

We experimented with primitive data types in JavaScript, as is the lesson contained in Module 2, Section 2 of the JavaScript Essentials (JSE) program in Cisco's NetAcad learning management system. Instructions given were as follows:

Study Module 2, Section 2 of our Netacad Course. Do the Tasks at the end of Section 2.

*Copy the given Question, attach screenshot of code and output per task.
Submit in a PDF file*

Type: Dropbox
Max score: 100
Category: Enabling Assessment
Start: Feb 12, 7:00 am
Due: Feb 12, 10:00 am
Max. attempts: 2
Allow late submissions:

The Six Micro-Tasks

These tasks are accessible to all JSE enrollees in NetAcad, using this link:

<https://edube.org/learn/jse1-1-0/data-types-tasks>

Task 1:

Write a code that will create variables and initialize them with values of Boolean, Number, BigInt, String, and undefined types using (when possible) literals and constructor functions.

Code

```
let isItHot = false;
let isWeatherRainy = Boolean(true);

let subsectionNum = 22;
let michaelJordan = Number(23);

let soManyDigits1 = 1246569035864034586n;
let soManyDigits2 = BigInt(2346569035864034586);

let greeting = "Hello!";
let farewell = String("Buh-bye!");

let nope = undefined;

console.log(`typeof isItHot contains: ${isItHot}`);
console.log(`typeof isWeatherRainy contains: ${isWeatherRainy}`);
console.log(`typeof subsectionNum contains: ${subsectionNum}`);
console.log(`typeof michaelJordan contains: ${michaelJordan}`);
console.log(`typeof soManyDigits1 contains: ${soManyDigits1}`);
console.log(`typeof soManyDigits2 contains: ${soManyDigits2}`);
console.log(`typeof greeting contains: ${greeting}`);
console.log(`typeof farewell contains: ${farewell}`);
console.log(`typeof nope contains: ${nope}`);
```

Task 2:

Print all values and all types of those values using `console.log`. Try to use string interpolation to display the value and type at the same time with a single `console.log` call, e.g. in the following form: 1000 [number].

Output

```
boolean contains: false
boolean contains: true
number contains: 22
number contains: 23
bigint contains: 1246569035864034586
bigint contains: 2346569035864034816
string contains: Hello!
string contains: Buh-bye!
undefined contains: undefined
```

Task 3:

Carry out a chain of conversions: create a Boolean from a BigInt created from a Number that was created from a String. Start with the value "1234". Is it possible?

Code	Output
<pre>let testObject = "1234"; console.log(`testObject encapsulates: \${testObject}`); let testObjectConv1 = Number(testObject); console.log(`tO conversion 01 encapsulates: \${testObjectConv1}`); let testObjectConv2 = BigInt(testObject); console.log(`tO conversion 02 encapsulates: \${testObjectConv2}`); let testObjectConv3 = Boolean(testObject); console.log(`tO conversion 03 encapsulates: \${testObjectConv3}`);</pre>	<pre>testObject encapsulates: 1234 tO conversion 01 encapsulates: 1234 tO conversion 02 encapsulates: 1234 tO conversion 03 encapsulates: true</pre>

Task 4:

Try adding two values of the same type and check the result type. Try it for all primitive types.

Code	Output
<pre>let addedBoolean = true + false; let addedNumber = 500 + 70; let addedString = "Ani" + "Mo!"; let addedBigInt = 1654465864564169819856n + 540684555627845586455857n; let addedUndefined = undefined + undefined; console.log(`true + false = \${addedBoolean}`); console.log(`500 + 70 = \${addedNumber}`); console.log(`Ani + Mo! = \${addedString}`); console.log(`1654465864564169819856n + 540684555627845586455857n = \${addedBigInt}`); console.log(`undefined + undefined = \${addedUndefined}`);</pre>	<pre>true + false = 1 500 + 70 = 570 Ani + Mo! = AniMo! 1654465864564169819856n + 540684555627845586455857n = 542339021492409756275713 undefined + undefined = NaN</pre>

Task 5:

Try adding two values of different types and check the results.

Code	Output
------	--------

```

45 // All variables commented out result in errors.
47 let npb = 400 + true;
48 let nps = 450 + "they say omg";
49 let np1 = 350 + 18630660797074268568515451414565n;
50 let np2 = false + 3682;
51 let bps = true + "dat lol";
52 // let bpl = true + 85439306858869485943n;
53 let spn = "this program's code num is " + 223;
54 let spb = "what you just said is patently " + false;
55 // let spi = "smash your keyboard and you get " + 6965924725425625256452156285n;
56 // let ips = 214545774878544515789657858154n + " icons are on my iMac's desk";
57 // let ipn = 2524524424141241741245215441n + " has a very predictable pattern";
58 // let ipb = 286748596976948938n + false;
59
60 console.log('number + boolean = ${npb}');
61 console.log('number + string = ${nps}');
62 console.log('boolean + number = ${bpn}');
63 console.log('boolean + string = ${bps}');
64 console.log('string + number = ${spn}');
65 console.log('string + boolean = ${spb}');
66

```

number + boolean = 401
 number + string = 450they say omg
 boolean + number = 3682
 boolean + string = truedat lol
 string + number = this program's code num is 223
 string + boolean = what you just said is patently false

Task 6:

Try to modify the line `const str1 = 42 + "1";` to get the result 43 (without removing the quotes around 1).

Code	Output
<pre> 66 67 const str1 = 42 + +"1"; 68 console.log("str1 is now:", str1); 69 </pre>	str1 is now: 43