

Front End Development

Tom Owens



# DOM Manipulation

SESSION TYPE : LESSON  
WBD4300

DOM Manipulation

Window Properties

Events

Web Design Prototyping

Workflow

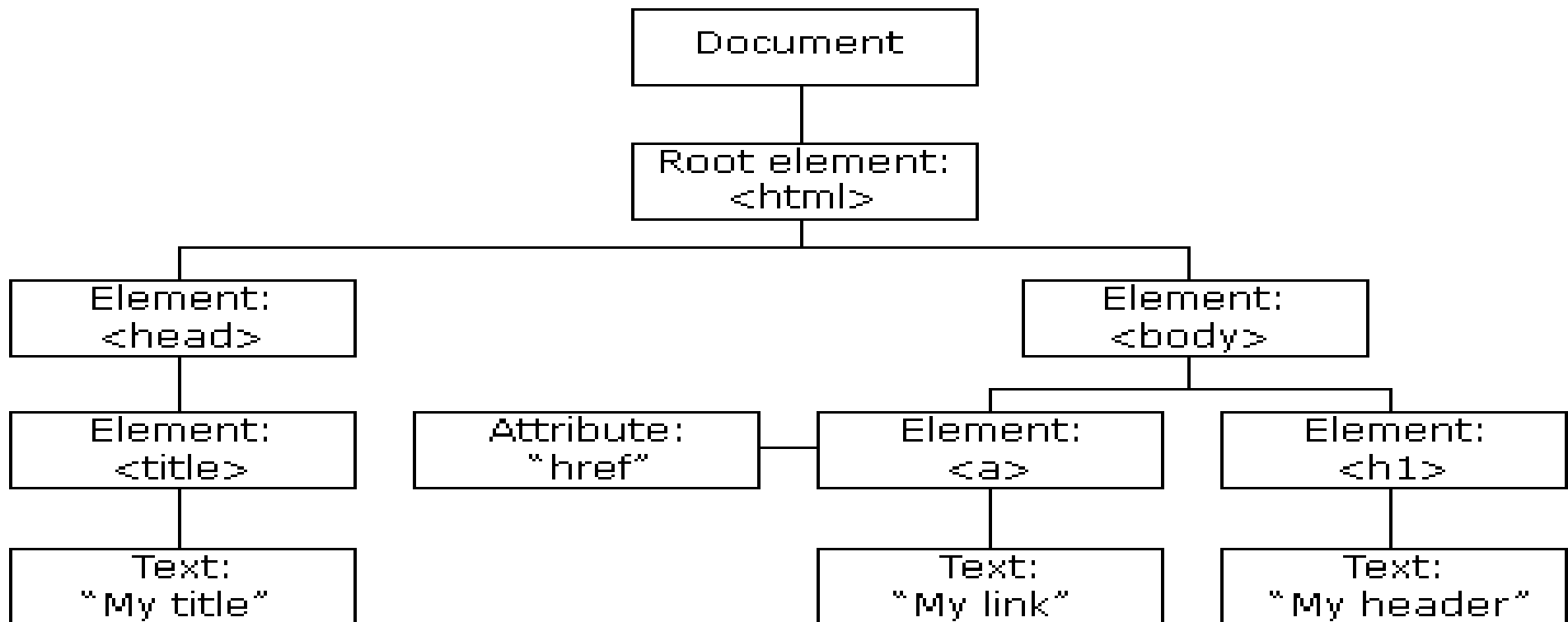
Wire framing/ Mock-ups

Self Directed: DOM Manipulation

# DOM Manipulation

"The W3C Document Object Model (DOM) is a platform that allows scripts to dynamically access the content, structure, and style of a document."

The HTML DOM defines a standard way for accessing and manipulating HTML documents



# DOM Manipulation

## Window Properties

It represents the browser's window. All global JavaScript objects, functions, and variables automatically become members of the **window object**.

Global variables are **properties** of the window object.

Global functions are **methods** of the window object.

## Events

events are often triggered by the end user's interaction with the page, such as when text is entered into a form element or the mouse pointer is moved.



# Web design Prototyping/workflow

## Web Design Prototyping

A website prototype can be any mock-up or demo of what a website will look like when it goes live.

It can be anything from a paper sketch, to a clickable HTML prototype.

There are a variety of different ways to create an interactive prototype.

## Workflow

the sequence of industrial, administrative, or other processes through which a piece of work passes from initiation to completion.

# Web design Prototyping/workflow

## Wire-framing

A website wireframe, also known as a page schematic or screen blueprint, is a visual guide that represents the skeletal framework of a website.

Wireframes are created for the purpose of arranging elements to best accomplish a particular purpose.

## Mock-ups

A mock-up is a prototype if it provides at least part of the functionality of a system and enables testing of a design.

# Web design Prototyping/workflow

## Web Design Prototyping (DOM Handling)

After mock-ups and wireframes are made, we need a place where testing and trial and error can take place for DOM manipulation. This would be a digital and interactive prototype.

We can use desktop applications such as Brackets.

Web based platforms such as Code Pen and JS fiddle.

## Workflow (DOM Handling)

The workflow of DOM Handling will take place in the development stage or prototyping of a project. You will have some idea during visualization, how the functionality will be performed. This forms part of the project specification.

# Workflow: Contextual Design

Developed by:

Hugh Beyer and Karen Holtzblatt

Definition: A way of developing a project based on UX and research centred workflow.

Interpretation → Answer to brief

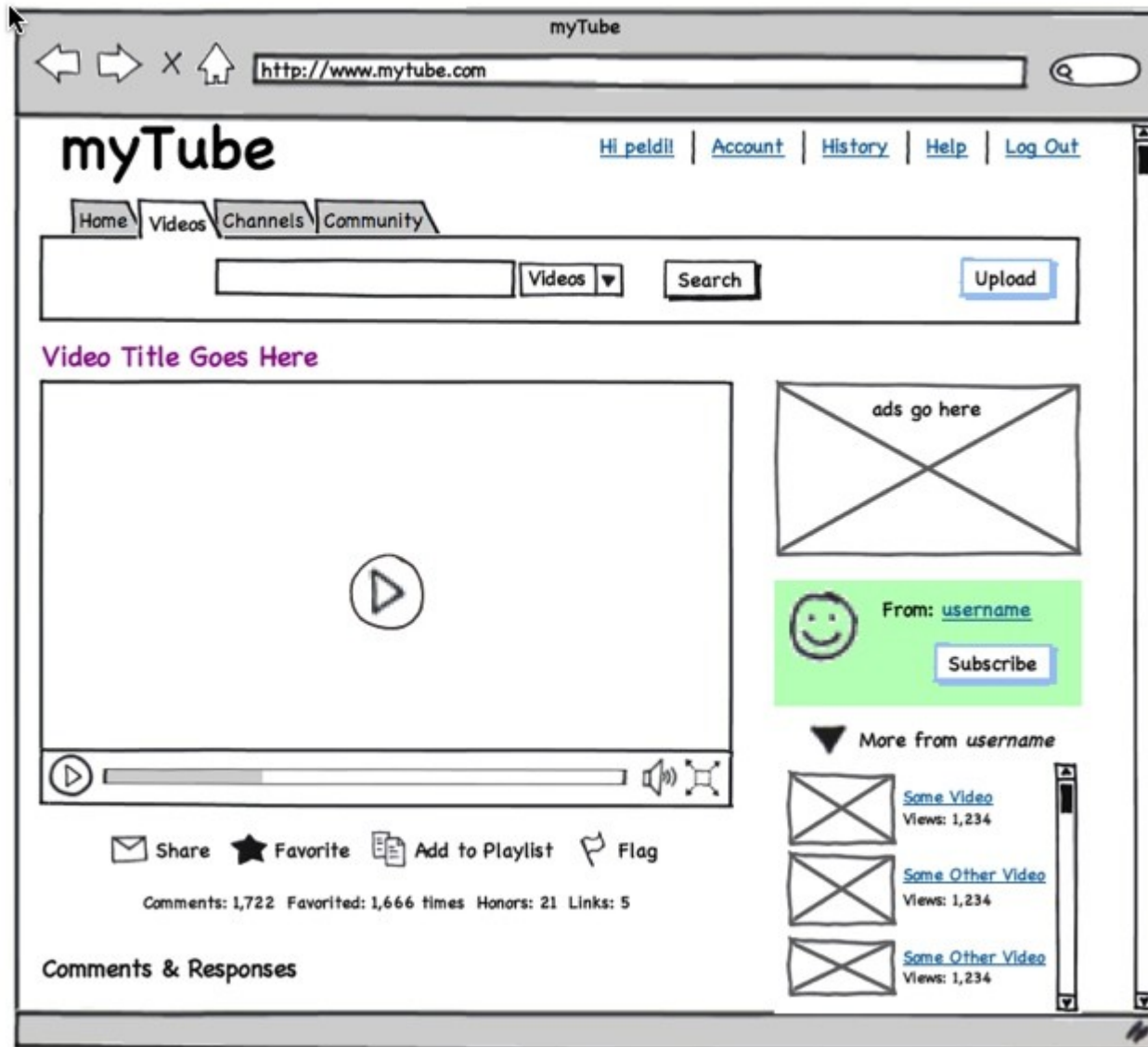
Data Compilation → Demographics and SEO

Visualization → Design Led, Mock-ups

Storyboarding → Mock-ups and Wireframes.

Prototyping → Mock-ups and working models.





# Self Directed: DOM Manipulation

Let's use w3schools and practice DOM Manipulation.

Then I will contrive an applied practical workshop

# Group Tutorial: DOM Manipulation (JS)

**The text below has been created dynamically.**

I just rolled on this button

I just rolled on this button

I just rolled on this button

Roll Over Me

We're going to try the example of an interactive exercise and prototype of DOM Manipulation. Not only will we learn the principal of DOM and Creating an Element but a working and interactive example of which, which, as students, you have asked for in order to understand better the purpose of DOM manipulation.

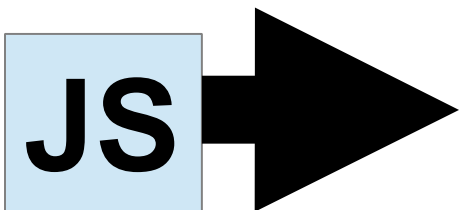
# Group Tutorial: DOM Manipulation (JS)

## HTML

```
<div id="div1">  
  <h1>The text below has been created dynamically.</h1>  
</div>  
<button onmouseover=addElement();>Roll Over Me</button>
```

## CSS

```
div {  
  border: 1px solid red;  
  margin: 5px;  
  padding: 5px;  
}
```



# Group Tutorial: DOM Manipulation (JS)

**JS**

```
window.onload = addElement;
```

```
function addElement() {  
  // create a new div element and give it some content  
  var newDiv = document.createElement("div");  
  var newContent = document.createTextNode("I rolled on this");  
  // add the text node to the newly created div.  
  newDiv.appendChild(newContent);  
  // add the newly created element and its content into the DOM  
  var currentDiv = document.getElementById("div1");  
  // document.body.insertBefore(newDiv, currentDiv);  
  currentDiv.appendChild(newDiv);  
}
```

```
console.log('hello console.');
```

# Advanced Validation Form with JS

OK, we are going to conduct a group tutorial on copying step by step a set of coded pages in HTML and JS.

The result will be a working and advanced Validation Form. An example of the completed code will accompany this slide show on SAE Campus Online.

After this we will attempt customization and discuss what is happening with the code.