**🧠 MAPA MENTAL – ESTUDAR C# (check list [ ])**

**🌱 1. Fundamentos da Linguagem**

* **Sintaxe básica**
  + Declaração de variáveis **[✅]**
  + Tipos de dados primitivos **[✅]**
  + Operadores (aritméticos, lógicos, relacionais) **[✅]**
* **Estruturas de controle**
  + Condicionais: if, else, switch
  + Laços: for, while, do while, foreach
* **Funções e métodos**
  + Definição e chamada de métodos
  + Parâmetros (ref, out, default)
  + Retorno e sobrecarga
* **Manipulação de strings**
  + Métodos da classe String
  + Interpolação e concatenação
* **Coleções básicas**
  + Arrays
  + Listas (List<T>)
  + Dicionários (Dictionary<TKey, TValue>)
  + Filas, Pilhas (Queue, Stack)

**🧱 2. Programação Orientada a Objetos (POO)**

* **Classes e objetos**
  + Instanciação
  + Construtores
* **Encapsulamento**
  + Modificadores de acesso (public, private, protected, internal)
  + Propriedades (get, set)
* **Herança**
  + Uso de base, herança simples
* **Polimorfismo**
  + Override, overload
  + Métodos virtuais e abstratos
* **Abstração**
  + Classes abstratas
  + Interfaces
* **Interfaces**
  + Implementação múltipla
  + Boas práticas
* **Modificadores especiais**
  + static, sealed, readonly, const

**🧰 3. Recursos Avançados da Linguagem**

* **Delegates e eventos**
  + Delegates simples e genéricos
  + Eventos personalizados
* **Lambda expressions**
* **LINQ (Language Integrated Query)**
  + Select, Where, GroupBy, Aggregate, etc.
* **Generics**
  + Métodos e classes genéricas
* **Extension methods**
* **Async/Await e programação assíncrona**
  + Task, async, await
* **Tratamento de exceções**
  + try, catch, finally, throw
  + Exceções personalizadas

**📁 4. Manipulação de Dados**

* **Trabalhando com arquivos**
  + System.IO, File, StreamReader, StreamWriter
* **Serialização e Desserialização**
  + JSON: System.Text.Json, Newtonsoft.Json
  + XML
* **Banco de dados**
  + ADO.NET
  + Entity Framework Core
    - Migrations
    - LINQ com banco
    - DbContext, DbSet

**🌐 5. Desenvolvimento Web com C#**

* **ASP.NET Core**
  + MVC (Model-View-Controller)
  + Razor Pages
  + Rotas, middlewares, controllers
* **APIs RESTful**
  + Criação de endpoints
  + JSON, HttpClient
  + Documentação com Swagger
* **Autenticação e Autorização**
  + ASP.NET Identity
  + JWT Tokens

**🖥️ 6. Aplicações Desktop e Mobile**

* **WinForms** (básico, sistemas legados)
* **WPF (Windows Presentation Foundation)**
  + XAML, bindings, MVVM
* **.NET MAUI** (multiplataforma)
  + Layouts, navegação, data binding
  + Integração com APIs
  + Armazenamento local

**🔐 7. Segurança em C#**

* **Criptografia**
  + MD5, SHA256, AES, RSA
* **Validação de entrada**
* **Proteção de APIs**
  + Autenticação via token
  + CORS
* **Boas práticas de segurança**
  + SQL Injection, XSS, CSRF

**🧪 8. Testes de Software**

* **Testes unitários**
  + xUnit, NUnit, MSTest
* **Testes de integração**
* **Mocking**
  + Moq, FakeItEasy
* **Cobertura de testes**
  + coverlet, Visual Studio Test Coverage

**🧱 9. Arquitetura e Boas Práticas**

* **Padrões SOLID**
* **Princípios de código limpo (Clean Code)**
* **Design Patterns**
  + Singleton, Factory, Repository, Adapter, Strategy, etc.
* **Injeção de Dependência**
  + Nativa no ASP.NET Core
* **Arquitetura em camadas**
  + Apresentação, Negócios, Dados
* **Clean Architecture e DDD (Domain Driven Design)**

**⚙️ 10. Ferramentas e Ambiente de Desenvolvimento**

* **IDEs e editores**
  + Visual Studio, Visual Studio Code
* **Controle de versão**
  + Git, GitHub, GitLab
* **Pacotes e dependências**
  + NuGet
* **Docker**
  + Containerização de apps .NET
* **DevOps**
  + CI/CD com GitHub Actions, Azure DevOps

**📚 11. Recursos Complementares**

* **Documentação oficial (Microsoft Docs)**
* **Cursos online**
  + Microsoft Learn, Udemy, Alura, Pluralsight
* **Projetos práticos**
  + CRUD completo com banco
  + API com autenticação
  + Sistema de login, dashboard, loja virtual
* **Comunidades**
  + Stack Overflow, Reddit, Dev.to, Discord, LinkedIn