<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>JavaScript Basics</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<div class="container">

<div class="header">

<h1>JavaScript Basics</h1>

</div>

<div class="navbar">

<a href="">Home</a>

<ul>

<li><a href="">JavaScript Overview</a></li>

<ul>

<li><a href="">Variable Declaration</a></li>

<li><a href="">Variable Assignment</a></li>

<li><a href="">Data Type</a></li>

<li><a href="">Conditionals</a></li>

<li><a href="">Loops</a></li>

<li><a href="">Function</a></li>

</ul>

<li><a href="">Projects</a></li>

<li><a href="">Assignments</a></li>

<li><a href="">Quiz</a></li>

</ul>

</div>

<div class="main-content">

<h2>JavaScript Basics</h2>

<p>JavaScript is a cross-platform, object-oriented scripted language. JavaScript is extremely popular for a variety of reasons. It is a small and lightweight language allowing maximum 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 the objects of these environments to provide programmatic control over them.</p>

<ul>

<li><p><a href="">Variable Declaration</a> JavaScript variables are containers for storing data values - imagine a cup you fill with coffee, the cup holds the coffee, a variable holds a value. All JavaScript variables must be identified with unique means. These unique names are called identifiers. <span>var x;</span></p></li>

<li><p><a href="">Variable Assignment</a> Assignment operators assign values to JavaScript variables - our cup can now have coffee poured in it, giving out variable a value to hold. The = assignment operator assigns a value to a variable. <span>var x = 10;</span></p></li>

<li><p><a href="">Data Type</a> Data Types are an important concept; to be able to operate on variables, you need to know the data type. These are six data types that are JavaScript primitives: Boolean - <span>true</span> or <span>false;</span> null - <span>null</span> aka nothing; Number - <span>42</span> or <span>3.14159;</span> String - <span>"Coding Dojo Rocks!";</span> and Object - <span >{first\_name: 'Donovan', last\_name:'An'}</span></p></li>

<li><p><a href="">Conditionals</a> When you write code, you want to perform different actions for different decisions - hitting different code blocks based on values or conditions that have been met. You can use conditional statements in your code to accomplis this. There are the following conditional statements: <span>if</span> a specified condition is true, do this code in our code block; <span>else if</span> to specify a new condition to test, if the first condition is false; <span>else </span> we execute this block of code.</p></li>

<li><p><a href="">Loops</a> There are many different kinds of loops in every programming language, but they all essentially do the same thing: they will repeat an action some number of times. Imagine you have to run a mile, well you run around the track four times and then you stop. That's a loop!</p></li>

<li><p><a href="">Function</a> Functions are an encapsulation of a code block. When we call out function this will run that code block. Think of it as a list of instructions. As an example imagine we are putting together a desk from Ikea, we open up the instruction manual and get started, first we screw the legs to the table top; next we place the table the right way up. Done! We finished our instructions. Sadly there a ton more tables to do so lets call our function over and over again.</p></li>

</ul>

</div>

<div class="footer">

<p>For more information, check out this url: <a href="">JavaScript!</a></p>

</div>

</div>

</body>

</html>

CSS

.container{

width: 970px;

background-color: #1d355c;

padding: 10px;

}

a{

color: #61cac7;

font-size: 12px;

}

p{

color: #c4d8ec;

font-size: 12px;

}

.header{

width: 100%;

/\*height: 100px;\*/

background-color: #cc323b;

border-radius: 10px;

margin-bottom: 10px;

}

.header h1{

margin: 0;

padding: 15px 0 15px 10px;

color: #c3d6eb;

}

.navbar{

width: 180px;

background-color: #2f7d8b;

border-radius: 10px;

padding: 10px;

display: inline-block;

vertical-align: top;

}

ul{

color: white;

padding-left: 20px;

margin: 0;

}

.main-content{

width: 720px;

display: inline-block;

margin-left: 15px;

background-color: #0f1822;

border-radius: 10px;

color: #c4d8ec;

padding: 15px;

}

.main-content h2{

margin: 10px 0;

}

.main-content ul li p{

margin: 5px 0;

}

span{

color: #6d80ce;

}

.footer{

margin-top: 10px;

width: 100%;

background-color: #5e6672;

border-radius: 15px;

}

.footer p{

padding: 15px 0 15px 15px;

}