Summary for Intro to HTML and Jquery (Well, I kinda got realllly lazy)

Introduction to HTML:

Summary:

The foundations of any webpage is created using HTML (HypterText Markup Langauge), which is what the browser "reads" to present the page on the client side computer. HTML uses a some pre-defined set of elements to identify content types, which then contain tags used to contain or express content. These tags can contain attributes that appears in the start tag to contain additional information. Otherwise, the basic outline of any HTML page consists of the <html>, <body>, some content inside, and closing all of those elements.

Notes:

Several techonologies (CSS, JS, Flash, AJAX, JSON) can be used to define the elements of a webpage. However, the foundations of any webpage is created using HTML (HypterText Markup Language). This is what the browser "reads" to present the page on the client-side computer.

What is HTML?

HTML is a markup language: A markup language as it relates to browsers is a language with specific syntax that gives instructions to a web browser about how to display a page.

HTML uses a pre-defined set of elements to identify content types. Elements contain one or more "tags" that contain or express content. Tags are surrounded by angle brackets, and the "closing" tag (the one that indicates the end of the content) is prefixed by a forward slash.

Basic structure is:

```
<html>
        <body>
                <p>
        </body>
</html>
```

Tags

HTML uses a pre-defined set of elements to identify content types. Elements contain one or more "tags" that contain or express content. Tags are surrounded by angle brackets, and the "closing" tag (the one that indicates the end of the content) is prefixed by a forward slash.

… a couple more stuffs but otherwise, I pretty much know HTML decently well and it's quite simple to catch on, so do yourself a favor and look it up.

https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Introduction

Jquery Basics:

Summary:

Jquery is more or less used to manipulate a page of HTML after it has been displayed by the browser. The $ symbol is shortcut for the jQuery function which can be used for a variety of things. When you call the $() function and pass a selector to it, it then creates a new jQuery object. The $() function can also act as an alias for $(document).ready() which is used to ensure that the page is in a state where it's ready to be manipulated. And the $ functions last role is creating new elements when you pass some HTML snippet to $(), which would then create a new element in memory.

Notes:

The jQuery library makes it easy to manipulate a page of HTML after it's displayed by the browser. It also provides tools that help you listen for a user to interact with your page, tools that help you create animations in your page, and tools that let you communicate with a server without reloading the page. The $ is a shorter, more convenient name for the jQuery function.

When you call the $() function and pass a selector to it, you create a new jQuery object. Of course, in JavaScript, functions are objects too, so that means that $ (and jQuery, of course) has properties and methods, too.

Before you can safely use jQuery to do anything to your page, you need to ensure that the page is in a state where it's ready to be manipulated. With jQuery, we accomplish this by putting our code in a function, and then passing that function to $(document).ready(). As you can see here, the function we pass can just be an anonymous function.

```
$( document ).ready(function() {
  console.log( 'ready!' );
});
```

of you can write the shortcut form which is:

```
$(function() {
  console.log( 'ready!' );
});
```

The $ has one more role of creating new elements. If you pass an HTML snippet to $(), it will create a new element in memory.

You can determine whether a selection meets certain criteria using .is() method, which returns true or false.

Implicit iteration means that jQuery automatically iterates over all the elements in a selection when you call a setter method on that selection. This means that, when you want to do something to all of the elements in a selection, you don't have to call a setter method on every item in your selection — you just call the method on the selection itself, and jQuery iterates over the elements for you.

One of the most lucrative parts of jQuery is the ability to "chain" methods together. This means that we can call a series of methods on a selection without having to repeat the selection or store the selection in a variable. We can even make new selections based on previous selections, all without breaking the chain.

Chaining is possible because every setter method in jQuery returns the selection on which it was called. It's extraordinarily powerful, and it's a feature that many libraries have adopted.