**Temario**

**1. Clases y POO**

* **Constructores y propiedades:** Asegúrate de saber cómo crear y usar propiedades (get y set) y constructores para inicializar objetos. Esto será clave para representar tus datos (como socios, turnos, etc.) en la aplicación.
* **Encapsulación y convenciones:** Practica cómo proteger datos sensibles y exponer solo lo necesario.
* **Herencia y polimorfismo:** Úsalos para abstraer comportamientos comunes entre tus entidades (por ejemplo, si tienes diferentes tipos de usuarios, como administradores y clientes).

**2. Colecciones**

* **List, LinkedList, Queue, Dictionary:** Estas estructuras son fundamentales para manejar datos dinámicamente. Por ejemplo, usarás listas para almacenar los registros de tu base de datos en memoria temporalmente.
* **Stach y colecciones avanzadas:** Aunque no son tan críticas para CRUDs, entenderlas amplía tus opciones para resolver problemas.

**3. Delegados y eventos**

* **Delegados y predicados:** Practica cómo aplicar lógica personalizada (como filtrar elementos en una lista) usando predicados.
* **Eventos:** Son importantes para manejar interacciones en la interfaz gráfica, como hacer algo cuando el usuario hace clic en un botón.

**4. Expresiones regulares**

* Esto será útil si necesitas validar datos (como correos electrónicos, números de teléfono o formatos específicos) antes de guardarlos en la base de datos.

**5. Manejo de excepciones**

* Asegúrate de manejar correctamente los errores para que tu aplicación no falle cuando algo inesperado suceda (por ejemplo, una conexión fallida a la base de datos).

**6. SQL y conexión con bases de datos**

* Si bien esto no está explícitamente en la lista, relacionarlo con tus conocimientos de C# es crucial. Aprende cómo usar **ADO.NET** o **Entity Framework** para realizar operaciones CRUD.

**7. Métodos y sobrecarga**

* Asegúrate de entender cómo crear métodos reutilizables y cómo usar sobrecarga para facilitar el manejo de diferentes parámetros.