WEB DEVLOPMENT(3151606)

UNIT-7:ADVANCED CONCEPTS &JQUERY

PREPARED BY: PROF. KRUPALI. DAVE

ADVANCED CONCEPTS:

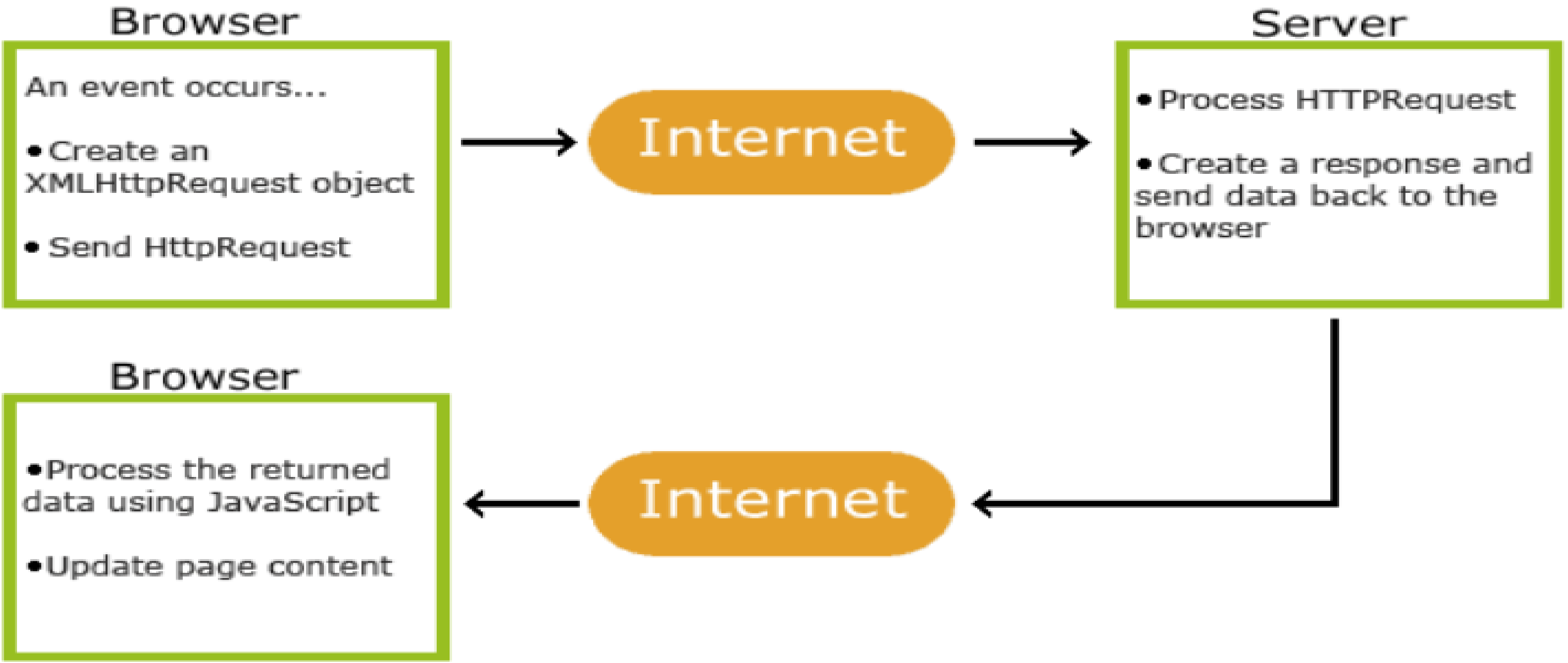
- AJAX, which **stands for asynchronous JavaScript and XML**, is a technique that allows web pages to be updated asynchronously, which means that the browser doesn't need to reload the entire page when only a small bit of data on the page has changed. AJAX passes only the updated information to and from the server.
- Standard web applications process interactions between web visitors and the server synchronously. This means that one thing happens after another; the server does not multitask. If you click a button, the message is sent to the server, and the response is returned. You cannot interact with any other page elements until the response is received and the page is updated.
- Obviously, this kind of delay can negatively affect a web visitor's experience — hence, AJAX.

WHAT IS AJAX?

- The term was **coined back in 2005 by Jesse James Garrett**. As the Mozilla Developer Network explains, Ajax "describes a 'new' approach to using a number of existing technologies together, including HTML or XHTML, Cascading Style Sheets, JavaScript, The Document Object Model, XML, XSLT, and most importantly the XMLHttpRequest object."

- AJAX is not a programming language or new technology, but a technique that incorporates a client-side script (i.e. a script that runs in a user's browser) that communicates with a web server. Further, its name is somewhat misleading: while an AJAX application might use XML to send data, it could also use just plain text or JSON text. But generally, it uses an XMLHttpRequest object in your browser to request data from the server and JavaScript to display the data.
- JavaScript Object Notation (**JSON**) is a standard **text**-based format for representing structured data based on JavaScript object syntax. It is commonly used for transmitting data in web applications

How AJAX Works



Browser

An event occurs...

- Create an XMLHttpRequest object
- Send HttpRequest

Internet

Server

- Process HTTPRequest
- Create a response and send data back to the browser

Internet

Browser

- Process the returned data using JavaScript
- Update page content

➢ Essentially, it serves as a bridge between database and server without requiring the user to refresh the page. **It is a better way to create faster, more responsive, and better web-based applications using HTML, Java Script, XML or JSON, and CSS**. Using Ajax, a page is able to request things from the server, process the results, and then update the page – all of which happens without the user sensing what's going on or having to take additional actions.

WHERE IS AJAX USED?

- Ajax is commonly used in Web applications where small tidbits of information are retrieved or saved without needing to reload an entire page. Take for example saving a comment on a message board. Ajax is also used on auto-complete boxes and text hints, where you can type the first letters and the system would try to guess what you are typing and suggest words or phrases for you.

- Note that while the "X" in Ajax represents XML, JSON is more commonly used today as it's both lightweight and a part of JavaScript. However, both JSON and XML can be used to package information in the Ajax model. In addition to JSON and XML, data can also be transported as plain text.

AJAX: Synchronous or Asynchronous

- AJAX can access the server both synchronously and asynchronously:

1. **Synchronously**, in which the script stops and waits for the server to send back a reply before continuing.
2. **Asynchronously**, in which the script allows the page to continue to be processed and handles the reply if and when it arrives.

Synchronous:

- Processing your request ***synchronously*** is similar to reloading the page, but only the requested information is downloaded instead of the entire page. Therefore, using AJAX synchronously is faster than not using it at all — but it still requires your visitor to wait for the download to occur before any further interaction with the page can proceed. People know that they sometimes need to wait for a page to load, but most folks are not used to continued, significant delays after they are on a site.

Asynchronous:

- Processing your request ***asynchronously*** avoids the delay while the retrieval from the server takes place because your visitor can continue to interact with the web page; the requested information will be processed in the background and the response will update the page as and when it arrives. Further, even if a response is delayed — for example, in the case of very large data — site visitors may not realize it because they are occupied elsewhere on the page.

- **Therefore, the preferred way to use AJAX is to use asynchronous calls wherever possible. This is the default setting in AJAX.**

JQUERY:

- jQuery is a fast and concise JavaScript Library created by John Resig in 2006 with a nice motto: **Write less, do more**
- jQuery is a JavaScript toolkit designed to simplify various tasks by writing less code.
- **jQuery** is an open source JavaScript library that simplifies the interactions between an HTML/CSS document, or more precisely the Document Object Model (DOM), and JavaScript.
- Elaborating the terms, jQuery simplifies HTML document traversing and manipulation, browser event handling, DOM animations, Ajax interactions, and cross-browser JavaScript development.
- **Note:** The only library available today that meets the needs of both designer types and programmer types is jQuery.

WHY JQUERY?

- Some of the key points which supports the answer for why to use jQuery:

1. It is incredibly popular, which is to say it has a large community of users and a healthy amount of contributors who participate as developers and evangelists.
2. It normalizes the differences between web browsers so that you don't have to.
3. It is intentionally a lightweight footprint with a simple yet clever plugin architecture.
4. Its repository of plugins is vast and has seen steady growth since jQuery's release.
5. Its API is fully documented, including inline code examples, which in the world of JavaScript libraries is a luxury. Heck, any documentation at all was a luxury for years.
6. It is friendly, which is to say it provides helpful ways to avoid conflicts with other JavaScript libraries.

CONT..

- jQuery is widely famous with its philosophy of **“Write less, do more.”** This philosophy can be further elaborated as three concepts:

1. Finding some elements (via CSS selectors) and doing something with them (via jQuery methods) i.e. locate a set of elements in the DOM, and then do something with that set of elements.
2. Chaining multiple jQuery methods on a set of elements
3. Using the jQuery wrapper and implicit iteration

HOW TO USE JQUERY?

- There are two ways to use jQuery.

- **Local Installation** – You can download jQuery library on your local machine and include it in your HTML code.

- **CDN Based Version** – You can include jQuery library into your HTML code directly from Content Delivery Network (CDN).

1. Use the Google-hosted/ Microsoft-hosted content delivery network (CDN) to include a version of jQuery.
2. Download own version of jQuery from jQuery.com and host it on own server or local filesystem.

- **Note:**All jQuery methods are inside a document ready event to prevent any jQuery code from running before the document is finished loading (is ready).

BASIC SYNTAX OF JQUERY:

- **Basic syntax for any jQuery function is:**

$(selector).action()

- A $ sign is to define/access jQuery
- A (selector) is to “query (or find)” HTML elements in html page
- A jQuery action() is the action to be performed on the selected element(s)

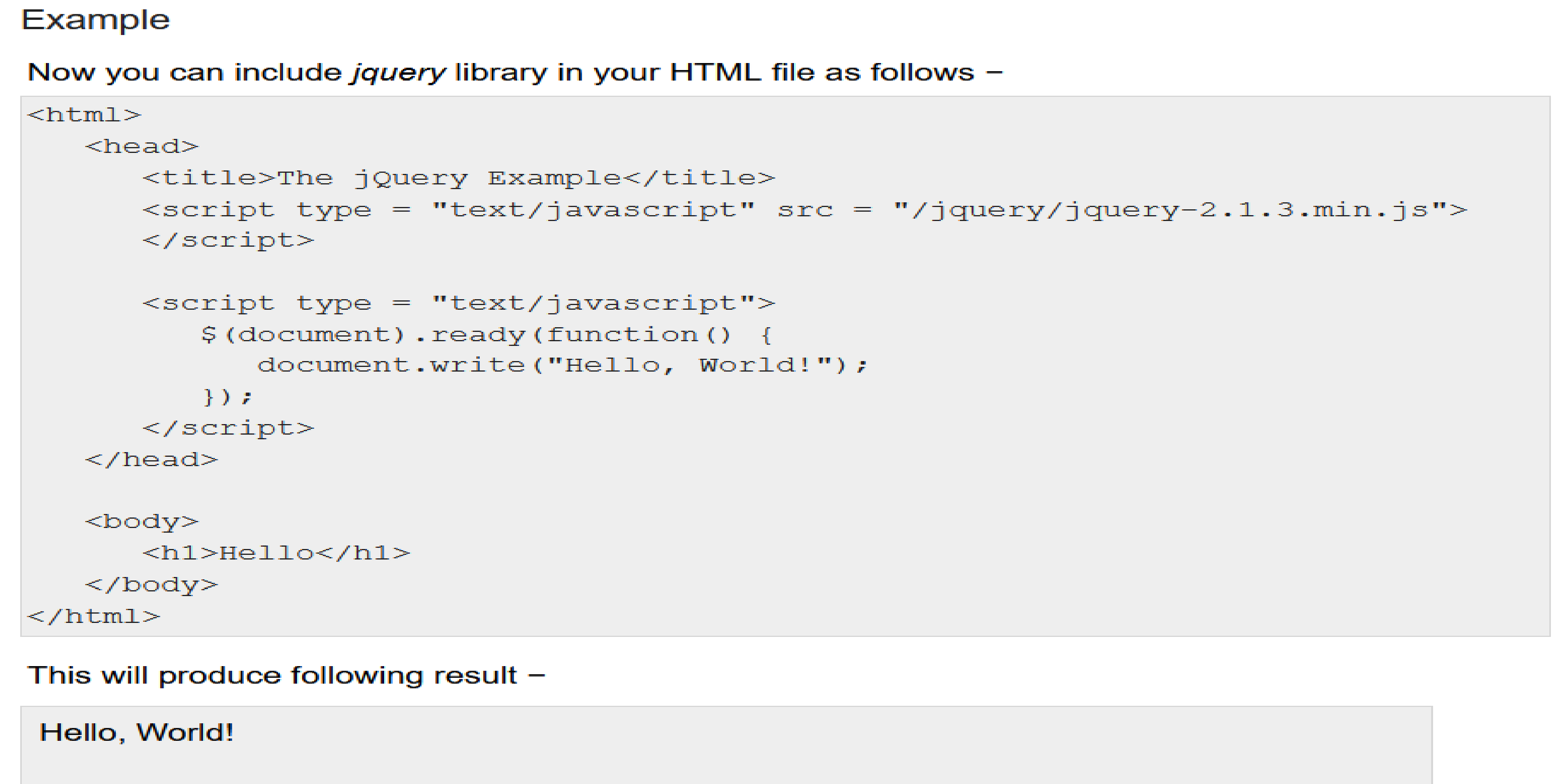
```
EX:  $(document).ready(function(){
     $("button").click(function(){
          $(".gfg").hide();
     };)
});
```

- This example explain a case where a user clicks on a button and the elements

with class = “gfg” will be hidden as an effect of that.

LOAD INSTALLATION:

- Go to the https://jquery.com/download/ to download the latest version available.
- Now put downloaded **jquery-2.1.3.min.js** file in a directory of your website, e.g. /jquery.



Example

Now you can include *jquery* library in your HTML file as follows −

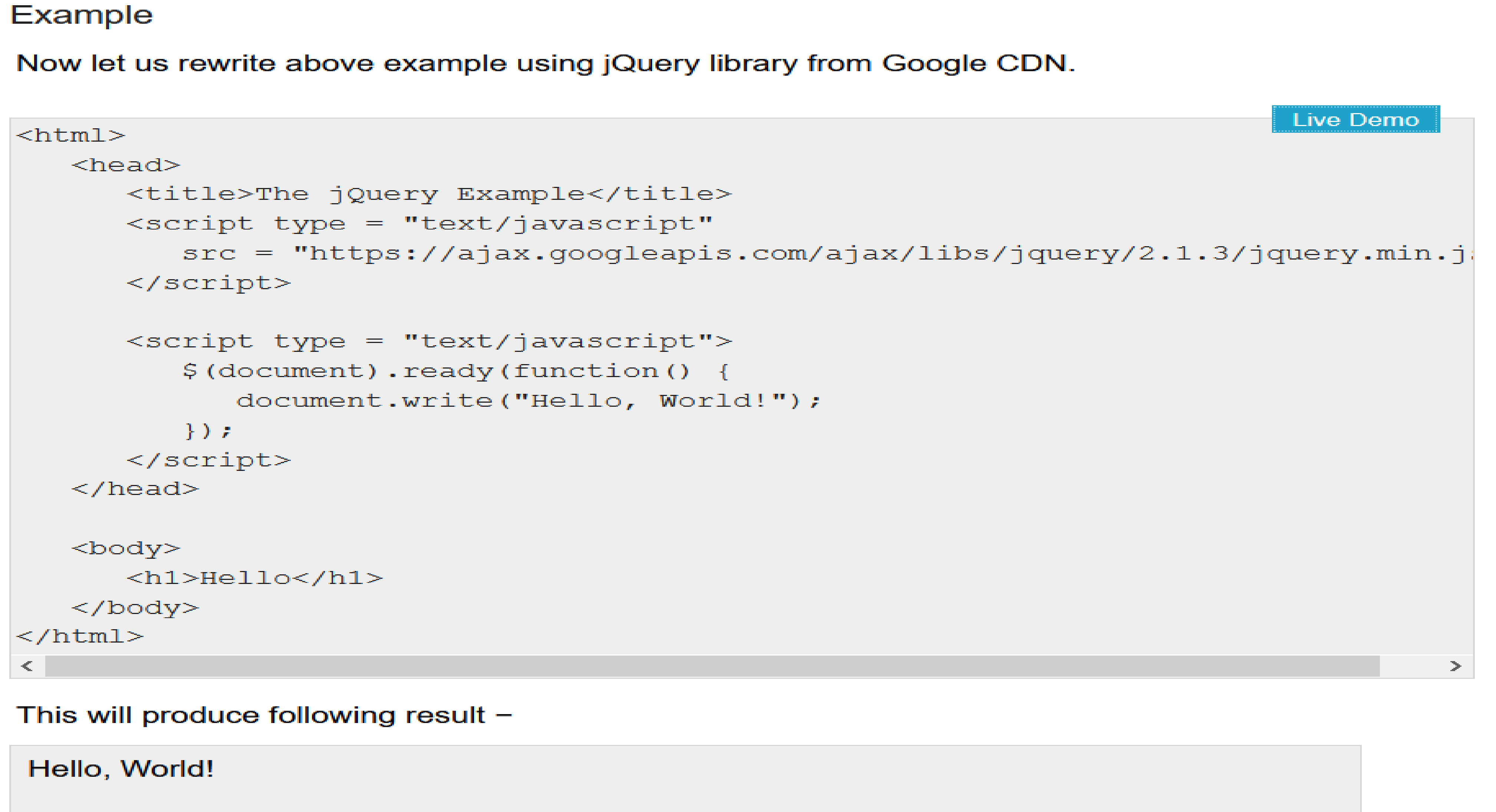
```
<html>
   <head>
      <title>The jQuery Example</title>
      <script type = "text/javascript" src = "/jquery/jquery-2.1.3.min.js">
      </script>

      <script type = "text/javascript">
         $(document).ready(function() {
            document.write("Hello, World!");
         });
      </script>
   </head>

   <body>
      <h1>Hello</h1>
   </body>
</html>
```

This will produce following result −

```
Hello, World!
```

CDN BASED VERSION:

- You can include jQuery library into your HTML code directly from Content Delivery Network (CDN). Google and Microsoft provides content deliver for the latest version.



Example

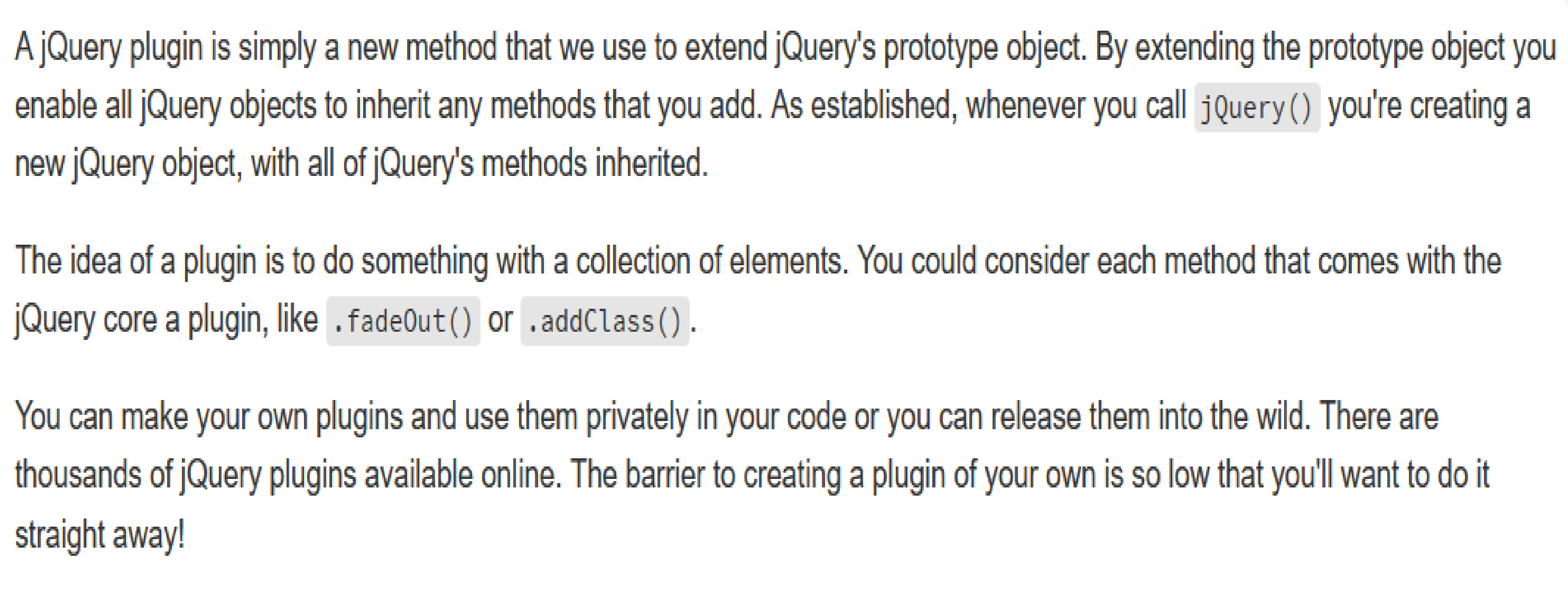
Now let us rewrite above example using jQuery library from Google CDN.

```
<html>
   <head>
      <title>The jQuery Example</title>
      <script type = "text/javascript"
         src = "https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.j
      </script>

      <script type = "text/javascript">
         $(document).ready(function() {
            document.write("Hello, World!");
         });
      </script>
   </head>

   <body>
      <h1>Hello</h1>
   </body>
</html>
```

This will produce following result –

```
Hello, World!
```

Using plugins in jQuery and Creating Image slider:



A jQuery plugin is simply a new method that we use to extend jQuery's prototype object. By extending the prototype object you enable all jQuery objects to inherit any methods that you add. As established, whenever you call `jQuery()` you're creating a new jQuery object, with all of jQuery's methods inherited.

The idea of a plugin is to do something with a collection of elements. You could consider each method that comes with the jQuery core a plugin, like `.fadeOut()` or `.addClass()`.

You can make your own plugins and use them privately in your code or you can release them into the wild. There are thousands of jQuery plugins available online. The barrier to creating a plugin of your own is so low that you'll want to do it straight away!

- A plug-in is piece of code written in a standard JavaScript file. These files provide useful jQuery methods which can be used along with jQuery library methods.

How to use Plugins:

- To make a plug-in's methods available to us, we include plug-in file very similar to jQuery library file in the <head> of the document.
- We must ensure that it appears after the main jQuery source file, and before our custom JavaScript code.
- Following example shows how to include jquery.plug-in.js plugin –

```
<html>
 <head>
<title>The jQuery Example</title>
 <script type = "text/javascript" src = "https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js"> </script>
<script src = "jquery.plug-in.js" type = "text/javascript"></script>
<script src = "custom.js" type = "text/javascript"></script>
 <script type = "text/javascript" language = "javascript"> $(document).ready(function() { .......your custom code..... }); </script>
</head>
<body>............................. </body>
 </html>
```

Generating charts from data using 3rd Party Libs:

- we will talk about top 5 best open source JavaScript chart libraries. Every site and dashboard are incomplete without charts and graphs so it is very important to find the right chart library for our application. The following libraries will help you to create customizable and beautiful charts for your application.

1. **D3.js — Data-Driven Documents**

- D3.js is an open source JavaScript library used for manipulating documents based on user data. It is a powerful tool which gives life to the data with the help of HTML, SVG, and CSS. D3 allows developers to bind arbitrary data to a DOM and then it applies the data-driven transformation to the DOM.
- For Example: Consider an array of numbers, You can use it to generate an HTML table or you can use the same data to generate interactive bar or pie charts.

2. **Google Charts**

- Google Charts is an open source chart library which is powerful and very simple to use. It has many interactive charts to display and render live data. It has a rich chart gallery that include options like pie charts, bar charts, Scatter Charts, donut charts, etc. Moreover, various customization options are available with the charts.
- It also has charts like Histograms, timelines, trend lines, Sankey Diagrams, etc.
- Official Website: https://google-developers.appspot.com/chart/

3. Chart.js

- Chart.js is a community maintained chart library and by using it we can create responsive charts for your website. Using it we can generate mixed charts and it has great rendering capacity in modern browsers. It has very good documentation and samples available.
- Angular Chart is built on top of the Chart.js library, and for Angular projects it will be really easy to implement Angular charts.
- Official Website: http://www.chartjs.org/

4. Chartist.js

- Chartist.js is also an open source JavaScript library similar to Chart.js. It has rich and response charts available. Using it, we can generate SVG charts that are DPI independent. It has support for most of the modern browsers and good community support. It has support for some other technologies like Node, Angular, Jjava, Wordpress, Ember, React, and Meteor.
- Official Website: http://gionkunz.github.io/chartist-js/

5. n3-charts

- n3-charts is also an open source JavaScript chart library and it makes the life of the Angular developer easy. n3-charts is built on top of D3.js and AngularJS, hence it posses more powerful charts and it's easy to implement.
- Official Website: http://n3-charts.github.io/line-chart/#/home

THANK YOU…