

The background features a dark blue-to-purple gradient with abstract white and light blue circular patterns. These patterns include concentric circles, arcs, and degree markings (e.g., 40, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260) that suggest a technical or design theme.

AJAX 程式設計

AJAX PROGRAMMING

DESIGN

張家瑋 博士

助理教授

國立臺中科技大學資訊工程系

The image features a dark blue background with faint, light blue circular patterns and arrows. In the foreground, there are three stylized device silhouettes: a large desktop monitor on the left, a tablet in the center, and a smartphone on the right. The desktop monitor displays the text 'AJAX PROGRAMMING' in white, with the 'A' and 'P' highlighted in yellow. The tablet displays the letters 'A', 'J', 'A', and 'X' stacked vertically in white. The smartphone displays the word 'AJAX' in white.

AJAX PROGRAMMING

A

J

A

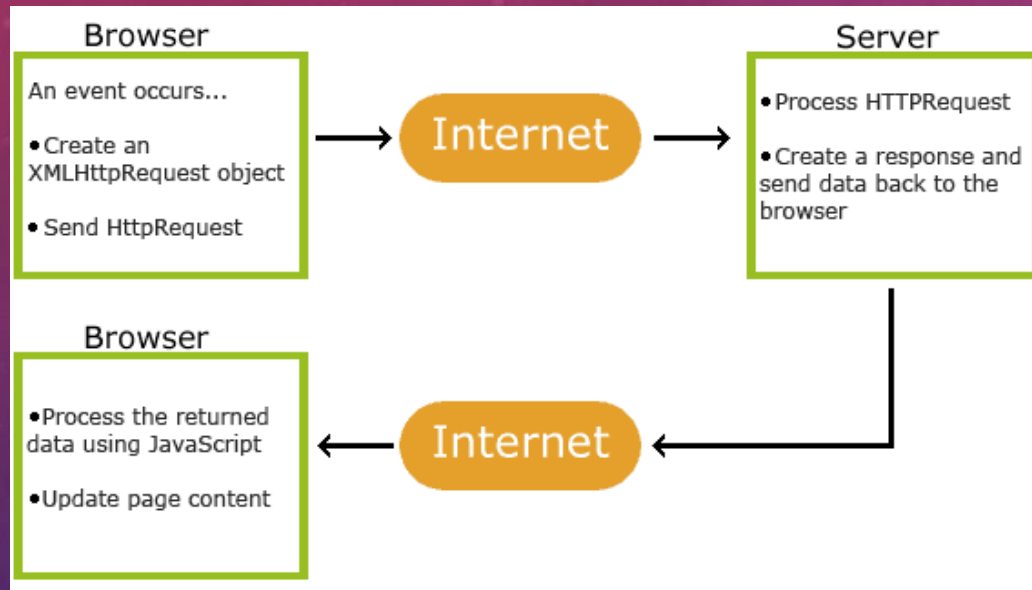
X

AJAX

WHAT IS AJAX?

- Asynchronous JavaScript And XML (AJAX)
 1. A browser built-in XMLHttpRequest object (to request data from a web server)
 2. JavaScript and HTML DOM (to display or use the data)
- AJAX allows web pages to be updated **asynchronously** by exchanging data with a web server behind the scenes. This means that it is possible to update parts of a web page, **without reloading the whole page**.

HOW AJAX WORKS



1. An event occurs in a web page (the page is loaded, a button is clicked)
2. An **XMLHttpRequest** object is created by JavaScript
3. The **XMLHttpRequest** object sends a request to a web server
4. The server processes the request
5. The server sends a response back to the web page
6. The response is read by JavaScript
7. Proper action (like page update) is performed by JavaScript

BASIC EXAMPLE

LOAD DATA FROM LOCAL SITE

```
<!DOCTYPE html>
<html>
<body>

<h2>The XMLHttpRequest Object</h2>

<p id="demo">Let AJAX change this text.</p>

<button type="button" onclick="loadDoc()">Change Content</button>

<script>
function loadDoc() {
  var xhttp = new XMLHttpRequest();
  xhttp.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
      document.getElementById("demo").innerHTML = this.responseText;
    }
  };
  xhttp.open("GET", "ajax_info.txt", true);
  xhttp.send();
}
</script>

</body>
</html>
```

var xhttp = new *XMLHttpRequest*();

xhttp.onreadystatechange

this.readyState == 4

this.status == 200

xhttp.open(*method*, *url*, *async*)

xhttp.send(*string*)

ONREADYSTATECHANGE 的三個重要屬性

属性	描述
onreadystatechange	存储函数（或函数名），每当 readyState 属性改变时，就会调用该函数。
readyState	<p>存有 XMLHttpRequest 的状态。从 0 到 4 发生变化。</p> <ul style="list-style-type: none">• 0: 请求未初始化• 1: 服务器连接已建立• 2: 请求已接收• 3: 请求处理中• 4: 请求已完成，且响应已就绪
status	<p>200: "OK"</p> <p>404: 未找到页面</p>

GET OR POST

- 與POST相比，GET更簡單也更快，並且在大部分情況下都能用。
- 然而，在以下情況中，請使用POST請求：
 1. 無法使用緩存文件（更新服務器上的文件或數據庫）
 2. 向服務器發送大量數據（POST沒有數據量限制）
 3. 發送包含未知字符的用戶輸入時，POST比GET更穩定也更可靠

異步與同步

- Async 是接收到需求，不用一直等到需求完成再執行其他需求。
- Async 與 Sync 的差別在於：發送需求的人是否需要等到需求完成才可以執行其他事情。

服務器響應

属性	描述
responseText	获得字符串形式的响应数据。
responseXML	获得 XML 形式的响应数据。

```
document.getElementById("demo").innerHTML =  
this.responseText;
```

服務器響應 XML

https://www.w3schools.com/js/cd_catalog.xml

```
<!DOCTYPE html>
<html>
<body>

<h2>The XMLHttpRequest Object</h2>

<p id="demo"></p>

<script>
var xhttp, xmlDoc, txt, x, i;
xhttp = new XMLHttpRequest();
xhttp.onreadystatechange = function() {
if (this.readyState == 4 && this.status == 200) {
  xmlDoc = this.responseXML;
  txt = "";
  x = xmlDoc.getElementsByTagName("ARTIST");
  for (i = 0; i < x.length; i++) {
    txt = txt + x[i].childNodes[0].nodeValue + "<br>";
  }
  document.getElementById("demo").innerHTML = txt;
}
};
xhttp.open("GET", "cd_catalog.xml", true);
xhttp.send();
</script>

</body>
</html>
```

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<CATALOG>
  <CD>
    <TITLE>Empire Burlesque</TITLE>
    <ARTIST>Bob Dylan</ARTIST>
    <COUNTRY>USA</COUNTRY>
    <COMPANY>Columbia</COMPANY>
    <PRICE>10.90</PRICE>
    <YEAR>1985</YEAR>
  </CD>
  <CD>
    <TITLE>Hide your heart</TITLE>
    <ARTIST>Bonnie Tyler</ARTIST>
    <COUNTRY>UK</COUNTRY>
    <COMPANY>CBS Records</COMPANY>
    <PRICE>9.90</PRICE>
    <YEAR>1988</YEAR>
  </CD>
  <CD>
    <TITLE>Greatest Hits</TITLE>
    <ARTIST>Dolly Parton</ARTIST>
    <COUNTRY>USA</COUNTRY>
    <COMPANY>RCA</COMPANY>
    <PRICE>9.90</PRICE>
    <YEAR>1982</YEAR>
  </CD>
  <CD>
    <TITLE>Still got the blues</TITLE>
    <ARTIST>Gary Moore</ARTIST>
    <COUNTRY>UK</COUNTRY>
    <COMPANY>Virgin records</COMPANY>
    <PRICE>10.20</PRICE>
    <YEAR>1990</YEAR>
  </CD>
  <CD>
    <TITLE>Eros</TITLE>
    <ARTIST>Eros Ramazzotti</ARTIST>
    <COUNTRY>EU</COUNTRY>
    <COMPANY>BMG</COMPANY>
    <PRICE>9.90</PRICE>
    <YEAR>1997</YEAR>
  </CD>
</CATALOG>
```

服務器響應 FROM PHP

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4
5 <h2>The XMLHttpRequest Object</h2>
6
7 <h3>Start typing a name in the input field below:</h3>
8
9 <p>Suggestions: <span id="txtHint"></span></p>
10
11 <p>First name: <input type="text" id="txt1" onkeyup="showHint(this.value)"></p>
12
13 <script>
14 function showHint(str) {
15     var xhttp;
16     if (str.length == 0) {
17         document.getElementById("txtHint").innerHTML = "";
18         return;
19     }
20     xhttp = new XMLHttpRequest();
21     xhttp.onreadystatechange = function() {
22         if (this.readyState == 4 && this.status == 200) {
23             document.getElementById("txtHint").innerHTML = this.responseText;
24         }
25     };
26     xhttp.open("GET", "gethint.php?q="+str, true);
27     xhttp.send();
28 }
29 </script>
30
31 </body>
32 </html>
```

```
1 <?php
2 // Array with names
3 $a[] = "Anna";
4 $a[] = "Brittany";
5 $a[] = "Cinderella";
6 $a[] = "Diana";
7 $a[] = "Eva";
8 $a[] = "Fiona";
9 $a[] = "Gunda";
10 $a[] = "Hege";
11 $a[] = "Inga";
12 $a[] = "Johanna";
13 $a[] = "Kitty";
14 $a[] = "Linda";
15 $a[] = "Nina";
16 $a[] = "Ophelia";
17 $a[] = "Petunia";
18 $a[] = "Amanda";
19 $a[] = "Raquel";
20 $a[] = "Cindy";
21 $a[] = "Doris";
22 $a[] = "Eve";
23 $a[] = "Evita";
24 $a[] = "Sunniwa";
25 $a[] = "Tove";
26 $a[] = "Unni";
27 $a[] = "Violet";
28 $a[] = "Liza";
29 $a[] = "Elizabeth";
30 $a[] = "Ellen";
31 $a[] = "Wenche";
32 $a[] = "Vicky";
```

```
34 // get the q parameter from URL
35 $q = $_REQUEST["q"];
36
37 $hint = "";
38
39 // lookup all hints from array if $q is different from ""
40 if ($q != "") {
41     $q = strtolower($q);
42     $len=strlen($q);
43     foreach($a as $name) {
44         if (striestr($q, substr($name, 0, $len))) {
45             if ($hint == "") {
46                 $hint = $name;
47             } else {
48                 $hint .= ", $name";
49             }
50         }
51     }
52 }
53
54 // Output "no suggestion" if no hint was found or output correct values
55 echo $hint == "" ? "no suggestion" : $hint;
56 ?>
```



THANK YOU

自學資源