# Department of Computing

**CS-344: Web Engineering**

**Class:** BESE-9AB

# Lab 09: jQuery, AJAX, XML, JSON

**Date: December 10, 2020**

# Instructor: Dr. Qaiser Riaz

# 

**Ahmed Hassan Ismail – BESE-9B – 237897**

# Lab 09: jQuery, AJAX, XML, JSON

### Introduction:

jQuery is a widely used and well-known library for JavaScript which helps in rapid web development. Most of the modern websites use jQuery as a tool to implement and control client side dynamic behavior of the websites. XML and JSON are two commonly used ways of storing and organizing data in text format. Students have learned basic and advanced concepts of using AJAX for asynchronous communication between server and client. This lab will help them to further understand these concepts by developing example applications.

### Lab Objectives:

The objective of this lab is helping students to familiarize themselves with basic and advanced concepts of AJAX by practically implementing them in given scenarios. The knowledge students have gained in the lectures will help them to develop and control dynamic behavior of web pages using advanced concepts of jQuery.

### Tools:

Notepad, DreamWeaver, browser.

### Helping Material:

Lecture slides.

W3Schools: <https://www.w3schools.com/js/default.asp>

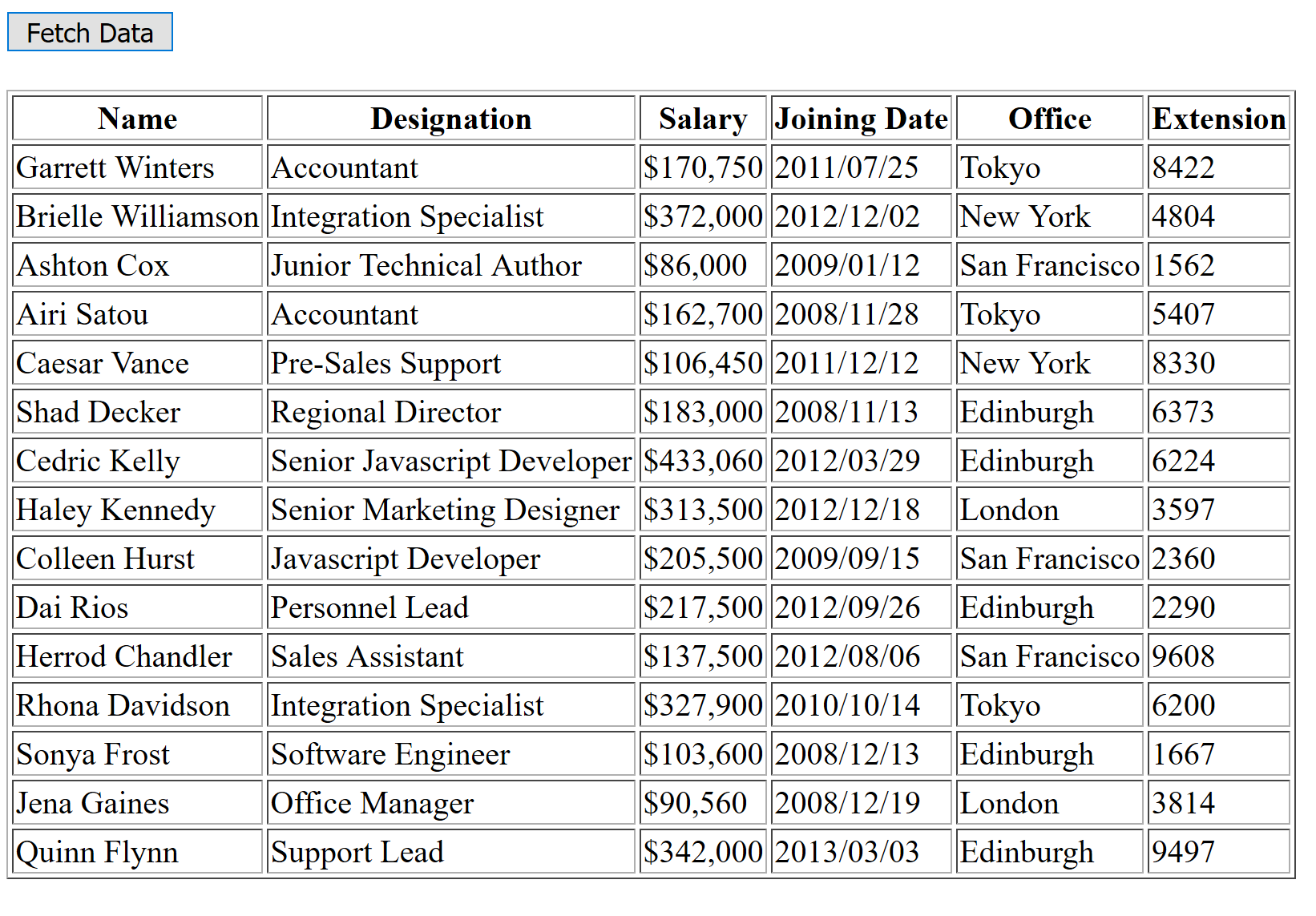
jQuery API: <http://api.jquery.com/>

jQuery Cheatsheet: <https://oscarotero.com/jquery/>

### Lab Task

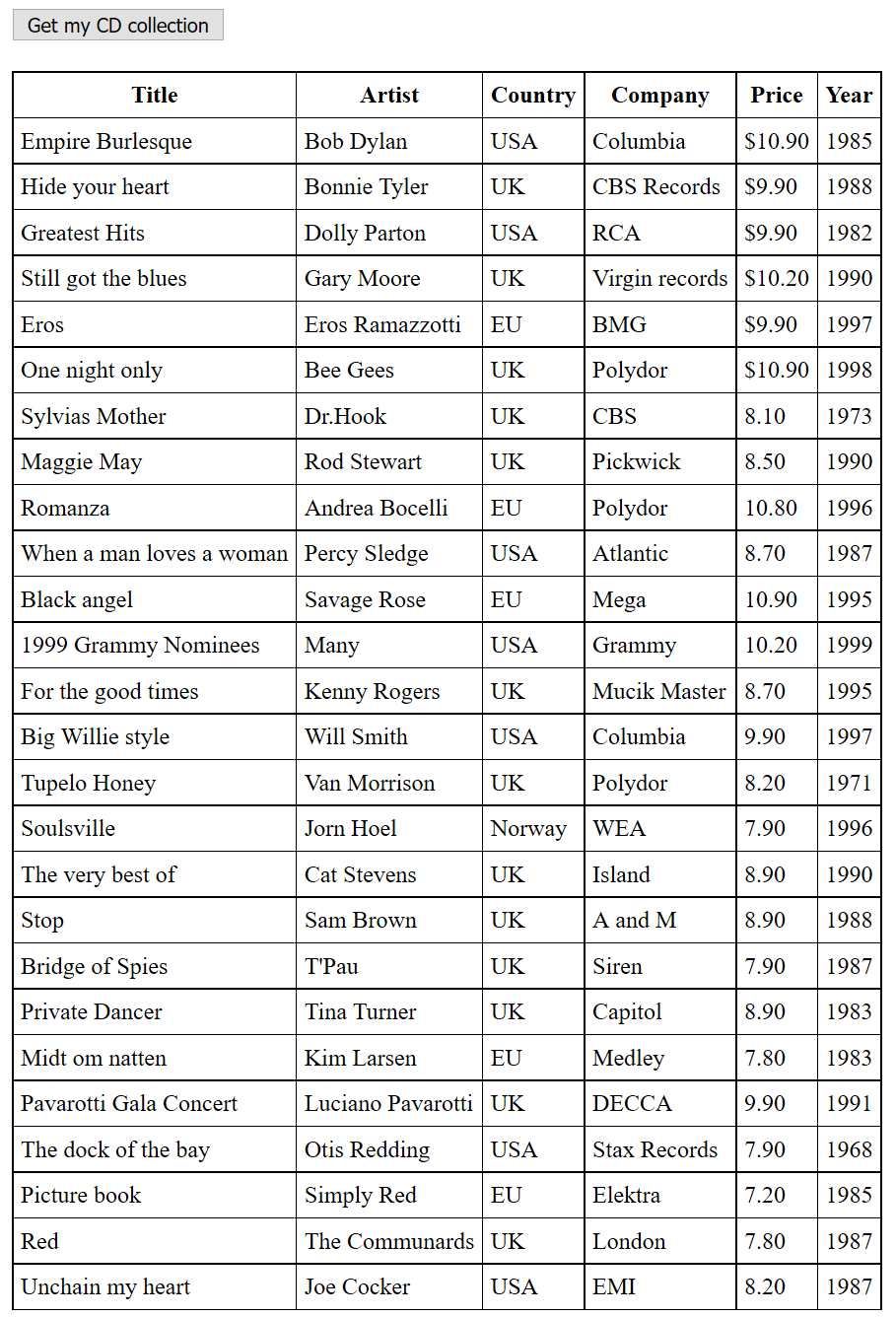
### Task 1

In the supporting material compressed file (see ‘lab-9-supporting-material.zip’), a skeleton html file is given which include basic skeleton (‘task-1-ajax-json-skeleton.html’) for this task. A sample JSON file is also include; see ‘emp\_data.json’ file. You can open this file in any text editor (e.g. notepad) to view its contents. Your task is to load this file using jQuery AJAX function, parse the JSON file, and convert the JSON data into an HTML table as shown in the below screenshot.



### Task 2

In the supporting material compressed file (see ‘lab-9-supporting-material.zip’), a skeleton html file is given which include basic skeleton (‘task-2-ajax-xml-skeleton.html’) for this task. A sample XML file is also included; see ‘cd\_catalog.xml’ file. You can open this file in any text editor (e.g. notepad) to view its contents. Your task is to load this file using jQuery AJAX function, parse the XML file, and convert the XML data into an HTML table as shown in the below screenshot. Important: For some of the <PRICE> elements, an option “currency” attribute is also included. When generating the HTML table, include the currency sign with price for only those elements where currency attribute is added. For the rest, price should be without any currency symbol. For example, see first six rows in the screen shot below.



Hints:

1. When parsing a JSON file, dataType should be json i.e. dataType: "json". Similarly, when parsing XML file, dataType should be xml i.e. dataType: "xml".
2. To find currency attribute in XML file: $(this).find("PRICE").attr("currency").
3. Use .each() function for looping

**Note: Upload complete solutions (css, html, js) for each task in in a single zip file along with adding jQuery and screenshots of your solutions in this word file.**

|  |
| --- |
| Solution |
| Task 1 jQuery:  <!DOCTYPE HTML>  <html>  <head>      <meta charset="utf-8">      <title>AJAX JSON</title>      <script src="jquery-3.2.1.js"></script>  </head>  <body>      <button type="button" id='submitButton'>Fetch Data</button>      <br><br>      <div id="location">      </div>      <script>          $(document).ready(function() {              alert('jQuery is working!')              $('#submitButton').on('click', function() {                  $("#location").html('<table border = "1" id ="json"><tr><th>ID</th><th>First\_Name</th><th>Last\_Name</th></tr></table>');                  $.ajax({                      url: "https://reqres.in/api/users?page=2",                      dataType: 'json',                      success: showData                  });              });              function showData(response) {                  console.log(response.data);                  for (i = 0; i < response.data.length; i++) {                      $("#json").append("<tr><td>" + response.data[i].id + "</td>" + "<td>" + response.data[i].first\_name + "</td>" + "<td>" + response.data[i].last\_name + "</td> </tr>");                  }                }          });      </script>  </body>  </html>  Task 1 screenshot:    Task 2 jQuery:  <!DOCTYPE html>  <html>  <style>      table,      th,      td {          border: 1px solid black;          border-collapse: collapse;      }        th,      td {          padding: 5px;      }  </style>  <body>      <button type="button" id='submitButton'>Get my CD collection</button>      <br><br>      <table id="demo"></table>      <div id="error"></div>      <script src="jquery-3.2.1.js"></script>      <script>          $(document).ready(function() {              $('#submitButton').on('click', function() {                  $("#demo").append("<tr><th>Title</th><th>Artist</th><th>Country</th><th>Company</th><th>Price</th><th>Year</th></tr>");                  $.ajax({                      url: "cd\_catalog.xml",                      dataType: 'xml',                      success: showData                  });              });              function showData(response) {                  $(response).find("CD").each(                      function() {                          var at = $(this).find("PRICE").attr('currency');                          if (typeof attr !== typeof undefined) {                          } else {                              attr = " "                          }                          $("demo").append("<tr> <td>" + $(this).find("TITLE").text() + "</td>" + $(this).find("ARTIST").text() + "</td>" + $(this).find("COUNTRY").text() + "</td>" + $(this).find("COMPANY").text() + "</td>" + at + $(this).find("PRICE").text() + "</td>" + $(this).find("YEAR").text() + "</td> </tr>");                      }                  )              }          })      </script>  </body>  </html>  Task 2 screenshot: |

### Deliverables

Compile a single word document by filling in the solution part and submit this Word file on LMS. You must include your name, ID, and class on first page. The lab grading policy is as follows: The lab is graded between 0 to 10 marks. For some of the labs, students have to present their solutions in a viva session. In case of any problems with submissions on LMS, you should contact your lab engineer Mr. Aftab Hussain by email at [aftab.hussain1@seecs.edu.pk](mailto:aftab.hussain1@seecs.edu.pk).