Ubiquitous Language of Page Representation

In this lesson, we'll study the ubiquitous language of page representation.
Let's begin!

WE'LL COVER THE FOLLOWING

The road to defining a standard for the HTML markup

For a moment, try to think with the mind of an HTML standard creator.

The standard defines *more than a hundred* markup tags with their semantics, properties, and behavior, including the visual appearance.

How would you describe a hundred markup tags in a formal way so that all browser vendors can understand them, without any ambiguities?

For each element, you need to create a definition that explains what that element represents, or in other words, you need to describe the semantics of the element. You cannot put elements in the markup arbitrarily, they all have a well-defined context where they can be put in.

For example, the `<title>` element can be placed only within the `<head>` element. You also need to define the content model of an element and describe what other elements it can embed. Most markup elements have attributes, so you need to explain their usage as well.

The object model – a solution?

The object model - a solution

As you can see, the description of markup elements needs a kind of **object model**. The browser also needs this model to render an HTML page based on the behavior or definition of the fundamental elements.

The creators of HTML5 put the Document Object Model into the standard as its backbone, as a **ubiquitous language** for standard designers and implementers.

> 📜**NOTE:** The term “ubiquitous language” comes from the world of modeling. It is used to characterize a common language used by business analysts and developers in the same way to avoid ambiguities. The main virtue of a ubiquitous language is that it avoids the abstraction gap that would otherwise exist between the domain language used by analysts and the language used by developers.

In the *next lesson*, we’ll discover the cornerstone of interactions.

Stay tuned! :)