**getNames()**

10088% *of* 667 *of* 3,273[jhoffner](https://www.codewars.com/users/jhoffner)

C#

* [Train Again](https://www.codewars.com/kata/getnames/train/csharp)
* [Next Kata](https://www.codewars.com/trainer/csharp)

Details

[Solutions](https://www.codewars.com/kata/getnames/solutions/csharp)

[Forks (2)](https://www.codewars.com/kata/getnames/forks/csharp)

[Discourse (10)](https://www.codewars.com/kata/getnames/discuss/csharp)

* Add to Collection
* |
* Share this kata:

The following code is not giving the expected results. Can you figure out what the issue is?

The following is an example of data that would be passed in to the function.

public class Person

{

public int Age;

public string Name;

public Person(string name = "John", int age = 21)

{

Age = age;

Name = name;

}

}

Person[] data = new Person[]

{

new Person("Joe", 20),

new Person("Bill", 30),

new Person("Kate", 23)

};

Kata.GetNames(data) => new string[] {"Joe", "Bill", "Kate"};

<https://www.codewars.com/kata/getnames/csharp>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication2

{

class Program

{

public class Person

{

public int Age;

public string Name;

public Person(string name = "John", int age = 21)

{

Age = age;

Name = name;

}

}

Person[] data = new Person[]

{

new Person("Joe", 20),

new Person("Bill", 30),

new Person("Kate", 23)

};

public static string[] GetNames(Person[] data)

{

List<string> names = new List<string>();

for (int i = 0; i < data.Length; i++)

{

names.Add(data[i].Name);

}

return names.ToArray();

}

public static string[] GetNames(Person[] data)

{

return data.Select(d => d.Name).ToArray<string>();

}

static void Main(string[] args)

{

}

}

}