



AVALIAÇÃO INDIVIDUAL	
MÓDULO	.NB – Formação .Net - Básico
OBJETIVO DA ATIVIDADE	TEMPO
Testar o conhecimento sobre os conceitos de Orientação a Objetos e recursos básicos de C# aplicados a um problema prático.	2H
DESCRIÇÃO	
<p>Crie uma Branch para esta atividade.</p> <ul style="list-style-type: none">• Faça commits, no mínimo, a cada 15 minutos desde o início do desenvolvimento.• No seu repositório no GitHub para a disciplina de .NET, crie uma Branch nova para essa atividade.• Publique, no Moodle, o endereço do repositório e a BRANCH em que está desenvolveu sua atividade. <p>Escopo: Você está se aplicando a um processo seletivo para estágio em desenvolvimento em C# na empresa POG Dev. Como desafio o recrutador pediu para você propor uma solução, em 2 horas, sobre o problema abaixo:</p> <p>Problema: Desenvolver as classes iniciais para um sistema de gerenciamento de escritório de advocacia e gerar alguns relatórios. Fique livre para criar classes acessórias se julgar necessário. Por exemplo, para aplicar conceitos de Herança e Polimorfismo, gerar os relatórios e aplicar as regras de negócio.</p> <p>Classes</p> <ul style="list-style-type: none">• Advogado:<ul style="list-style-type: none">○ Nome○ Data de Nascimento○ CPF (tratar entrada com 11 dígitos)○ CNA• Cliente:<ul style="list-style-type: none">○ Nome○ Data de Nascimento○ CPF (tratar entrada com 11 dígitos)○ Estado civil (solteiro ou casado)○ Profissão <p>Requisitos</p> <ul style="list-style-type: none">• Criar uma coleção de Advogados, validando CPF e CNA únicos.• Criar uma coleção de Clientes, validando CPF único.	



Relatórios:

1. Advogados com idade entre dois valores
2. Clientes com idade entre dois valores
3. Clientes com estado civil informado pelo usuário
4. Clientes em ordem alfabética
5. Clientes cuja profissão contenha texto informado pelo usuário
6. Advogados e Clientes aniversariantes do mês informado

Dicas:

- Utilize expressões lambda para consultas.
- Implemente consultas LINQ para gerar os relatórios.
- Utilize tratamento de exceções para garantir uma experiência de usuário mais robusta e amigável.

C

- <https://learn.microsoft.com/pt-br/dotnet/csharp/language-reference/builtin-types/value-tuples>
- <https://learn.microsoft.com/pt-br/dotnet/csharp/language-reference/operators/lambda-expressions>
- <https://learn.microsoft.com/pt-br/dotnet/csharp/programming-guide/concepts/linq/introduction-to-linq-queries>
- <https://learn.microsoft.com/pt-br/dotnet/csharp/fundamentals/exceptions/>