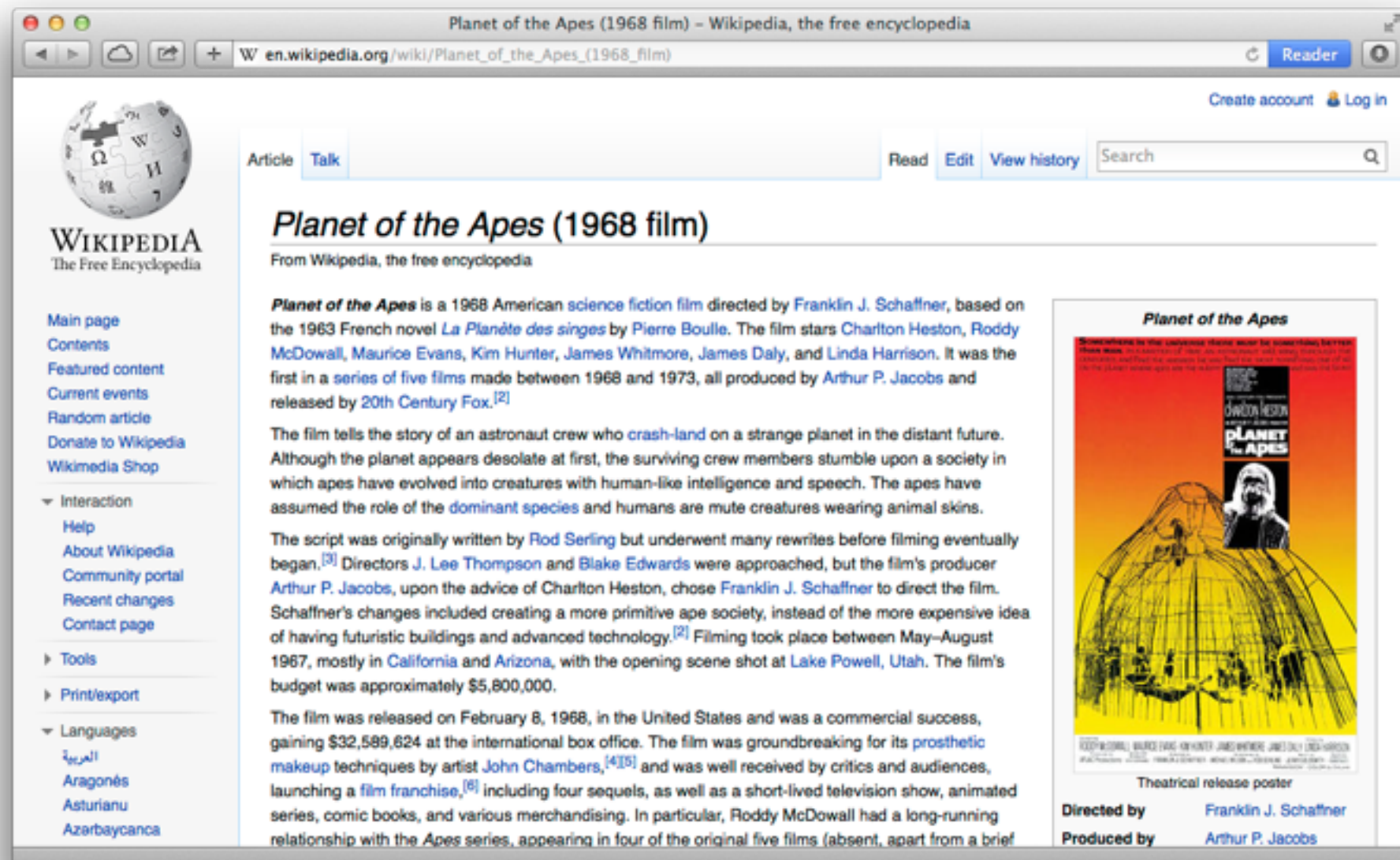
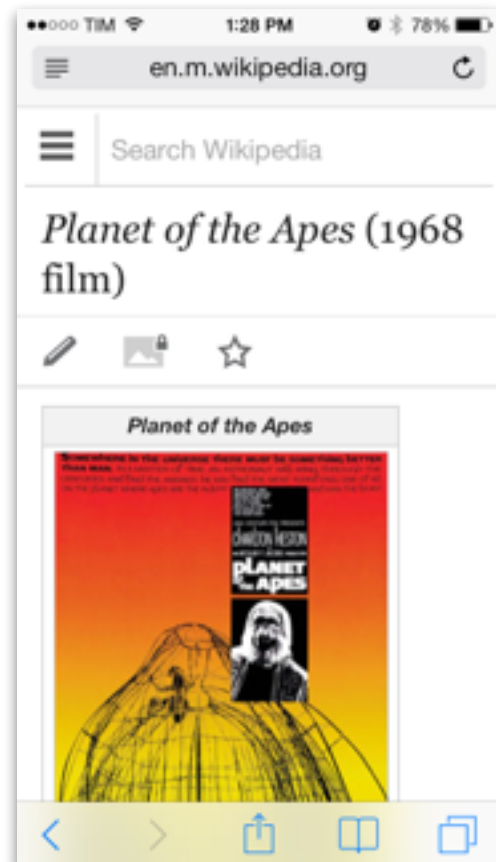
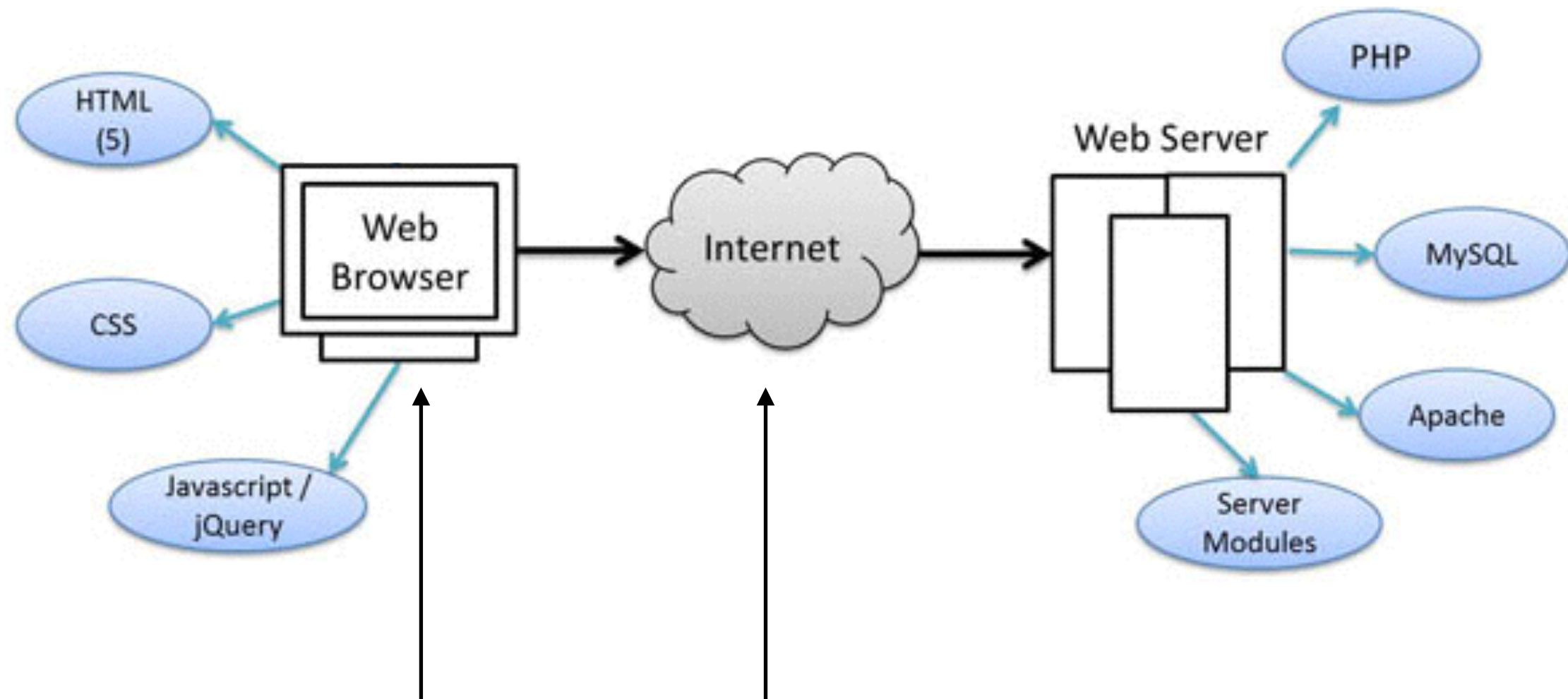


js();
Javascript



What is behind?



[Planet of the Apes \(1968 film\) - Wikipedia, the free encyclopedia en.wikipedia.org/wiki/Planet_of_the_Apes_\(1968_film\)](https://en.wikipedia.org/wiki/Planet_of_the_Apes_(1968_film)) ▼

Planet of the Apes is a 1968 American science fiction film directed by Franklin J. Schaffner, based on the 1963 French novel *La Planète des singes* by Pierre ...

[Beneath the Planet of the Apes](#) - [Linda Harrison](#) - [Kim Hunter](#) - [Maurice Evans](#)

crawlers and other agents

Web page

HTML

```
<html>
<head>...
<body>
<h1>
<h2>
<p>...</p>
<ol><li><li>
```

Content

- Headings
- Paragraphs
- Lists
- Images
- Links

CSS

```
<style>
Body {
  Color:...
ul#mylist {
  font-family
</style>
```

Presentation

- Colors
- Fonts
- Positioning

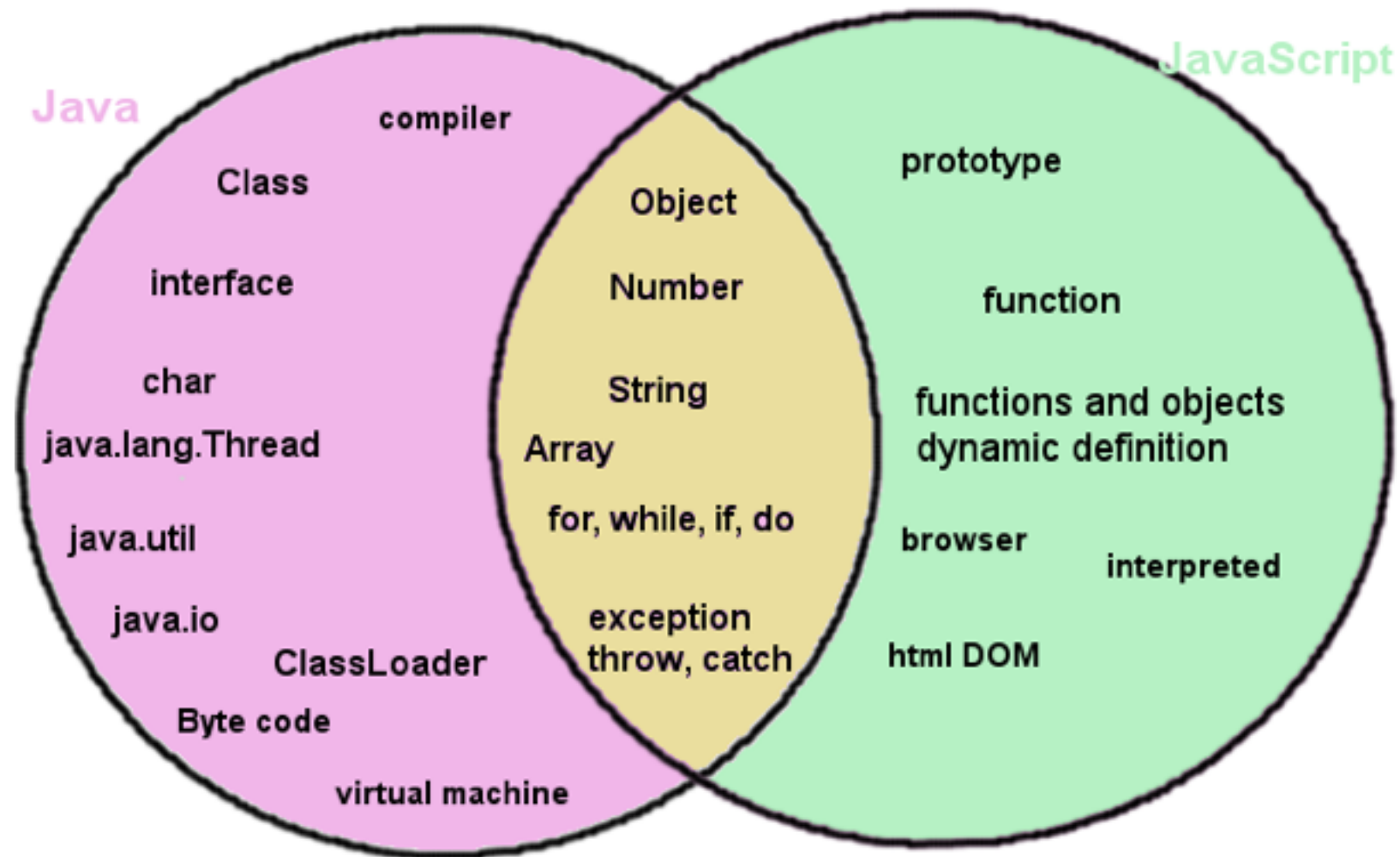
JavaScript

```
<script type=
"text/javascript"
```

Behavior

- AJAX data manipulation
- Error checking
- pop-up calendars
- special effects

Javascript



Hands On!

Reference material



The screenshot shows the 'JavaScript Beginner Tutorial' page from the HTML Dog website. The page has a dark sidebar on the left with a navigation menu. The main content area has a dark header with the title 'JavaScript Beginner Tutorial' and an introductory paragraph. Below the header, there is a light gray warning box with a yellow warning icon and text. At the bottom of the main content area, there is a 'Contents' section with a list of links to various tutorial topics.

HTML Dog

HTML
CSS
JavaScript
Beginner
Making Stuff Happen
Variables and Data
Doing Math
Logic
Conditional
Looping
Functions
Objects
Arrays
Intermediate
Advanced
Techniques
Examples
References
HTML Tags
CSS Properties

JavaScript Beginner Tutorial

As with the [HTML Beginner Tutorial](#) and [CSS Beginner Tutorial](#), the **JavaScript Beginner Tutorial** assumes that you know zip all about JavaScript. The purpose of this guide is to teach the bare essentials - just enough to get started. The [JavaScript Intermediate Tutorial](#) and [JavaScript Advanced Tutorial](#) go into more depth.

This tutorial is an introduction to programming in general, through the medium of JavaScript. We'll get into interacting with web pages in the [Intermediate section](#).

As thrilling as it is to jump in at the deep end, these waters contain rabid sharks AND angry piranhas. That's right; salt water fish AND fresh water fish all in one swimming pool.

For your own safety, **you should be as comfortable as a cushion with HTML and CSS before proceeding**. Don't say we didn't warn you...

Contents

- [Making Stuff Happen](#): Applying JavaScript and using the console.
- [Variables and Data](#): How to create variables and how you use them. Strings!
- [Doing Math](#): Using variables to store and use numbers.
- [Logic](#): True and false.
- [Conditional](#): If and but... no, or else... oh no I'm confused now.
- [Looping](#): While do for while do for while do for... argumentum ad nauseam.
- [Functions](#): Creating functions, using them, passing data in and out.
- [Objects](#): Properties, methods. Inspecting objects.
- [Arrays](#): Getting elements out, putting them back in. How long is a piece of string?

<http://htmldog.com/guides/javascript/beginner/>

Variables and data types

var name;
name = value;

dynamic typing

```
name = "Pippo";  
name = 5;  
name = true;  
name = [1,2,3];  
name = {};  
name = function(){  
};
```


Conditional

```
if ([condition]){  
    /* true block */  
}
```

```
if ([condition]){  
    /* true block */  
} else {  
    /* true block */  
}
```

```
if ([condition1]){  
    /* true condition 1 */  
} else if ([condition2]) {  
    /* true condition 2 */  
} else {  
    /* false both */  
}
```

Loops

```
while ([condition]){  
    /* true */  
}
```

```
do {  
    /* loop block */  
} while ([condition]);
```

```
for(init; [condition1], inc){  
    /* true condition */  
}
```

functions

```
function name(param) {  
    return value;  
}
```

```
var name = function (param) {  
    return value;  
}
```

Objects & Arrays

```
var person = {  
  name : value,  
  age : value,  
  speak : function(){  
    return sth;  
  }  
}
```

```
person.name  
person.speak()  
person["name"]
```

```
var list = [val1, val1, val3]
```

```
list.length
```

```
list[index]
```

```
list.push(val)
```

```
list.pop(val)
```