

JQUERY

jQuery is a "Write Less, Do More" JavaScript library. A library is a tool (a collection of simplified commands) used to make it easier to code. It is not a full programming language.

jQuery is a concise JavaScript Library with syntax similar to CSS.

jQuery greatly simplifies JavaScript programming.

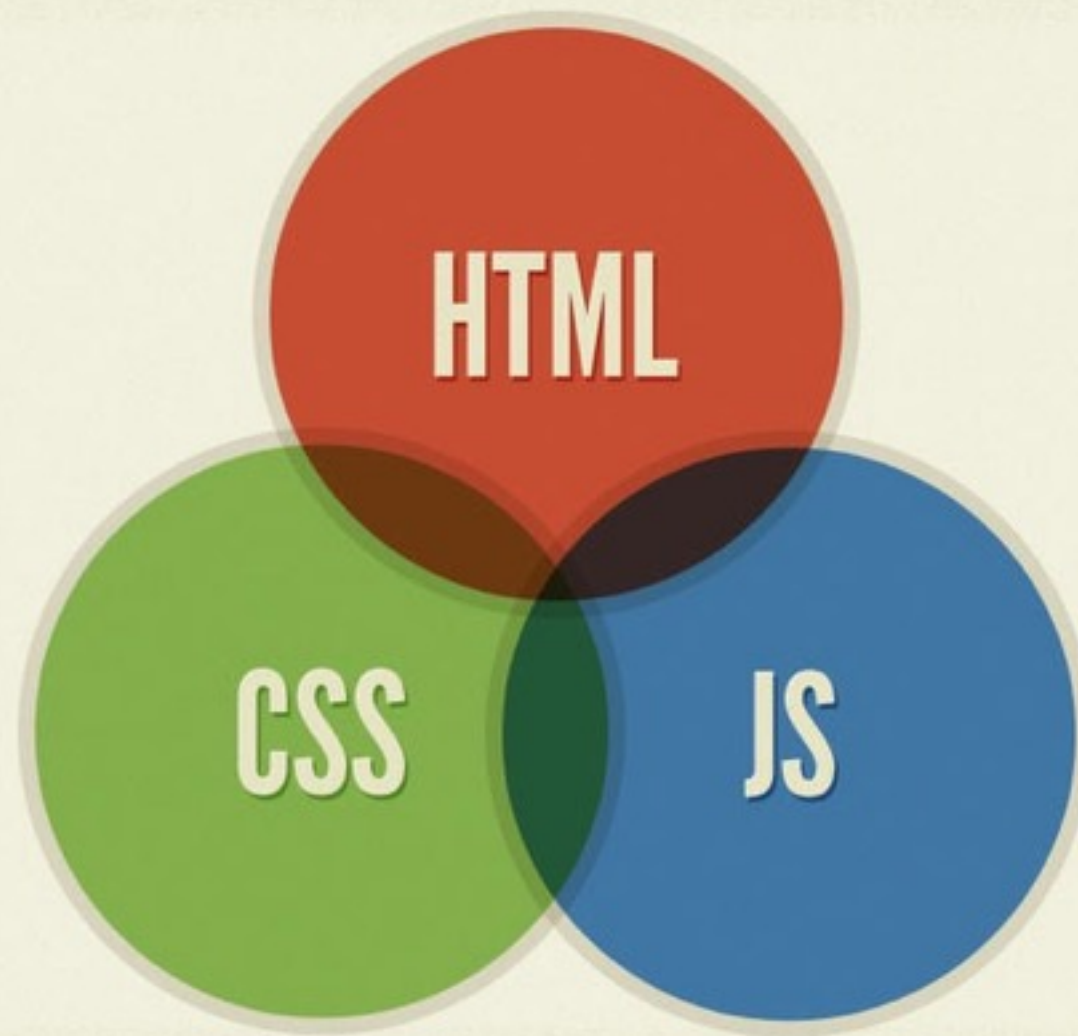
jQuery is easy to learn.

HTML is for your content.

CSS is for styling.

jQuery (and Javascript) are used to define behaviors and events.

The Three Amigos



jQuery is often used for ***events***

For example,
when you click a button, start
an animation.

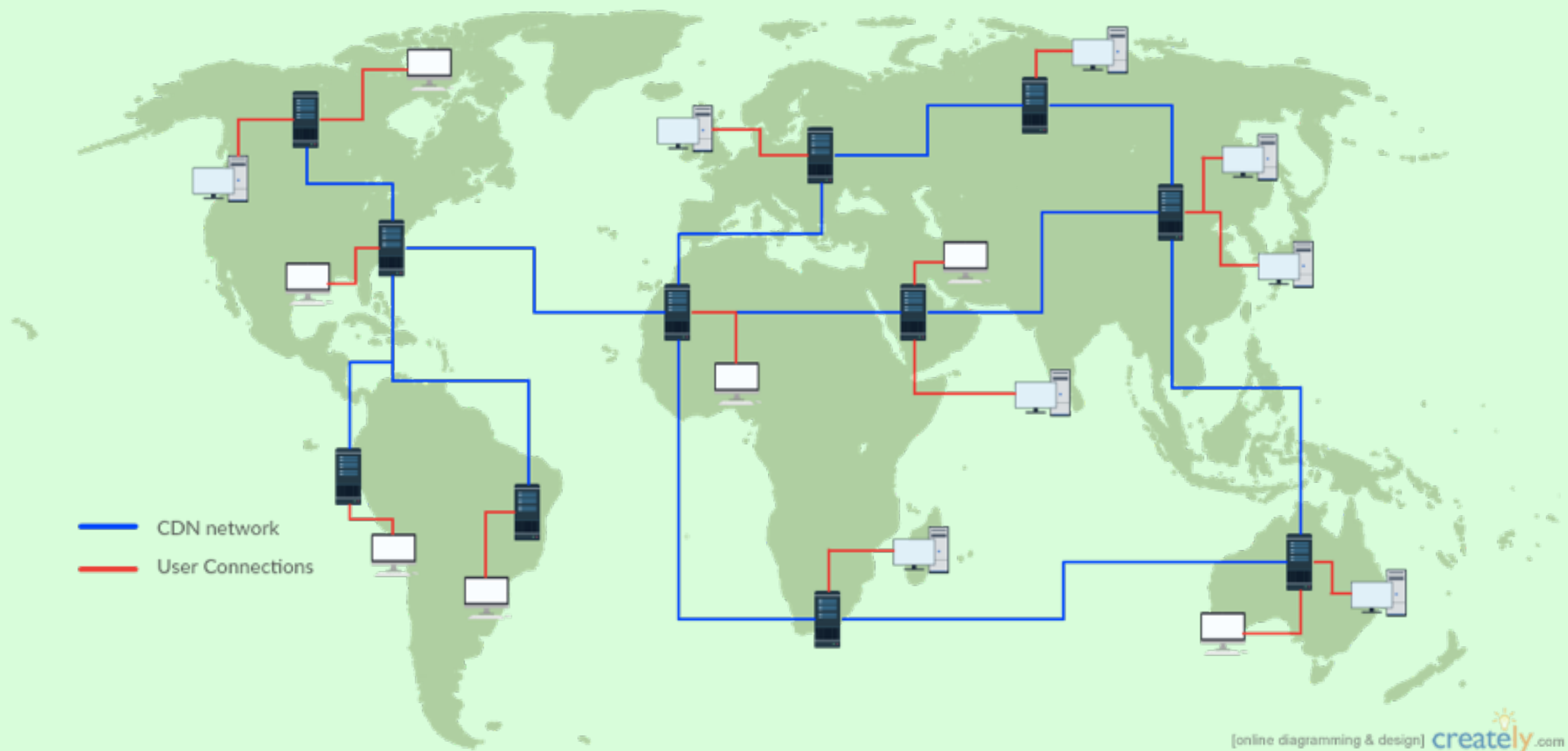
When the page loads, fade in
a specific photo.

How to use jQuery?

2 ways to add its functionality to your page.

1. Download a local copy.
2. Use a CDN

A Content Delivery Network (CDN) is a system of multiple servers that deliver web content to a user based on geographical location. When you link to a hosted jQuery file via CDN, it will potentially arrive faster and more efficiently to the user than if you hosted it on your own server.




```
<script src="https://code.jquery.com/jquery-3.3.1.js"></script>
```

Typical project layout

```
project/  
├── css/  
│   └── style.css  
├── js/  
│   └── scripts.js  
└── index.html
```

Basic HTML starter

index.html

```
<!doctype html>
<html lang="en">

<head>
  <title>jQuery Demo</title>
  <link rel="stylesheet" href="css/style.css">
</head>

<body>
</body>

</html>
```

Link to the jQuery CDN right before the closing `</body>` tag, followed by your own custom JS file.

We do this because we want to load our HTML content and style first before running our script.

index.html

```
<!doctype html>
<html lang="en">

<head>
  <title>jQuery Demo</title>
  <link rel="stylesheet" href="css/style.css">
</head>

<body>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.0/jquery.min.js"></script>
<script src="js/scripts.js"></script>
</body>

</html>
```

Note: If you downloaded a local copy of jQuery (instead of using a CDN), save it in your js/ folder and link to it at
js/jquery.min.js

You must link with a <script> tag to jQuery on your server or at a CDN's URL **before** you link to your own script.

DOM

Document Object Model (DOM) is the method by which JavaScript (and jQuery) interact with the HTML in a browser. To view exactly what the DOM is, in your web browser, right click on the current web page and select Inspect. This will open up Developer Tools. The HTML code you see here is the DOM.

Each HTML element is considered a node in the DOM - an object that JavaScript can touch. These objects are arranged in a tree structure, with `<html>` being closer to the root, and each nested element being a branch further along the tree. JavaScript and jQuery can add, remove, and change any of these elements.

If you right click on the site again and click View Page Source, you will see the raw HTML output of the website. It's easy at first to confuse the DOM with the HTML source, but they're different - the page source is exactly what is written in the HTML file. It is static and will not change, and will not be affected by JavaScript. The DOM is dynamic, and can change.

The outermost layer of the DOM, the layer that wraps the entire `<html>` node, is the document object. To begin manipulating the page with jQuery, we need to ensure the web page "document " is ready (loaded) first.

js/scripts.js

```
$(document).ready(function() {  
    // all custom jQuery will go here  
});
```

jQuery is called with and represented by the dollar sign (\$). We access the DOM with jQuery using mostly CSS syntax, and apply an action with a method. A basic jQuery example follows this format.

```
$( "selector" ).method( );
```

Below is a brief overview of some of the most commonly used selectors.

`$("*")` - Wildcard: selects every element on the page.

`$(this)` - Current: selects the current element being operated on within a function.

`$("p")` - Tag: selects every instance of the `<p>` tag.

`$(".example")` - Class: selects every element that has the example class applied to it.

`$("#example")` - Id: selects a single instance of the unique example id.

`$("[type='text']")` - Attribute: selects any element with text applied to the type attribute.

`$("p:first-of-type")` - Pseudo Element: selects the first `<p>`.

Generally, classes and ids are what you will encounter the most – classes when you want to select multiple elements, and ids when you want to select only one.