

Descripción de la actividad. Generar un programa que retorne por consola el número del mes a partir del nombre del mes que se pasa como parámetro

Objetivos de la actividad: Se espera que puedan desarrollar la solución usando los tipos de condicionales explicados y comparar ambas soluciones para ver pros y contras

las condicionales son if y switch

GPT-4

A continuación, te proporcionaré dos versiones de un programa en Java que recibe el nombre de un mes como parámetro y devuelve el número del mes correspondiente. La primera versión utilizará la estructura condicional `if`, y la segunda versión utilizará `switch`.

Versión con `if`:

JAVA

```
public class MesANumeroIf {  
    public static void main(String[] args) {  
        String mes = "Enero"; // Aquí se puede cambiar el  
        mes para probar con diferentes meses  
        int numeroMes = 0;  
  
        if (mes.equalsIgnoreCase("Enero")) {  
            numeroMes = 1;  
        } else if (mes.equalsIgnoreCase("Febrero")) {  
            numeroMes = 2;  
        } else if (mes.equalsIgnoreCase("Marzo")) {  
            numeroMes = 3;  
        } else if (mes.equalsIgnoreCase("Abril")) {  
            numeroMes = 4;  
        } else if (mes.equalsIgnoreCase("Mayo")) {  
            numeroMes = 5;  
        } else if (mes.equalsIgnoreCase("Junio")) {  
            numeroMes = 6;  
        } else if (mes.equalsIgnoreCase("Julio")) {  
            numeroMes = 7;  
        } else if (mes.equalsIgnoreCase("Agosto")) {  
            numeroMes = 8;  
        } else if (mes.equalsIgnoreCase("Septiembre")) {  
            numeroMes = 9;  
        } else if (mes.equalsIgnoreCase("Octubre")) {  
            numeroMes = 10;  
        } else if (mes.equalsIgnoreCase("Noviembre")) {  
            numeroMes = 11;  
        }  
    }  
}
```

```

        } else if (mes.equalsIgnoreCase("Diciembre")) {
            numeroMes = 12;
        } else {
            System.out.println("Mes no válido.");
            return;
        }
    }

    System.out.println("El número del mes " + mes + "
es: " + numeroMes);
}
}

```

Versión con switch:
JAVA

```

public class MesANumeroSwitch {
    public static void main(String[] args) {
        String mes = "Enero"; // Aquí se puede cambiar el
mes para probar con diferentes meses
        int numeroMes;

        switch (mes.toLowerCase()) {
            case "enero":
                numeroMes = 1;
                break;
            case "febrero":
                numeroMes = 2;
                break;
            case "marzo":
                numeroMes = 3;
                break;
            case "abril":
                numeroMes = 4;
                break;
            case "mayo":
                numeroMes = 5;
                break;
            case "junio":
                numeroMes = 6;
                break;
            case "julio":
                numeroMes = 7;
                break;

```

```

        case "agosto":
            numeroMes = 8;
            break;
        case "septiembre":
            numeroMes = 9;
            break;
        case "octubre":
            numeroMes = 10;
            break;
        case "noviembre":
            numeroMes = 11;
            break;
        case "diciembre":
            numeroMes = 12;
            break;
        default:
            System.out.println("Mes no válido.");
            return;
    }

    System.out.println("El número del mes " + mes + "
es: " + numeroMes);
}
}

```

Comparación de `if` vs `switch`:

Pros de `if`:

- Flexibilidad: `if` puede manejar rangos de valores y condiciones más complejas que no se limitan a igualdades.
- Legibilidad: para pocas condiciones, `if` puede ser más legible y claro.

Contras de `if`:

- Verbosidad: para muchos valores, como en el caso de los meses, puede resultar más verboso y menos claro que `switch`.
- Rendimiento: en algunos casos, `if` puede ser ligeramente menos eficiente que `switch`, especialmente con un gran número de condiciones.

Pros de `switch`:

- Claridad: `switch` puede ser más claro y directo para comparaciones basadas en un solo valor con múltiples casos.
- Rendimiento: `switch` puede ser más eficiente que `if` en escenarios con muchos casos, ya que no necesita evaluar todas las condiciones en secuencia.

Contras de `switch`:

- Limitación: `switch` solo funciona con tipos compatibles y comparaciones de igualdad. No puede manejar rangos o condiciones más complejas.
- Fallthrough: si se olvida incluir un `break`, se puede producir una ejecución no deseada de múltiples casos (aunque esto puede ser también una ventaja si se usa intencionadamente).

Ambas estructuras tienen sus usos y dependiendo del contexto una puede ser más adecuada que la otra. En el caso de los meses del año, `switch` es probablemente la opción más limpia y eficiente.

Improve this agentReset Chat