JavaScript Basics

JavaScript Basics

Home

* JavaScript OverView

o Variable Declaración o Variable Assignment o DataTvpe o Conditionals o Loops o Function

* Proiects
* Assignments
* OüÍ7.

JavaScript is a cross-platform, object-oriented scripting language. JavaScript is extremely popular for a variety of reasons. It is a small and lightweight language allowing máximum flexibility for developers to take it in a bunch of different directions. JavaScript lives inside a host environment (a web browser or Node server), it can be connected to tbe objects of these environments to provide programmatic control over them.

* Variable Declararon JavaScript variables are containers for storing data valúes \* imagine a cup you fill with coffee, tbe cup holds tbe coffee, a variable bolds a valué. All JavaScript variables musí be identified with unique ñames. These unique ñames are called identifiers. var x;
* Variable Assignment Assignment operators assign valúes to JavaScript variables - our cup can now have coffee poured in it, giving our variable a valué to hold. The = assignment operaíor assigns a valué to a variable, var x = 10;
* Data Types Data types are an important concept; to be able to opérate on variables you need to know the data type. Tbere are six data types that are JavaScript primitives: Boolean - true or false; nuil • nuil aka nothing; Number • 42 or 3.14159; String - "Coding Dojo Rocks!"; Array - [1, ’Coding’, 2, ’Dojo’J; and Object - {first.name: 'Jane', last.name: ’Doe'}
* Conditionals When you write code, you want to perform different actions for different decisions - hitting different code blocks based on valúes or conditions that have been met. You can use condicional statements in your code to accomplish this. Tbere are the following conditional statements: if a specified condition is true, do this code in our code block; else if to specify a new condition to test, if the first condition is false; else we execute this block of code;
* Loops Tbere are many different kinds of loops in every programming language, but tbey all essentially do tbe same thing: tbey will repeat an action some number of times. Imagine you have to run a mile, well you run around tbe track four times and tben you stop. Thats a loop!
* Function Functions are an encaspulation of a code block. When we cali our function this will run that code block. Think of it as a list of instructions. As an example imagine we are putting togetber a desk from Ikea, we open up the instruction manual and get started, first we screw tbe legs to the table top; next we place the table tbe right way up. Done! We finished our instructions. Sadly diere are a ton more tables to do so lets cali our ñmction over and over and over again.

For more useftil Information check out this url: JavaScript!