

Administrador Impresora

Este ejemplo ilustra cómo usar el patrón Singleton para asegurar que solo haya una única instancia de la clase PrinterManager en toda la aplicación, permitiendo que la gestión de la impresora sea accesible y consistente desde cualquier parte del programa.

Clase PrinterManager

AdminImpresora: tiene una variable estática privada instancia que almacena la única instancia de AdminImpresora.

Constructor Privado: El constructor de AdminImpresora es privado para evitar la creación de instancias desde fuera de la clase. Aquí, se inicializa el estado de la impresora.

Método getInstance(): Este método devuelve la única instancia de AdminImpresora. Si la instancia aún no existe, se crea.

Método estaLista(): Devuelve el estado de la impresora, indicando si está lista para imprimir.

Método imprimirDocumento(): Simula la impresión de un documento. Si la impresora está lista, imprime el documento y cambia el estado a no listo.

Método restablecerImpresora(): Restablece el estado de la impresora a lista.

Clase Principal

Demostración del Singleton: La clase muestra cómo usar el AdminImpresora Singleton para verificar el estado de la impresora, imprimir documentos y restablecer la impresora, y verifica que solo existe una instancia de AdminImpresora.