

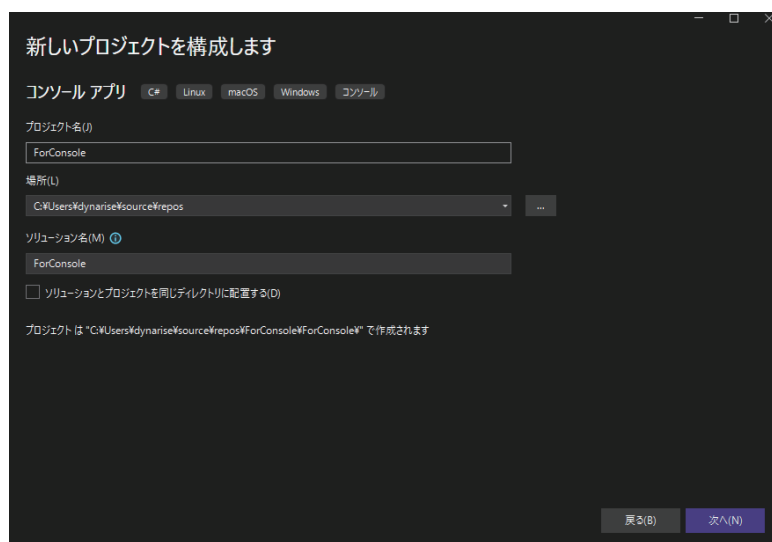
C# 課題 レベル 3

レベル 2 と同じ配列のコンソール出力を

「for 文 (繰り返し構文)」を使って効率的に行いたい。 (p86 のプログラムを参考にしよう)

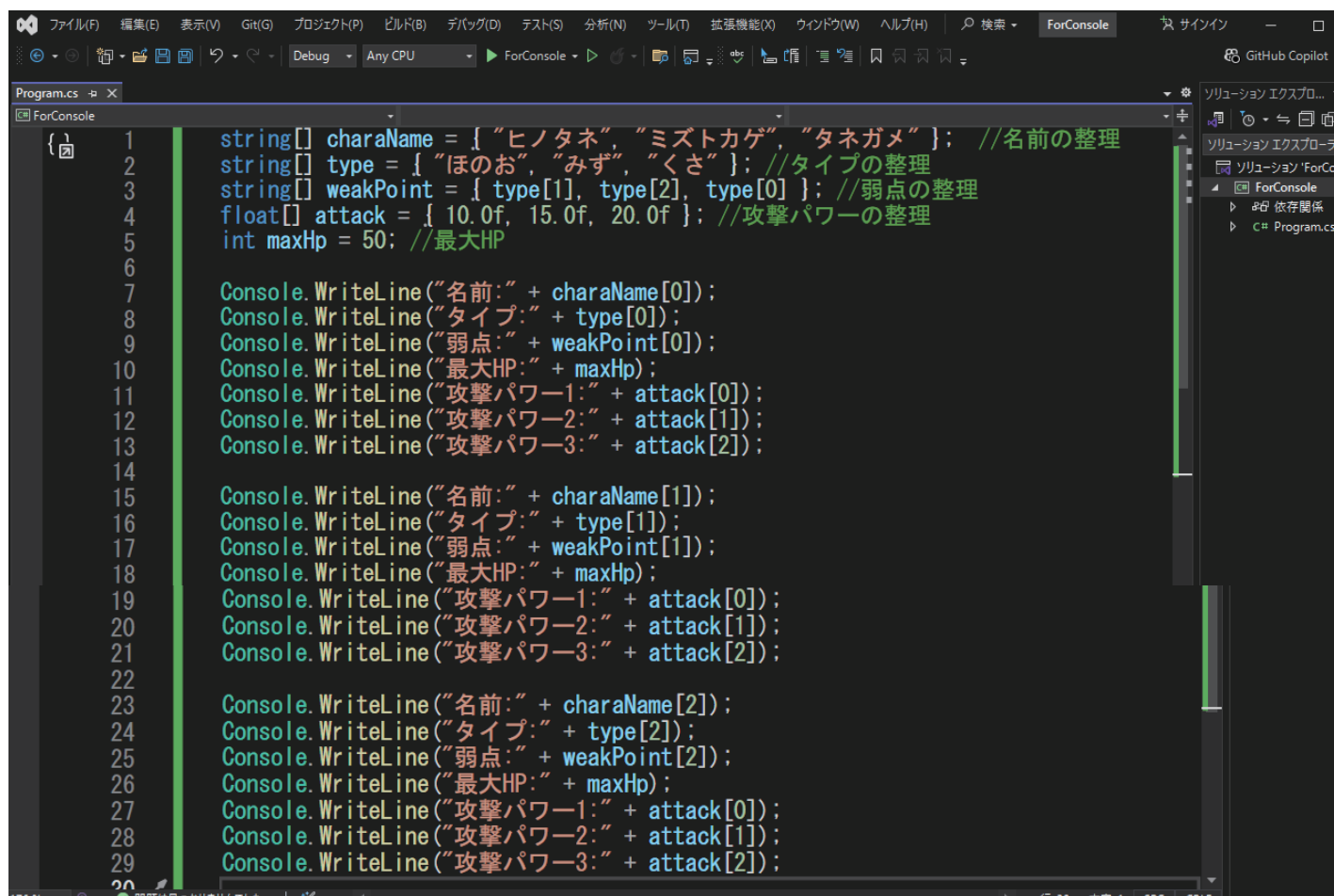
※レベル 2 ですでに for 文を使った出力ができている人はレベル 3 の問題②にチャレンジ!

配列をいくつか用意して、for 文で効率的にコンソールに表示するプロジェクトを作成したいです。まずは新規プロジェクト「ForConsole」を作成して、自動生成された「Program.cs」にプログラムが記述できる準備をしましょう。



Program.cs にレベル 2 の回答と同じプログラムを用意しましょう。(Copy & Paste しても OK)

参考: 下記はレベル 2 の模範解答 ※まったく同じでなくても OK



Program.cs の改良

① コンソール出力を for 文をつかって効率よく行いましょう。

※模範解答を例にしています

```
Console.WriteLine("名前:" + charaName[0]);  
Console.WriteLine("タイプ:" + type[0]);  
Console.WriteLine("弱点:" + weakPoint[0]);  
Console.WriteLine("最大HP:" + maxHp);  
Console.WriteLine("攻撃パワー1:" + attack[0]);  
Console.WriteLine("攻撃パワー2:" + attack[1]);  
Console.WriteLine("攻撃パワー3:" + attack[2]);
```

```
Console.WriteLine("名前:" + charaName[1]);  
Console.WriteLine("タイプ:" + type[1]);  
Console.WriteLine("弱点:" + weakPoint[1]);  
Console.WriteLine("最大HP:" + maxHp);  
Console.WriteLine("攻撃パワー1:" + attack[0]);  
Console.WriteLine("攻撃パワー2:" + attack[1]);  
Console.WriteLine("攻撃パワー3:" + attack[2]);
```

```
Console.WriteLine("名前:" + charaName[2]);  
Console.WriteLine("タイプ:" + type[2]);  
Console.WriteLine("弱点:" + weakPoint[2]);  
Console.WriteLine("最大HP:" + maxHp);  
Console.WriteLine("攻撃パワー1:" + attack[0]);  
Console.WriteLine("攻撃パワー2:" + attack[1]);  
Console.WriteLine("攻撃パワー3:" + attack[2]);
```

for 文を使わないと、
同じような事を 3 回記述していて
非常に冗長です。

for 文によって
処理をまとめましょう。

この時、カウンタ変数 *i* を
上手に活用しますが、
カウンタ変数 *i* をうまく
当てたい配列と、
逆にカウンタ変数を
使いたくない配列が
でてくると思うので
注意深く考えましょう。

出力結果は今までと同じ

```
名前:ヒノタネ  
タイプ:ほのお  
弱点:みず  
最大HP:50  
攻撃パワー1:10  
攻撃パワー2:15  
攻撃パワー3:20  
名前:ミストカゲ  
タイプ:みず  
弱点:くさ  
最大HP:50  
攻撃パワー1:10  
攻撃パワー2:15  
攻撃パワー3:20  
名前:タネガネ  
タイプ:くさ  
弱点:ほのお  
最大HP:50  
攻撃パワー1:10  
攻撃パワー2:15  
攻撃パワー3:20
```

② 次のデータを配列に足して出力しましょう。(4 匹目のキャラクター)
charaName: ヒカッチュウ type: かみなり weakPoint: みず
maxHp、3 つの attack パワーの値は他のキャラクターと同じ

```
名前:ヒカッチュウ  
タイプ:かみなり  
弱点:みず  
最大HP:50  
攻撃パワー1:10  
攻撃パワー2:15  
攻撃パワー3:20
```

出力結果