




# UT03. Bloque 2. Ejercicios: Trabajo con bucles y condicionales

---

## Guía de Ejercicios y Pistas de Resolución

### Ejercicio 1: Validador de Hora

Escribe un programa que lea por teclado tres números enteros (H, M, S) y compruebe si la hora que indican es válida en formato 24h.

-  **Pista 1: ¿Cómo leo los datos?**
  - Necesitarás crear un objeto **Scanner** para leer desde el teclado.
  - Pide al usuario que introduzca la hora, los minutos y los segundos. Guarda cada valor en una variable **int**.
-  **Pista 2: La lógica de validación**
  - Una hora es válida si **TODAS** sus partes son válidas.
  - La hora (H) debe estar entre 0 y 23
  - Los minutos (M) y los segundos (S) deben estar entre 0 y 59
-  **Pista 3: Estructura del código**
  - Un **if** principal puede agrupar todas las condiciones. Si se cumplen todas, la hora es "válida".
  - Si la condición del **if** no se cumple, el **else** te servirá para mostrar el mensaje "hora no válida".

### Solución

```
public static void main(String[] args) throws Exception {  
  
    int hora = 0, minuto = 0, segundos = 0;  
  
    Scanner teclado = new Scanner(System.in);  
  
    System.out.println("Introduzca la hora;");  
    hora = teclado.nextInt();  
    System.out.println("Introduzca el minuto;");  
    minuto = teclado.nextInt();  
    System.out.println("Introduzca el segundo;");  
    segundos = teclado.nextInt();  
  
    boolean esValida = true;  
  
    boolean esHoraValida = (hora >= 0 && hora <= 23);  
    boolean esMinutoValido = (minuto >= 0 && minuto <= 59);  
    boolean esSegundosValido = (segundos >= 0 && segundos <= 59);  
}
```

```
if ((hora >= 0 && hora <= 23)
    && (minuto >= 0 && minuto <= 59)
    && (minuto >= 0 && minuto <= 59)) {

    esValida = true;
} else {
    esValida = false;
}

if (esValida) {
    System.out.println("La hora es válida");
} else {
    System.out.println("La hora no es válida");
}

}
```

---

## Ejercicio 2: 📅 Días del Mes

Lee un número de mes (1-12) y muestra cuántos días tiene y su nombre. Febrero siempre tendrá 28 días.

- 💡 **Pista 1: Validar la entrada**
  - Lo primero es asegurarte de que el número introducido está entre 1 y 12. Un `if` inicial te servirá para esto. Si no lo está, muestra un mensaje de error y no continúes.
- 💡 **Pista 2: Agrupando los meses**
  - Hay meses con 31 días (1, 3, 5, 7, 8, 10, 12), meses con 30 días (4, 6, 9, 11) y uno con 28 (2).
  - La estructura `switch` es perfecta aquí. Puedes agrupar los `case` que tienen el mismo número de días para no repetir código.

```
// Ejemplo de agrupación en un switch
switch (mes) {
    case 1:
    case 3:
    case 5:
        // ...código para meses de 31 días
        System.out.println("Enero tiene 31 días");
        break; // No olvides el break!
    case 4:
    case 6:
        // ...código para meses de 30 días
        break;
}
```

---

## Solución

```
public static void main(String[] args) throws Exception {

    String mesNombre = "";
    int mesDias = 0;

    Scanner teclado = new Scanner(System.in);

    System.out.println("Indique el número del mes:");
    int mesNumero = teclado.nextInt();
    boolean mesValido = false;

    if (mesNumero >= 0 && mesNumero <= 12) {
        mesValido = true;
    } else {
        System.out.println("El mes no es válido");
    }

    if (mesValido) {

        switch (mesNumero) {
            case 1:
                mesNombre = "Enero";
                mesDias = 31;
                break;
            case 2:
                mesNombre = "Febrero";
                mesDias = 28;
                break;
            case 3:
                mesNombre = "Marzo";
                mesDias = 31;
                break;
            case 5:
                mesNombre = "Mayo";
            case 7:
                mesNombre = "Julio";
            case 8:
                mesNombre = "Agosto";
            case 10:
                mesNombre = "Octubre";
            case 12:
                mesNombre = "Diciembre";
                break;

            case 4:
                mesNombre = "Abril";
                mesDias = 30;
            case 6:
                mesNombre = "Junio";
                mesDias = 30;
            case 9:
                mesNombre = "Septiembre";
                mesDias = 30;
```

```
        case 11:
            mesNombre = "Noviembre";
            mesDias = 30;
            break;

        default:
            System.out.println("Este mes no existe");
            break;
    }

    System.out.printf("El mes %s y número %d tiene %d días",
        mesNombre, mesNumero, mesDias);

    } else {
        System.out.printf("El mes número %d no existe", mesNumero);
    }
}
```

---

### Ejercicio 3: 🎓 Nota Final del Curso

Pide las notas de tres evaluaciones. Si todas están aprobadas ( $\geq 5$ ), calcula la media. Si no, indica qué evaluación/es se deben recuperar.

- 💡 **Pista 1: Comprobación inicial**
  - Primero, lee las tres notas y guárdalas en variables (por ejemplo, `eval1`, `eval2`, `eval3`).
  - Usa un `if` para comprobar la condición principal: ¿Están **todas** aprobadas?.
- 💡 **Pista 2: Dos caminos diferentes**
  - Si se cumple el `if`:** Entra en el bloque y calcula la media.
  - Si no se cumple (bloque `else`):** Significa que al menos una está suspensa. Ahora necesitas comprobar una por una cuál de ellas es la que hay que recuperar.

---

### Solución

```
public static void main(String[] args) throws Exception {

    int eval1 = 0, eval2 = 0, eval3 = 0;
    String evalRecuperar = "";
    double notaMedia = 0;

    Scanner teclado = new Scanner(System.in);

    for (int i = 1; i <= 3; i++) {

        System.out.println("Introduzca la nota:");
```

```
switch (i) {
    case 1:
        eval1 = teclado.nextInt();
        if (eval1 < 5) {
            evalRecuperar = "Primera Evaluación";
        }
        break;
    case 2:
        eval2 = teclado.nextInt();
        if (eval2 < 5) {
            evalRecuperar = "Segunda Evaluación";
        }
        break;
    case 3:
        eval3 = teclado.nextInt();
        if (eval3 < 5) {
            evalRecuperar = "Tercera Evaluación";
        }
        break;
    default:
        break;
}

if (evalRecuperar != "") {
    break;
}

}

if (evalRecuperar == "") {
    // No hay ninguna suspensa
    notaMedia = (eval1 + eval2 + eval3) / 3;
    System.out.printf("La nota media de las evaluaciones es: %.2f",
notaMedia);

} else {
    // Hay que recuperar alguna
    System.out.printf("Debe recuperar la %s", evalRecuperar);
}


}
```

---

#### Ejercicio 4: 💰 Cálculo de Sueldo

Calcula el sueldo de un trabajador pidiendo el número de horas. Las primeras 150h se pagan a 10€ y el resto a 20€.

- 💡 **Pista 1: ¿Hay horas extra?**
  - La clave es determinar si el trabajador ha superado las 150 horas.

-  **Pista 2: Los dos escenarios de cálculo**
  - El cálculo se hace en dos partes, dependiendo de si hay horas extra o no.

---

## Solución

```
public static void main(String[] args) throws Exception {

    Scanner teclado = new Scanner(System.in);

    System.out.println("Indique el número de horas trabajadas:");
    int numHoras = teclado.nextInt();
    int horasBase = numHoras;
    int horasExtra = 0;

    if (horasBase > 150) {
        horasExtra = horasBase - 150;
        horasBase = horasBase - 150;
    }

    double sueldoBase = horasBase * 10;
    double sueldoExtra = horasExtra * 20;

    double sueldo = sueldoBase + sueldoExtra;


    System.out.printf("El sueldo es %.2f€ para el número de horas: %d",
        sueldo, numHoras);

}
```

---

## Ejercicio 5: Número a Letra

Pide un número del 0 al 10 y muéstralo en pantalla escrito con letras. Si el número no está en ese rango, "Número no válido".

-  **Pista 1: La herramienta perfecta**
  - Este problema es el ejemplo perfecto para usar una estructura `switch`. Evalúa la variable numérica que introduce el usuario.

---

## Solución

```
public static void main(String[] args) {

    Scanner teclado = new Scanner(System.in);

    System.out.println("--- Ejercicio 5: Número a Letra ---");
    System.out.print("Introduce un número entero del 0 al 10: ");
```

```
int numero = teclado.nextInt();

String nombreEnLetras;

// 5. Usar la estructura switch para asignar el nombre
switch (numero) {
    case 0:
        nombreEnLetras = "Cero";
        break;
    case 1:
        nombreEnLetras = "Uno";
        break;
    case 2:
        nombreEnLetras = "Dos";
        break;
    case 3:
        nombreEnLetras = "Tres";
        break;
    case 4:
        nombreEnLetras = "Cuatro";
        break;
    case 5:
        nombreEnLetras = "Cinco";
        break;
    case 6:
        nombreEnLetras = "Seis";
        break;
    case 7:
        nombreEnLetras = "Siete";
        break;
    case 8:
        nombreEnLetras = "Ocho";
        break;
    case 9:
        nombreEnLetras = "Nueve";
        break;
    case 10:
        nombreEnLetras = "Diez";
        break;
    default:
        // Bloque 'default' para cualquier número fuera del rango
        nombreEnLetras = "Número no válido";
        break;
}

// 6. Mostrar el resultado por pantalla
System.out.println("El número que introdujiste es: " +
nombreEnLetras + ".");

// 7. Cerrar el objeto Scanner
teclado.close();
}
```

## Ejercicio 6: Letra a Número

Pide un número en letras ("cero", "uno",...) y muestra el número correspondiente en cifra.

- 💡 **Pista 1: Leyendo texto**
  - Para leer una palabra del teclado, necesitarás usar `teclado.next()` en lugar de `teclado.nextInt()`. El resultado se guarda en una variable de tipo `String`.
- 💡 **Pista 2: Comparando texto**
  - El `switch` también funciona con `String` en Java. ¡Es la forma más limpia de resolverlo!
  - Para que no importe si el usuario escribe "uno", "Uno" o "UNO", puedes convertir la entrada a minúsculas antes de compararla, usando el método `.toLowerCase()`. Ejemplo: `String texto = teclado.next().toLowerCase();`

### Solución

## Ejercicio 7: Comprobador de Año Bisiesto

Pide un año y di si es bisiesto. No se aceptan años negativos.

- 💡 **Pista 1: Búsqueda y validación**
  - Primero, valida que el año no sea negativo. `if (anio < 0)`.
  - Luego, busca la regla para los años bisiestos. Te la resumo aquí:
    - Un año es bisiesto si es divisible por 4.
    - Excepción:** Si es divisible por 100, entonces **no** es bisiesto.
    - Excepción a la excepción:** A menos que también sea divisible por 400. En ese caso, **sí** es bisiesto.

### Solución

```
public static void main(String[] args) throws Exception {  
  
    Scanner teclado = new Scanner(System.in);  
    int año = teclado.nextInt();  
  
    boolean esBisiesto = false;  
  
    if (año % 100 != 0 && año % 400 == 0) {  
        esBisiesto = true;  
    }  
}
```



```
    } else if (año % 100 != 0) {  
        esBisiesto = false;  
    } else if (año % 4 == 0) {  
        esBisiesto = true;  
    } else {  
        esBisiesto = false;  
    }  
  
    if (esBisiesto)  
        System.out.println("El año %d es BISIESTO");  
    else  
        System.out.println("El año %d es NO ES BISIESTO");  
  
}
```