**Análisis del Programa**

Tareas Repetitivas o que pueden encapsularse en funciones

Agregar estudiantes, Mostrar la lista de estudiantes, Calcular promedios, Encontrar al estudiante

**Variables Locales vs Variables Globales**

El uso adecuado de variables locales y globales ayuda a mejorar la seguridad y claridad del código.

Variables Globales: List<string> estudiantes, List<double> calificaciones

Variables Locales: nombre, calificacion, suma, maxCalificacion, estudianteMax, etc.

**Definir Variables Locales y Globales**

**Variables que necesitan ser compartidas entre múltiples funciones:**

estudiantes y calificaciones deben ser globales, ya que contienen la información que se utiliza en diversas funciones del programa.

**Variables relevantes solo dentro de una función:**

nombre y calificacion dentro de AgregarEstudiante(). suma y promedio dentro de CalcularPromedio(). maxCalificacion y estudianteMax dentro de ObtenerEstudianteConMayorCalificacion().

**Preguntas Guía**

**1. ¿Qué ventajas tiene dividir el código en funciones?**

Organización: Se mejora la estructura del código, haciéndolo más fácil de entender y mantener.

Reutilización: Se pueden llamar las funciones en distintos puntos del programa sin repetir código.

Mantenimiento: Facilita la modificación del código, ya que los cambios en una función no afectan al resto del programa.

**2. ¿Por qué es importante limitar el uso de variables globales?**

Evitar errores inesperados: Si una variable global se modifica accidentalmente en una función, puede afectar otras partes del programa.

Depuración más sencilla: Es más fácil rastrear errores cuando las variables solo existen dentro de funciones específicas.

Mejor seguridad y encapsulamiento: Se evita la modificación involuntaria de datos por funciones que no deberían tener acceso a ellos.

**3. ¿Cómo se puede mejorar la legibilidad del código?**

Usar nombres descriptivos: Las funciones y variables deben tener nombres claros y representativos de su función.

Agregar comentarios: Explicar el propósito de cada función o bloque de código ayuda a otros programadores (o a uno mismo en el futuro) a entender mejor el programa.

Estructura clara: Mantener una organización adecuada con sangrías, espacios y separación de bloques de código facilita la lectura y comprensión.