010 TypeScript - ternary operator

TypeScript Kata List – Blog Page

TypeScript Kata List – Github Page

# Duration

5 minutes

# Brief

Using a ternary operator (?) to check for null.

### For More Information

BING/GOOGLE: “TypeScript ternary operator ?”

# 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/010%20TypeScript%20-%20ternary%20operator/before
* After
  + https://github.com/robertdunaway/katas-typescript/tree/master/010%20TypeScript%20-%20ternary%20operator/after

# Kata

Recreate the logic below with the ternary operator (?). The ternary operator can be used to look for null but what it’s really doing is creating conditional logic based on a true/false response. If null then false is returned.

var myVal: number = null;

var result: number = null;

// using an if statement

if (myVal == null) {

result = 0;

}

else {

result = myVal;

}

Now change the “myVal” variable to equal 5 and execute operation again. Result is now set equal to “myVal” because “myVal” is not null.

[highlight below for one possible answer]

// Using a ternary operator to check for null.

result = myVal ? myVal : 0;

console.log('result = myVal ? myVal : 0; // result = ' + result);

myVal = 5;

result = myVal ? myVal : 0;

console.log('result = myVal ? myVal : 0; // result = ' + result);

Create logic that checks if two values are equal. At this point in the exercise result and myVal should both be 5 so a ternary operator will return true for (myVal = return).

[highlight below for one possible answer]

// Using a ternary operator to evaluate.

result = (myVal == result) ? 'myVal and result both equal ' + result : 'values are not equal';

console.log(result);

