

### Comparación entre los dos tipos de programación

	Programación Tradicional	Programación Orientada a Objetos
<b>Estructura del Código</b>	<ul style="list-style-type: none"> <li>• Utiliza funciones globales para las operaciones.</li> <li>• ingresar_temperaturas y calcular_promedio son funciones independientes.</li> <li>• La función main coordina la ejecución llamando a estas funciones.</li> </ul>	<ul style="list-style-type: none"> <li>• Utiliza una clase (ClimaDiario) para encapsular los datos y las operaciones.</li> <li>• Los métodos ingresar_temperaturas, calcular_promedio y mostrar_promedio están encapsulados dentro de la clase.</li> <li>• La función main crea una instancia de la clase y llama a sus métodos.</li> </ul>
<b>Encapsulamiento</b>	<ul style="list-style-type: none"> <li>• Las variables y funciones están en el ámbito global.</li> <li>• No hay encapsulamiento explícito, lo que puede llevar a conflictos de nombres y dificultades en la gestión de estados.</li> </ul>	<ul style="list-style-type: none"> <li>• Los datos (_temperaturas) están encapsulados dentro de la clase.</li> <li>• Los métodos operan sobre los datos encapsulados, proporcionando una interfaz clara para interactuar con ellos.</li> <li>• Mejora la modularidad y facilita el mantenimiento y la expansión del código.</li> </ul>
<b>Reutilización y Extensibilidad</b>	<ul style="list-style-type: none"> <li>• Las funciones son reutilizables, pero la organización del código puede dificultar la extensión.</li> <li>• Añadir nuevas funcionalidades o modificar el comportamiento puede requerir cambios significativos en varias partes del código.</li> </ul>	<ul style="list-style-type: none"> <li>• La clase ClimaDiario puede ser fácilmente extendida o reutilizada en otros contextos.</li> <li>• Añadir nuevos métodos o modificar los existentes es más sencillo y está encapsulado.</li> <li>• Facilita la herencia y el polimorfismo, permitiendo crear subclases con comportamientos específicos.</li> </ul>
<b>Manejo de Errores y Validaciones</b>	<ul style="list-style-type: none"> <li>• La validación de entrada se realiza dentro de la función ingresar_temperaturas.</li> <li>• Los errores se manejan dentro de cada función de manera individual.</li> </ul>	<ul style="list-style-type: none"> <li>• La validación de entrada y el manejo de errores están encapsulados en los métodos de la clase.</li> <li>• Permite un manejo de errores más estructurado y centralizado.</li> </ul>