de Control de Flujo del Programa

---

- Sentencia IF.
  - La sentencia **if** evalúa una expresión condicional, si el resultado es true se ejecuta un bloque de sentencias, si su resultado es false, se ejecuta otro resultado.

```
if (<expresion_condicional>){  
    <sentencias_true>  
}  
else{  
    <sentencias_false>  
}
```

# Sentencias de Control de Flujo del Programa

- If anidados.
- Cualquiera de los dos bloques de sentencias anteriores, pueden contener otras sentencias **if**. La precaución que hay que tener en estos casos es que una instrucción **else**, siempre hace referencia a la instrucción **if** más cercana

```
if (<expresion_condicional1>){  
    <sentencias_true1>  
}  
else if (<expresion_condicional2>){<sentencias_true2>  
}  
else if (<expresion_condicional3>){<sentencias_true3>  
}  
...  
else if (<expresion_condicionalN>){<sentencias_trueN>  
}
```

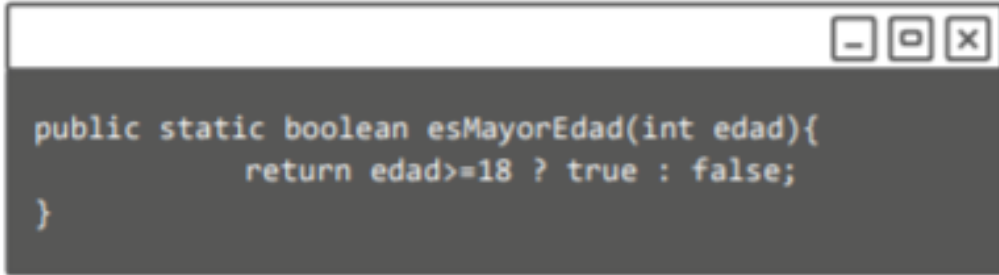
```
if( i <= 100){  
    if( j > 10) incto = - 0.5F;  
    if( k < 5) incto = 0.5F;  
    else incto = .0F; //este else hace referencia a if( k < 5 )  
}else incto = 1.5F; //este else hace referencia a if( i <= 100)
```

# Sentencias de Control de Flujo del Programa

---

- El operador ternario ?:.
  - Este operador se relaciona con sus operando de la siguiente manera:

```
<expresion_condicional> ?  
<expresion_verdadera> :  
  <expresion_falsa>;
```



```
public static boolean esMayorEdad(int edad){  
    return edad>=18 ? true : false;  
}
```

# Sentencias de Control de Flujo del Programa

- La sentencia switch.
  - Se trata de una sentencia de decisión múltiple, permite elegir entre varias alternativas. Es equivalente al uso de varias sentencias **if** anidadas.

```
switch (<expresion_switch>){  
    case <constante_1>:  
        <sentencias_1>  
        break;  
    case <constante_2>:  
        <sentencias_2>  
        break;  
    . . .  
    case <constante_N>:  
        <sentencias_3>  
        break;  
    [default:  
        <sentencias_default>]  
}
```

```
System.out.println("Teclea una opcion");  
String opcion = teclado.next(); boolean opcionvalida=  
true;  
switch(opcion){  
    case "A":      abrirArchivo();  
                   break;  
    case "B":      borrarArchivo();  
                   break;  
    case "C":      crearArchivo();  
                   break;  
    case "L":      listarArchivo();  
                   break;  
    case "X":      terminar();  
                   break;  
    default: opcionvalida=false;  
}
```

# Ejercicios de Programación en Java

---

- Ejercicio\_1.
    - Hacer un programa que lea un número entero y muestre si el número es múltiplo de 10.
  - Ejercicio\_2.
    - Pedir dos números y decir cual es el mayor o si son iguales.
  - Ejercicio\_3.
    - Hacer un programa que lea un carácter por teclado y compruebe si es una letra mayúscula.
  - Ejercicio\_4.
    - Hacer un programa que calcule lo que pagará una persona, sabiendo que aquellas personas cuya compra supere los 300€, se le aplicará un descuento del 20%.
-

# Ejercicios de Programación en Java

---

- Ejercicio\_5.
    - Realizar un programa que calcule el salario semanal de un obrero, sabiendo que:
      - Si trabaja 40 horas o menos se le paga a 16€ por hora.
      - Si trabaja más de 40 horas se le paga a 16€ por cada una de las primeras 40 horas y 20€ por cada hora extra.
  - Ejercicio\_6.
    - Hacer un programa que tome dos números y diga si ambos son pares o impares.
  - Ejercicio\_7.
    - Hacer un programa, que dados 3 números los ordene de mayor a menor.
  - Ejercicio\_8.
    - Hacer un programa que dado un número entre 0 y 99999 decir cuántas cifras tiene.
-

# Ejercicios de Programación en Java

---

- Ejercicio\_9.
    - Mostrar mensajes dependiendo de una nota.
  - Ejercicio\_10.
    - Cajero automático con menú.
-



# Sentencias de Control de Flujo del Programa

---

- El bucle for.
  - La sentencia for en su formato más general se usa para que se repita la ejecución de un bloque de sentencias un numero concreto de veces.

```
for( <expresion_inicializacion>; <expresion_
condicional>; <expresion_incremento>{
    <sentencias>
}
```

# Sentencias de Control de Flujo del Programa

- Ejemplos de uso de for.

```
int i=0, j=0;
for (i=10, j=1; i>j; i--, j++) //dos variables contadoras
    System.out.println("i: " + i + " j: " + j); //
una sentencia
//i y j siguen existiendo fuera del bucle for
System.out.println("i: " + i + " j: " + j); //después del
for
```

```
int k=0;
for( ;k<10; ){
    k++;
    System.out.println("Iteración numero:
"+ k);
}
```

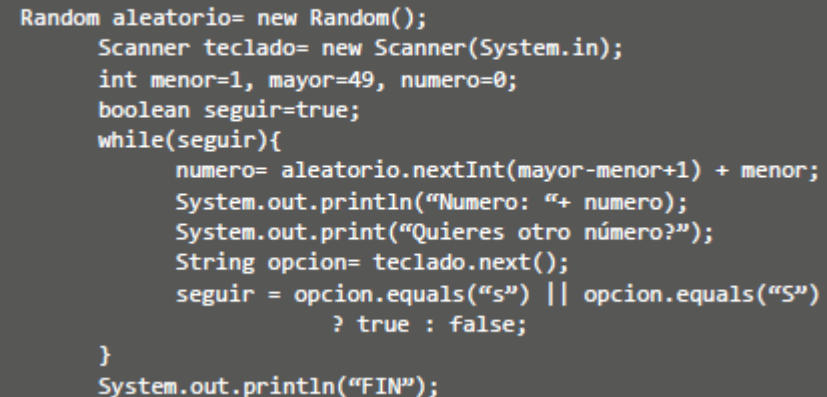
```
for( ; ; ){
    //Sentencias que se están ejecutando
    siempre
}
```

```
System.out.println("Teclea x para parar");
//System.in.read lee un carácter y lo retorna
como int
//un int son 4 char, se leen los 4 del buffer
for(i=1; ((char)System.in.read()) != 'x'; i++)
    System.out.println("Iteración número: "
+ i);
```

# Sentencias de Control de Flujo del Programa

- El bucle while.
  - EL bucle while ejecuta un bloque de sentencias si se cumple una condición determinada, cuando deja de cumplir ya no se ejecutan dichas sentencias.

```
while (<expresion_condicional>){  
    <sentencias>  
}
```



```
Random aleatorio= new Random();  
Scanner teclado= new Scanner(System.in);  
int menor=1, mayor=49, numero=0;  
boolean seguir=true;  
while(seguir){  
    numero= aleatorio.nextInt(mayor-menor+1) + menor;  
    System.out.println("Numero: "+ numero);  
    System.out.print("Quieres otro número?");  
    String opcion= teclado.next();  
    seguir = opcion.equals("s") || opcion.equals("S")  
        ? true : false;  
}  
System.out.println("FIN");
```

# Sentencias de Control de Flujo del Programa

- El bucle do while.
  - Este bucle ejecuta un bloque de sentencias mientras se cumpla una condición determinada.

```
do {  
    <sentencias>  
} while (<expresion_condicional>);
```

```
Random aleatorio= new Random();  
Scanner teclado= new Scanner(System.in);  
int menor=1, mayor=49, numero=0;  
boolean seguir=false;  
do{  
    numero= aleatorio.nextInt(mayor-menor+1) + menor;  
    System.out.println("Numero: "+ numero);  
    System.out.print("Quieres otro número?");  
    String opcion= teclado.next();  
    seguir = opcion.equals("s") || opcion.equals("S")  
        ? true : false;  
}while( seguir );  
System.out.println("FIN");
```

# Ejercicios en Java

---

- Ejercicio\_1.
    - Calcular el cuadrado de distintos números hasta que se introduzca uno negativo.
  - Ejercicio\_2.
    - Números positivos o negativos hasta que se introduzca un cero.
  - Ejercicio\_3.
    - Indicar si los números son par o impar hasta que sea cero.
  - Ejercicio\_4.
    - Pedir números hasta que se introduzca un negativo.
-

# Ejercicios en Java.

---

- Ejercicios\_5.
    - Adivinar un número aleatorio.
  - Ejercicio\_6.
    - Suma iterativa.
  - Ejercicio\_7.
    - Calcular el promedio de N números.
  - Ejercicio\_8
    - Mostrar todos los números del 1 al N.
  - Ejercicio\_9.
    - Mostrar los números del 100 al 0 de 7 en 7.
  - Ejercicio\_10.
    - Calcular la suma de 10 números.
-