**1. Getting Started with Programming**

**What is your name?**

Hi! Let's get to know each other. What is your name?

**Instructions**

Write your name within quotes like this:"Ryan" Then click "Save & Submit Code".

"Zeljko"

**Discover the length**

Very good!   
What would the length of your name be?

**Instructions**

To discover the length of your name write your name within quotes. Then write a period (full stop) and the word **length** like this:  
"yourName".length

When you're done click **Save & Submit Code**   
(Do this from now on every time you finish your exercise.).  
In my case (my name is Leng) it would be"Leng".length

"Zeljko".length

**Basic math**

Great job! Now, let's do some math. You can do math through programming!

**Instructions**

Add any two numbers, like this: 3 + 4

3 + 4

**Numbers and more**

See what happened? You can use the command line to do basic math operations. Try playing around some more.

**Instructions**

You can use \* for multiplication and / for division if you want. Enter another valid expression to pass this lesson.

56/4

**Error: does not compute!**

There are some things you *can't* do in the console. Computers only speak certain languages, like the one you've been using today: JavaScript!

If you use words that aren't in the JavaScript language, it will get confused and give you an error.

**Instructions**

Try to confuse the interpreter by using a word it doesn't know, like eggplant. It will give you a ReferenceError.

Eggplant

**Editor and comments**

So far we've been writing lines of code in the editor. Now we see two lines that start with//.

The // sign is for **comments**. A comment is a line of text that JavaScript won't try to run as code. It's just for humans to read.

Comments make your program easier to understand. When you look back at your code or others want to collaborate with you, they can read your comments and easily figure out what your code does.

**Instructions**

The computer will ignore the code on [lines 1-2](javascript:void(0)), since it is commented out.

On [line 3](javascript:void(0)), find the length of the word "cake".

// This is a comment that the computer will ignore.

// It is for your eyes only!

"cake".length

**What am I learning?**

This is JavaScript (JS), a programming language. There are many languages, but JS has many uses and is easy to learn.

**What can we use JavaScript for?**

* make websites respond to user interaction
* build apps and games (*e.g.* blackjack)
* access information on the Internet (*e.g.*find out the top trending words on Twitter by topic)
* organize and present data (*e.g.* automate spreadsheet work; data visualization)

**Instructions**

Press Save & Submit Code to see an example of how JavaScript can be interactive!

confirm('This is an example of using JS to create some interaction on a website. Click OK to continue!');

**Interactive JavaScript**

What we just saw was a fun example of how JavaScript can be interactive. Try it yourself!

**Examples**:  
  
confirm("I feel awesome!");  
confirm("I am ready to go.");

These boxes can be used on websites to*confirm* things with users. You've probably seen them pop up when you try to delete important things or leave a website with unsaved changes.

**Instructions**

Write your own message that you want the user to confirm.

confirm('I feel awesome!');

**What is programming?**

Programming is like writing a list of instructions to the computer so it can do cool stuff with your information.

Programs can't yet make your bed, but they can do math, keep track of your bank account, or send a message to a friend.

To do any of these actions, the program needs an input. You can ask for input with a**prompt**.

**Examples**:  
  
1. prompt("What is your name?");  
2. prompt("What is Ubuntu?");

**Instructions**

Use the prompt command to ask the user where they are from. Check out the examples above for how to do this!

prompt("Where are you from?");

**Data Types I & II: Numbers & Strings**

Data comes in various **types**. You have used two already!

**a. numbers** are quantities, just like you're used to. You can do math with them.

**b. strings** are sequences of characters, like the letters a-z, spaces, and even numbers. These are all strings: "Ryan", "4" and "What is your name?" Strings are extremely useful as labels, names, and content for your programs.

To make a *number* in your code, just write a number as numerals *without quotes*: 42,190.12334.

To write a string, surround words *with quotes*:"What is your name?"

**Instructions**

1. Write a string with at least 3 words. Check out the examples of strings above.
2. Find the length of the string by writing a period (full stop) and the word length, like this:
3. "string".length

Length counts every character in the string - including spaces!

"Hello to everyone!".length

**Data Type III: Booleans**

Nice job! Next let's look at **booleans**. A boolean is either true or false.

For example, comparing two numbers returns a true or false result:

* 23 > 10 is true
* 5 < 4 is false

**Instructions**

Let's compare two numbers to return a true result:

1. First, write the string "I'm coding like a champ"
2. Next, find the length of the string using.length
3. Then, compare the string's length to see if it is greater than 10

If you want to check your code, click "Stuck? Get a hint!" below.

"I'm coding like a champ".length > 10

**Using console.log**

You may have noticed that the interpreter doesn't print out every single thing it does. So if we want to know what it's thinking, we sometimes have to ask it to speak to us.

console.log() will take whatever is inside the parentheses and log it to the console below your code—that's why it's calledconsole.log()!

This is commonly called **printing out**.

**Instructions**

Please print the following two console.log statements at the same time. Type one on[line 1](javascript:void(0)) and the other on [line 2](javascript:void(0)). Then press Save & Submit Code.

console.log(2 \* 5)  
console.log("Hello")

console.log(2 \* 5)

console.log("Hello")

**Comparisons**

So far we've learned about three data types:

* **strings** (*e.g.* "dogs go woof!")
* **numbers** (*e.g.* 4, 10)
* **booleans** (*e.g.* false, 5 > 4)

Now let's learn more about comparison operators.

**List of comparison operators**:

* > Greater than
* < Less than
* <= Less than or equal to
* >= Greater than or equal to
* === Equal to
* !== **Not** equal to

**Instructions**

Try to use each of the operators above.

1. Choose the correct comparison operator to make each of the four statements print outtrue.

// Here is an example of using the greater than (>) operator.

console.log(15 > 4); // 15 > 4 evaluates to true, so true is printed.

// Fill in with >, <, === so that the following print out true:

console.log("Xiao Hui".length < 122);

console.log("Goody Donaldson".length > 8);

console.log(8\*2 === 16);

**Decisions, decisions**

Nice work on comparisons! Now let's see how we can use comparisons to ask yes or no questions.

Say we want to write a program that asks whether your name is longer than 7 letters. If the answer is yes, we can respond with "You have a long name!" We can do this with an ifstatement:

if( "myName".length >= 7 ) {

console.log("You have a long name!");

}

An if statement is made up of the ifkeyword, a condition like we've seen before, and a pair of curly braces { }. If the answer to the condition is yes, the code inside the curly braces will run.

**Instructions**

Check out the if statement in the editor.

1. On [line 1](javascript:void(0)), add a condition inside the parentheses ( ).
2. If the answer to the condition is yes, the code inside the curly braces will run. So on[line 2](javascript:void(0)), use console.log to print out a string.

if ( "Zeljko".length <= 7 ) {

console.log( "You have a short name!" );

}

**Computers are smart**

Great! We used an if statement to do something if the answer to the condition was yes, or true as we say in JavaScript.

In addition to doing something when the condition is true, we can do something else if the condition is false. For example, if your name is shorter than 7 letters, we can respond with "You have a short name!" We can do this using an if / else statement:

if( "myName".length >= 7 ) {

console.log("You have a long name!");

}

else {

console.log("You have a short name!");

}

Just like before, if the condition is true, then only the code inside the first pair of curly braces will run. Otherwise, the condition isfalse, so only the code inside the second pair of curly braces after the else keyword will run.

In the example above the condition"myName".length >= 7 evaluates to falsesince "myName" only has 6 letters. Since the condition is false, only the code inside the curly braces after the else keyword runs, and prints out You have a short name!.

**Instructions**

1. In [line 1](javascript:void(0)), fill in a condition that will evaluate to false
2. Fill in some code to run in the elseportion (this will run if the condition is false). Use console.log for this part.

if ("urlnihvkjwenui" <= 10 )

{

console.log("Let's go down the first road!");

}

else

{

console.log("Let's go up on the second road!");

}

**More practice with conditionals**

Now let's practice using if/else statements. Do as much as you can by yourself, but if you need a reminder, click the "Stuck? Get a hint!" button below.

**Instructions**

1. Write an if/else statement, just like we did in the last exercise. Here's what the outline of the code looked like:
2. if (condition)
3. {
4. *// if condition is true*
5. *// do this code*
6. }
7. else *// "otherwise"*
8. {
9. *// do this code instead*
10. }
11. *If* your condition is true, use console.log to print "The condition is true".
12. Otherwise (*else*) when it is false, useconsole.log to print "The condition is false".
13. Make sure your condition evaluates to false, so that your program prints out "The condition is false".

if(3>5){

console.log("The condition is true");

}else{

console.log("The condition is false");

}

**Computers aren't that smart**

Well done! Now, computers are very literal. Syntax needs to be in exactly the right place for the computer to understand the code.

As you get started with programming, we will teach you many syntax rules. This is sort of like the grammar of programming languages. Grammar first, then programming poetry!

**Instructions**

There are many mistakes in this code. Find them and fix them all.

You are doing what's called "debugging," a term popularized by [Grace Hopper](http://en.wikipedia.org/wiki/Grace_Hopper) when she literally [removed a moth](http://en.wikipedia.org/wiki/File:H96566k.jpg) from her computer.

if (10 === 10) {

console.log("You got a true!");

} else {

console.log("You got a false!");

}

**Mid-lesson breather**

We've covered a lot of ground so far! So many new terms, so much syntax. Let's take a breather and review. We have learned:

**1. Confirm and prompt**  
  
We can make pop-up boxes appear!   
confirm("I am ok");  
prompt("Are you ok?");

**2. Data types**  
  
a. numbers (*e.g.* 4.3, 134)  
  
b. strings (*e.g.* "dogs go woof!", "JavaScript expert")  
  
c. booleans (*e.g.* false, 5 > 4)

**3. Conditionals**  
  
*If* the first condition is met, execute the first code block. *If* it is not met, execute the code in the else block. See the code on the right for another example.

**Instructions**

Hope this breather was helpful! Click 'Save and Submit' to continue.

// This is an example of an if / else statement.

if (12 / 4 === "Ari".length) {

confirm("Will this run the first block?");

} else {

confirm("Or the second block?");

}

**Math**

We saw basic math before. The basic math symbols we learned in school work here. Even the order in which the computer understands the math is the same as in school!

**Code**:  
  
1. ( ): control order of operations  
2. \* and /: multiplication and division  
3. - and +: subtraction and addition

**Examples**:  
  
1. 100/10 evaluates to 10  
2. "Jane".length + 5 evaluates to 9  
3. 5\*(3+1) evaluates to 20

**Instructions**

1. Complete the missing bits of code to construct the if / else statement. Make the condition evaluate to true.
2. Finish the else statement by printing out the string "Error Error Error" to the console.

if("Jon".length \* 2 / (2+1) === 2)

{

console.log("The answer makes sense!");

}

else {

console.log("Error Error Error");

}

**Math and the modulo**

Let's meet an interesting symbol called**modulo**. When % is placed between two numbers, the computer will divide the first number by the second, and then return the**remainder** of that division.

So if we do 23 % 10, we divide 23 by 10 which equals 2 with 3 left over. So 23 % 10 evaluates to 3.

**More examples**:  
  
17 % 5 evaluates to 2  
  
13 % 7 evaluates to 6

**Instructions**

Use console.log and modulo three times to print the remainder of the following equations:

a. 14 / 3  
b. 99 / 8  
c. 11 / 3

console.log(14%3);

console.log(99%8);

console.log(11%3);

**Modulo and if / else**

So why learn modulo? For one thing, it's good at testing divisibility. Consider 30 % 10. What does it return? There is nothing left over, so0.

How about 9 % 3? Also 0.

We can use modulos in comparisons, like this:

* 10 % 2 === 0 evaluates to true
* 7 % 3 === 0 evaluates to false because there is 1 left over.

**Instructions**

Let's get the if/else" statement to display"The first number is even".

1. Edit [line 5](javascript:void(0)) by adding a comparison that evaluates to true.
2. In the comparison, use a modulo and an even number, like we did in the example above.

if(23%5 < "example".length ) {

console.log("The first number is even");

} else {

console.log("The first number is odd");

}

**Substrings**

We've learned a few ways to manipulate numbers. What about manipulating strings?

Sometimes you don't want to display the*entire* string, just a part of it. For example, in your Gmail inbox, you can set it to display the first 50 or so characters of each message so you can preview them. This preview is a*substring* of the original string (the entire message).

**Code**:  
  
"some word".substring(x, y) where x is where you start chopping and y is where you finish chopping the original string.

The number part is a little strange. To select for the "he" in "hello", you would write this:

"hello". substring(0, 2);

Each character in a string is numbered starting from 0, like this:

0 1 2 3 4

| | | | |

h e l l o

The letter h is in position 0, the letter e is in position 1, and so on.

Therefore if you start at position 0, and slice right up till position 2, you are left with justhe

**More examples**:  
  
1. First 3 letters of "Batman"  
"Batman".substring(0,3);  
  
2. From 4th to 6th letter of "laptop"  
"laptop".substring(3,6);

**Instructions**

Find the 4th up to and including the 7th letter of the string "wonderful day".

"wonderful day".substring(3,7);

**More substring practice**

Getting the positioning of substring letter positions is tricky! Let's make sure we really have it nailed down.

Remember that each character in a string is numbered starting from 0. So for the word "hello", The letter h is in position 0, the letter e is in position 1, and so on.

**Instructions**

Using console.log, on three separate lines, print out the substrings for the following strings.

a. "Jan" in "January"  
b. "Melbourne is" in "Melbourne is great" (note the space!)  
c. "burgers" in "Hamburgers"

console.log("January".substring(0,3));

console.log("Melbourne is great".substring(0,12));

console.log("Hamburgers".substring(3,10));

**Variables**

We have learned how to do a few things now: make strings, find the length of strings, find what character is in the nth position, do basic math. Not bad for a day's work!

To do more complex coding, we need a way to 'save' the values from our coding. We do this by defining a variable with a specific, case-sensitive name. Once you create (or**declare**) a variable as having a particular name, you can then call up that value by typing the variable name.

**Code**:  
  
var varName = data;

**Example**:  
  
a. var myName = "Leng";  
b. var myAge = 30;  
c. var isOdd = true;

**Instructions**

Create a variable called myAge and type in your age.

var myAge = 27;

console.log(myAge);

**More Variable Practice**

We have seen how to create a variable. But how do we use it? It is useful to think that any time you type the variable's name, you are asking the computer to swap out the variable name and swap in the *value* of the variable.

**For example:**  
  
var myName = "Steve Jobs";  
  
myName.substring(0,5)

Look at the second line above. You have asked the computer to swap out myName and swap in Steve Jobs, so

myName.substring(0,5)

becomes

"Steve Jobs".substring(0,5)

which evaluates to Steve.

**Another example**

var myAge = 120;

What is

myAge % 12 ? See the hint to check your answer.

So the variable stores the *value* of the variable, whether that is a number or a string. As you will see soon, this makes writing long programs much easier!

**Instructions**

Follow the instructions in the comments in the code to continue.

// Declare a variable on line 3 called

// myCountry and give it a string value.

var myCountry = "Croatia";

// Use console.log to print out the length of the variable myCountry.

console.log(myCountry.length);

// Use console.log to print out the first three letters of myCountry.

console.log(myCountry.substring(0,3));

**Change variable values**

So far, we've seen  
a. how to create a variable  
b. how to use a variable

Let's now see how to change a variable's value. A variable's value is easily changed. Just pretend you are creating a new variable while using the same name of the existing variable!

**Example**:  
  
var myAge = "Thirty";  
Say I had a birthday and I want to change my age.  
myAge = "Thirty-one";

Now the value of myAge is "Thirty-one"!

**Instructions**

Follow the instructions on [line 1](javascript:void(0)), [line 3](javascript:void(0)), [line 5](javascript:void(0))and [line 8](javascript:void(0)). We're using this method to show you the order in which you tell the computer what to do is very important.

// On line 2, declare a variable myName and give it your name.

myName = "Zeljko";

// On line 4, use console.log to print out the myName variable.

console.log(myName);

// On line 7, change the value of myName to be just the first 2

// letters of your name.

console.log(myName.substring(0,2));

// On line 9, use console.log to print out the myName variable.

console.log(myName);

**Conclusion: Part 1**

Let's do a quick review!

**Data types**

* strings (e.g. "dogs go woof!")
* numbers (e.g. 4, 10)
* booleans (e.g. false, 5 > 4)

**Variables**  
We store data values in variables. We can bring back the values of these variables by typing the variable name.

**Manipulating numbers & strings**

* comparisons (e.g. >, <=)
* modulo (e.g. %)
* string length (e.g. "Emily".length;)
* substrings (e.g. "hi".substring(0, 1);)

**console.log( )**   
Prints into the console whatever we put in the parentheses.

**Instructions**

1. On [line 1](javascript:void(0)), create a variable myColor and give it a string value.
2. On [line 2](javascript:void(0)), print the length of myColor to the console.

var myColor = "Black";

console.log(myColor.length);

**Conclusion: Part 2**

Congratulations on making it this far. You have learned a lot! Just one more exercise before a big pat on the back!

The last tricky thing we learned was about **if / else** statements.

If / else statements are conditional statements. Under different conditions, the computer will output different things.

**Instructions**

1. Write your own if / else statement.
2. The only instruction is that the result of evaluating the statement is a log to the console of "I finished my first course!".

if("Zeljko".length >= 25%7){

console.log("I finished my first course!");

}

else{

console.log("I didn't finish my first course!");

}