// var myName = 'Kadie';

// var myAge = 29;

// var myBirthday;

// var actualbirthday = '02/06/1997';

// console.log('Hello World!');

// // => Hello World!

// console.log(myName);

// // => Kadie

// console.log(myAge);

// // => 29

// console.log(myBirthday);

// // => undefined

// console.log(actualbirthday);

// // => 02/06/1997

// var x = [1, 2, 3];

// var y = x;

// x.push('Hello world!')

// // push adds to the end of arrays in JavaScript

// console.log(x);

// console.log(y);

// let x = Math.floor(Math.random() \* 20);

// console.log(x);

// // => logs random number between 0 and 19

// let y = Math.floor(Math.random() \* 21);

// console.log(y);

// // => logs a random number between 0 and 20

// let z = Math.floor(Math.random() \* 20) + 1;

// console.log(z);

// // => logs a random number between 1 and 20

// let x = 11;

// let y = 5;

// let z = x % y;

// console.log(z);

// // => 1, % returns the remainder after division. Meaning 5 goes into 11 two times leaving with 1 left over, that will return as the answer.

// let x = 16;

// console.log(x % 2);

// // when we modulus by 2, even numbers will return 0. It returns 0 due to there not being anything leftover to return, as 2 is able to entirely divide into 16.

// let y = 17;

// console.log(y % 2);

// // while odd numbers will return a value that isn't 0! As 17 is an odd number and there's 1 left over after division, it will return the remainder being 1.

// //JS NORMAL

// let firstName = "Oscar";

// let lastName = "Vazquez";

// let message = "Hello, my name is " + firstName + " " + lastName;

// console.log(message);

// //ES6 METHOD BELOW

// let firstName = "Oscar";

// let lastName = "Vazquez";

// let message = `Hello, my name is ${firstName} ${lastName}`;

// // Take note of the backticks instead of quotes

// console.log(message);

// // ES6 JS, \n causes that string to move over to the next line, just like when you're writing a poem or a parragraph and have to move on to the next line due to space.

// let haiku = "Having been erased, \nThe document you're seeking \nMust now be retyped."

// console.log(haiku);

// // ES6 Method, \u is short for Unicode. Instead of having to copy and paste symbols or using the keyboard, you can use this to directly print specific symbols.

// console.log('\u263A \u2603 \u272f');

// let x = [];

// x[334] = 'Hello world!';

// console.log(x); //Since we have an empty array and x is calling for 334 to print 'Hello World', 'Hello World' will only print on the value 334 and the rest will show up as 'undefined' as we are only declaring a single value.

// let num = 1;

// while (num < 6){

// console.log("I'm counting! The number is " + num);

// num = num + 1;

// }

// console.log("We are done. Goodbye world!");

// let num = 1;

// while (num < 6){

// console.log("I'm counting! The number is " + num);

// num = num + 1;

// }

// console.log("We are done. Goodbye world!");

// let colors = ['blue', 'green', 'red', 'chartreuse'];

// // a simple array of strings

// for(let i = 0; i < colors.length; i++){

// // by using the length of our colors array, we can make the condition

// // of our for loop match the number of elements in the array!

// console.log(colors[i]);

// // now we can use i to log the elements of the color array individually

// };

// let names = ['Anna', 'Oscar', 'Kadie', 'Steve', 'Elle', 'Boris', 'Lord Humongous'];

// for(let i = 0; i < names.length; i++){

// if(names[i] === 'Kadie'){

// console.log('Kadie is in our array!');

// break;

// }

// }

// console.log('We finished looping!');

// let names = ['Anna', 'Oscar', 'Kadie', 'Steve', 'Elle', 'Boris', 'Lord Humongous'];

// for(let i = 0; i < names.length; i++){

// if(names[i] === 'Steve'){ continue };

// console.log(names[i]);

// };