018 TypeScript - arrays using an interface

TypeScript Kata List – Blog Page

[TypeScript Kata List on GitHub](https://github.com/robertdunaway/katas-typescript)

# Duration

10 minutes

# Brief

[subject of kata]

### For More Information

BING/GOOGLE: “TypeScript arrays interface”

# Instructions

Get tutorial folder or the entire katas-typescript repo.

Open the [before/\*.sln] file and execute the kata.

Feel free to execute this kata multiple times because repetition creates motor memory.

# Github

* Before (start kata with this)
  + https://github.com/robertdunaway/katas-typescript/tree/master/018%20TypeScript%20-%20arrays%20using%20an%20interface/before
* After
  + https://github.com/robertdunaway/katas-typescript/tree/master/018%20TypeScript%20-%20arrays%20using%20an%20interface/after

# Kata

Create a basic string array and send its content to the console.

var pets: string[] = ['Jasmin', 'Roxie', 'Sally', 'Rush'];

console.log(pets);

Create an interface.

interface pet {

name: string;

age: number;

weight: number;

}

Create an empty array based on the pet interface.

var complexPetsArray: pet[] = [];

Push each dog onto the new array.

complexPetsArray.push({

name: 'Jasmin',

age: 9,

weight: 55

});

Create a new object of type pet and push it onto the array.

var roxie: pet = {

name: 'Roxie',

age: 6,

weight: 85

}

complexPetsArray.push(roxie);

console.log(complexPetsArray);

Create an array of the last two dogs, 'Sally' and 'Rush' then push the array onto the complexPetsArray and output to the console..

var myPets: pet[] = [{ name: 'Sally', age: 18, weight: 85 },

{ name: 'Rush', age: 15, weight: 45}];

// complexPetsArray.push(myPets);

We are not allowed to do this but we can use a foreach loop and get it done that way.

Loop over myPets and push each onto the complexPetsArray and output to the console.

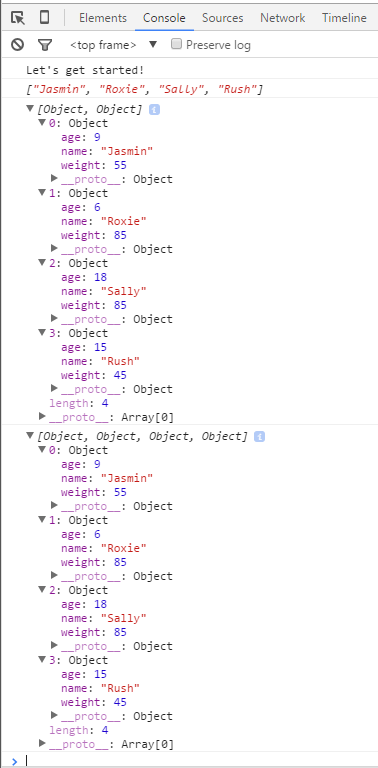
for (var p of myPets) {

complexPetsArray.push(p);

}

console.log(complexPetsArray);

End result



# Next

Take a few minutes and imagine more examples.