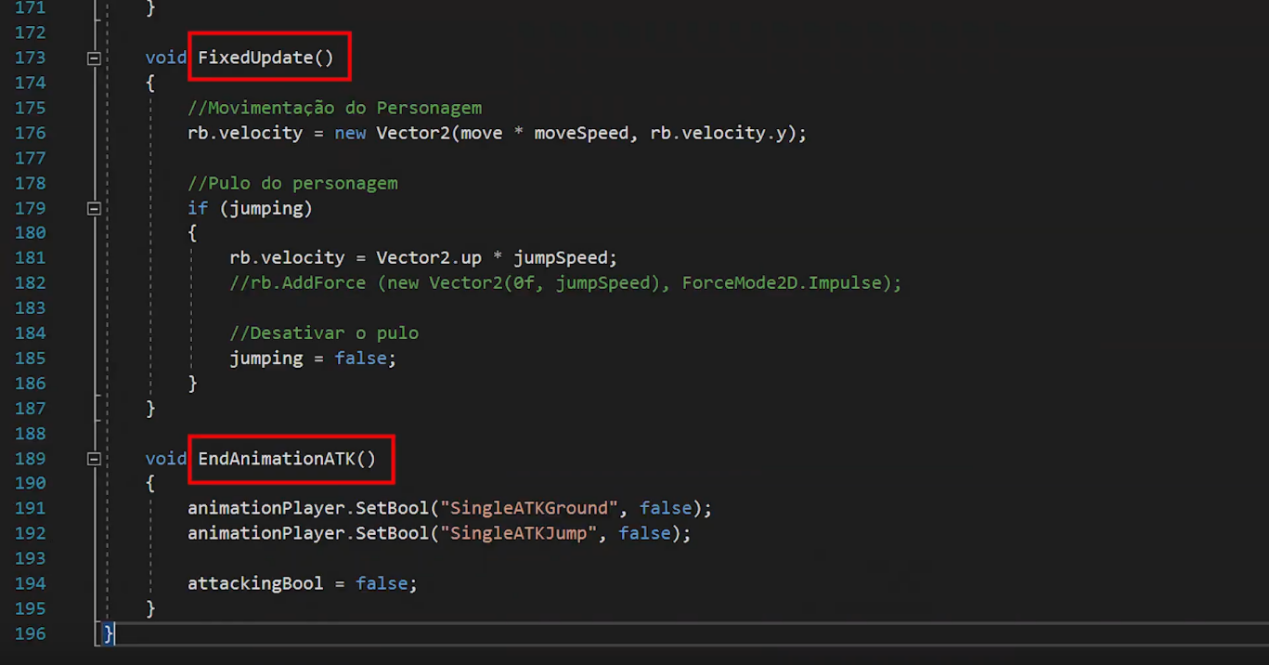
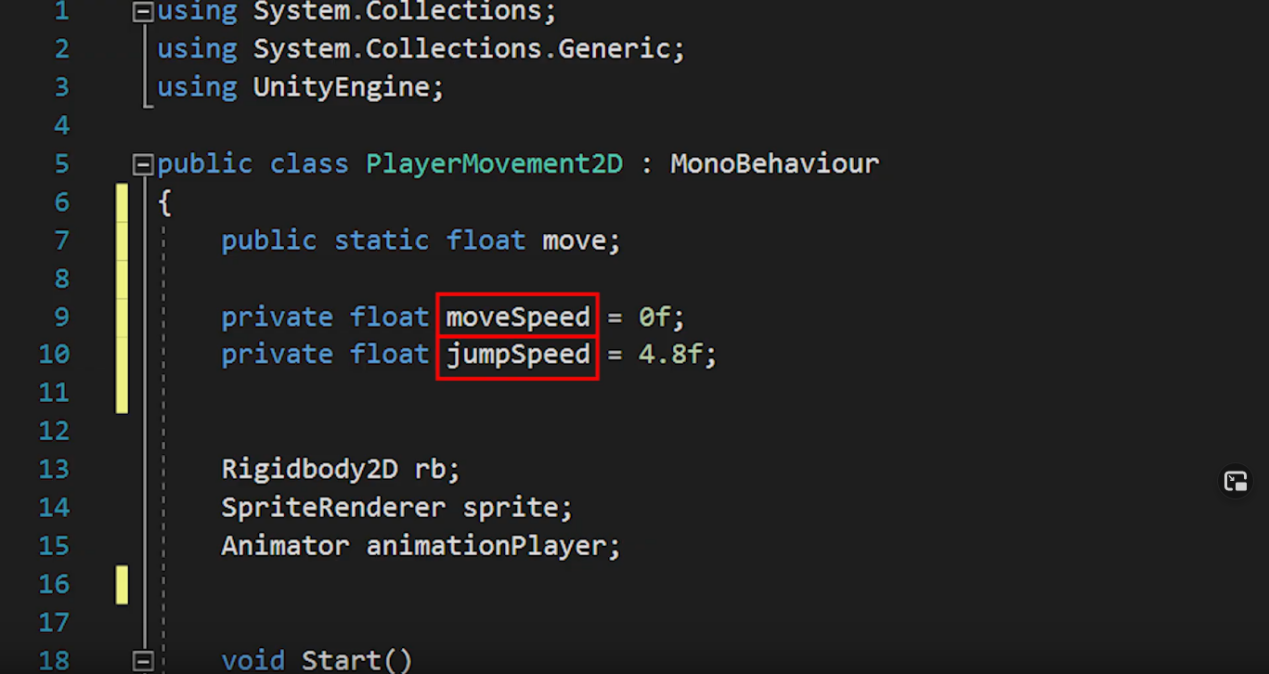
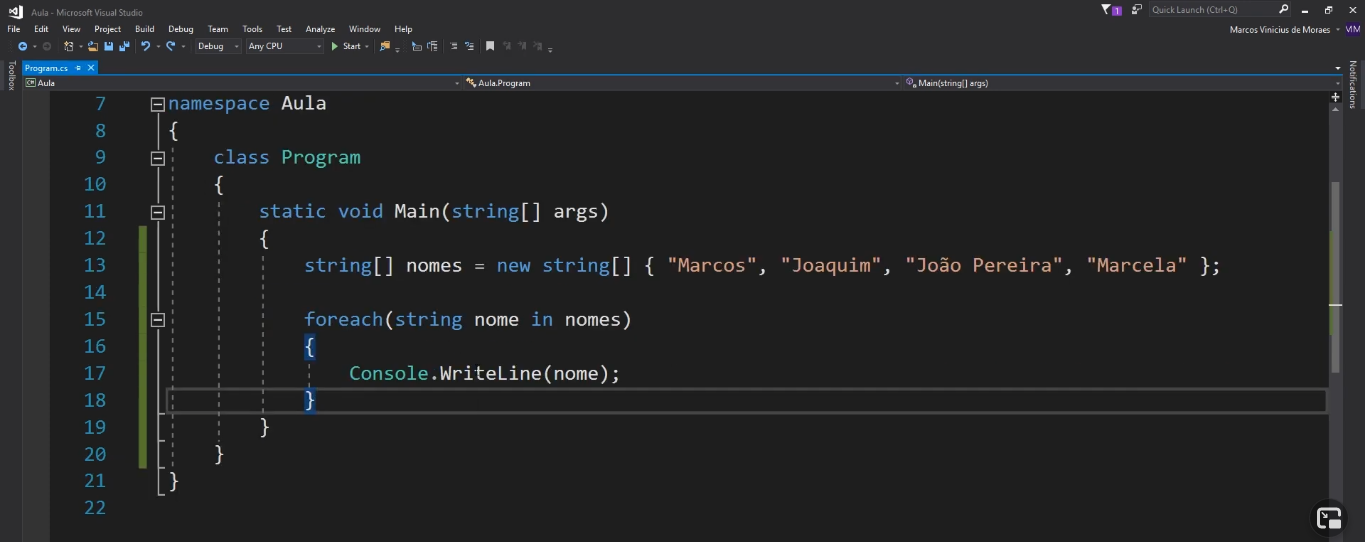
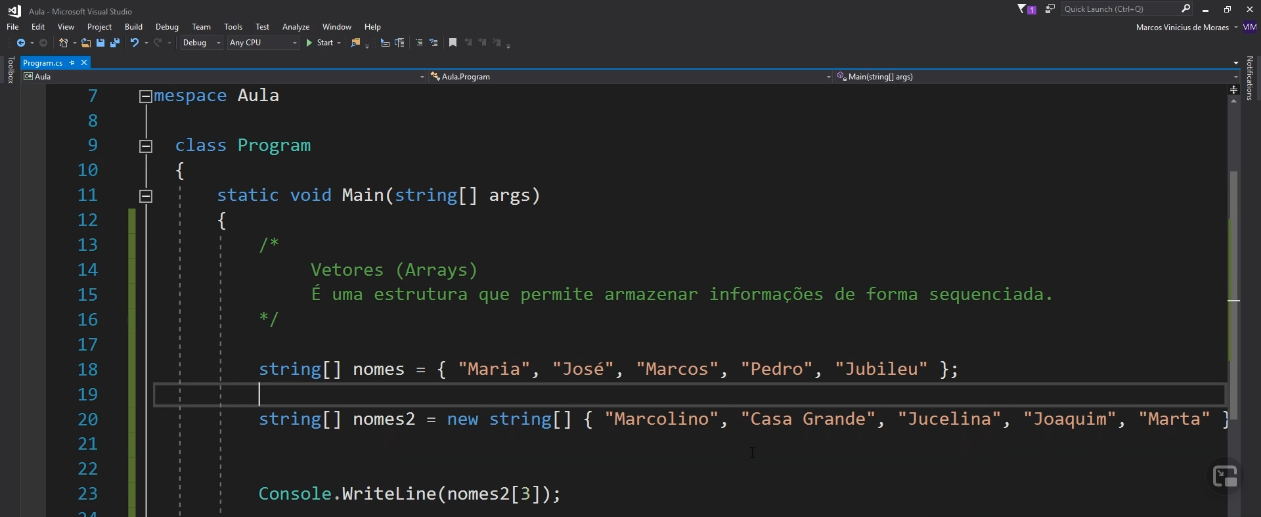
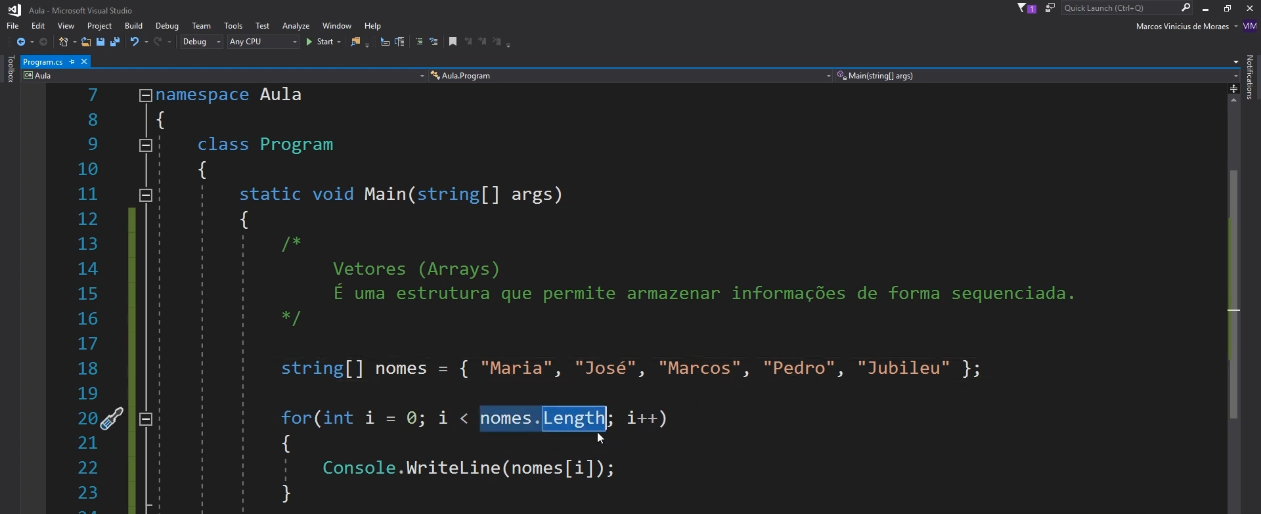
1. Lógica de programação = conjunto de regras coerentes, lógicas e detalhadas passadas para o computador. Chamamos esse conjunto de algoritmo.
2. Constantes = valores fixos durante a execução do programa
3. Variáveis = valores que precisam variar durante a execução do programa
   1. Ex: “Digite um número: ” (é variável)
4. UpperCamelCase = palavras escritas juntas em que a primeira letra é maiúscula. Ajuda na leitura do código. Indicado para classes e métodos (em vermelho):
   1. 
5. lowerCamelCase = a diferença é que a primeira palavra começa com letra minúscula. Indicada para variáveis, atributos e métodos (em vermelho):
   1. 
6. Tipos de dados:
   1. Numérico: armazenamento de números que podem ser usados para cálculos futuros.
   2. Caractere: sequência de caracteres que contém apenas letras.
   3. Alfanumérico: sequência de letras, números, dígitos/símbolos especiais. Aparecem entre aspas.
   4. Lógico: verdadeiro ou falso. Sim ou não. 1 ou 0.
7. Operadores aritméticos = 4 operações básicas da matemática
   1. (=, -, \*, /)
   2. Resto (ou mod) é %. Esse é o resto de uma divisão. Ex: 19/3 tem resto 1. O cálculo, no programa, ficaria 19%3 e o resultado apareceria 1.
   3. Potência: **Match.Pow(**5**,**9**)** = é o mesmo que 59.
8. Operadores relacionais são utilizados para comparar valores. O resultado é Verdadeiro ou Falso (bool).
   1. Igual: **==**
   2. Diferente: **!=**
   3. Menor: **<**
   4. Maior: **>**
   5. Menor ou igual: **<=**
   6. Maior ou igual: **>=**
9. Estrutura de decisão
   1. Simples (**if**)
   2. Composta (**if-else**)
   3. Para muitas alternativas = **switch/case**
10. Estrutura de repetição (laço)
    1. Quando sabemos a quantidade de vezes que precisamos mostrar a informação = **for**
    2. Quando a condição é testada logo no início do programa = **while**. Ele executa se a condição for verdadeira.
    3. Quando a condição é testada no final do programa = **do while**.
       1. A principal diferença é que o **do while** executa pelo menos uma vez, mesmo que a condição não seja verdadeira. Já o **while**, se a condição não for verdadeira logo no início, nem testa.
    4. Recompendado para trabalhar com vetores e matrizes = **foreach.**
       1. 
11. Vetores (**arrays** simples) = estrutura que permite armazenar informações de forma sequenciada.
    1. 
       1. Ele vai mostrar em tela o nome 3 da lista nomes2. No caso, o nome 3 é Joaquim, porque a lista começa em 0.
    2. 
       1. O **for** vai mostrar todos os nomes enquanto o **i** for menor do que o comprimento de “nomes”.
12. Matrizes (**array** composto) = estrutura encadeada de vetores. Diferente dos vetores, são tridimensionais. São como tabelas com eixos X e Y, enquanto que os vetores só têm o eixo X.