

# KOKKU DEV TEST

## Modifications

João Fortunato Jr.




- First, I fixed a compilation error, the line responsible for adding a new tile to the grid was being called from the wrong place in the Grid class.

```
public Grid(int Lines, int Columns)
{
    xLenght = Lines;
    yLength = Columns;
    Console.WriteLine("The battle field has been created\n");
    for (int i = 0; i < Lines; i++)
    {
        grids.Add(newBox);
        for(int j = 0; j < Columns; j++)
        {
            GridBox newBox = new GridBox(x:j, y:i, occupied:false, index:(Columns * i + j));
            Console.Write($"{newBox.Index}\n");
        }
    }
}
```

```
public Grid(int Lines, int Columns)
{
    xLenght = Lines;
    yLength = Columns;
    Console.WriteLine("The battle field has been created\n");
    for (int i = 0; i < Lines; i++)
    {
        for(int j = 0; j < Columns; j++)
        {
            GridBox newBox = new GridBox(x:j, y:i, occupied:false, index:(Columns * i + j));
            grids.Add(newBox);
            Console.Write($"{newBox.Index}\n");
        }
    }
}
```

- The next step was to refactor the code, starting with the naming of variables and functions following Microsoft Coding Conventions.
- Explicit access modifiers.

- I removed two classes that were in the root of the project, but that were not being used for anything and were expendable.
- It was set to choose the enemy randomly only from 1 to 3, and it should be from 1 to 4, so I fixed it.
- The board size was hard coded.
- The game was not asking for grid size.
- I added input validation.
- I replaced the grid List with a matrix to make it easier to deal with and leaner.
- Removed references to libraries that are not used

```
1 using System;
2 using System.Collections.Generic;
3 using System.Text;
4
5 namespace AutoBattle
6 {
7      3 usages  João Fortunato Jr
8     public class Types
9     {
10
11          João Fortunato Jr
12         public struct CharacterClassSpecific
13         {
14             CharacterClass CharacterClass;
15             float hpModifier;
16             float ClassDamage;
17             CharacterSkills[] skills;
18         }
19     }
20 }
```

```

namespace AutoBattle
{
    3 usages João Fortunato Jr
    public class Types
    {
        João Fortunato Jr
        public struct CharacterClassSpecific
        {
            CharacterClass CharacterClass;
            float hpModifier;
            float ClassDamage;
            CharacterSkills[] skills;
        }
    }
}

```

- Player and enemy starting tile were not random.
- Character position variable was passed to Character class instead of Program.
- Rewrote the way to detect the character's action, both attack and movement.
- Classes were created for each character class, instead of just one enum.