**3 áreas em que o C# é amplamente usado**

**Desenvolvimento de jogos:** Com a engine Unity, C# é a principal linguagem para criar jogos 2D e 3D.

**Aplicações desktop (Windows):** Usado com Windows Forms e WPF para criar softwares com interface gráfica.

**Desenvolvimento web (back-end):** Muito usado com ASP.NET para criar APIs e sites dinâmicos.

**Diferenças entre C# e Java**

1. **Plataforma:** C# foi criado pela Microsoft para rodar no ambiente .NET. Já o Java é multiplataforma, rodando em qualquer sistema que tenha a JVM (Java Virtual Machine).
2. **Sintaxe e recursos modernos:** C# tem uma sintaxe mais moderna e rica. Já o Java é mais conservador, o que garante estabilidade, mas algumas funcionalidades demoram mais a ser adicionadas.
3. **Desenvolvimento de aplicações desktop e mobile:** C# é muito usado para criar programas de desktop no Windows, com ferramentas como Windows Forms, WPF e MAUI. Também é usado em mobile com Xamarin. Java, por outro lado, é a principal linguagem do Android e tem menos foco em desktop.

**Resumo final:**

C# é mais moderno e integrado ao ecossistema Microsoft.

Java é mais portátil, amplamente usado em Android e empresas.

**O que Python tem em comum com C#**

1. **Orientação a objetos:** Ambos suportam classes, herança, etc.
2. **Gerenciamento automático de memória:** Usam garbage collection.
3. **Sintaxe legível:** Embora Python seja mais conciso, C# também preza pela clareza.
4. **Multiplataforma (atualmente):** C# com .NET Core e Python rodam em vários sistemas operacionais.

**Pesquisa da Atividade 2:** Sim, todo programa C# precisa de um ponto de entrada que tradicionalmente é o Main. Se você não puder usar o Main, basta usar o recurso de top-level statements, disponível a partir do C# 9.0 (com .NET 5 ou superior).

**Pesquisa da Atividade 3:** Não, temos muitos tipos de variáveis como: char, object, decimal e var.

**Pesquisa da Atividade 4:** Sim, é possível não usar chaves em C#, mas só em blocos que têm apenas uma única instrução. Como if, for, while.

**Pergunta 2:** Para conversão é normal usarmos o Convert e o Parse. O Convert fornece métodos seguros para converter entre quaisquer tipos, e o Parse converte exclusivamente strings para tipos primitivos. Para testar tipos temos o is e o typeof. O is verifica se uma variável é de determinado tipo (Como no python). E o typeof verifica se uma variável é de determinado tipo.

**Pesquisa da Atividade 6:** Sim, você não pode omitir as chaves em uma função completa.

**Exemplo:**

static void Bernardo\_vcs\_ja\_sabem()

Console.WriteLine("Olá");

Isso geraria erro porque funções sempre precisam de chaves para delimitar o corpo do código no geral.

**Pesquisa da Atividade 7:** A diferença entre for e foreach está principalmente em como você itera sobre coleções e quanto controle você tem sobre o índice. foreach é mais seguro e claro, for tem mais controle sobre a iteração.

**Pergunta 2:** O tipo list em Python equivale a List<T> em C#, o tipo tuple em Python equivale a tuplas Tuple<T1, T2> , e o tipo dict em Python equivale a Dictionary<TKey, TValue>.

**Exemplos:**

List<int> lista = new List<int> {1, 2, 3};

var tupla = (Id: 1, Nome: "Ana");

Dictionary<string, int> idade = new Dictionary<string, int> { {"idade", 18} };