## APPENDIX 3 PUBLIC API

If you wish to follow along with this part of the demonstration you will need to visit <https://www.edamam.com/> and navigate to their developer API section of the site. There you can sign up for a developer’s key to access their APIs.

Please note that Skillsoft does not endorse any API consumption and this is for demonstration purposes only.

Also, the site has strict rules and policies about the consumption of their APIs, please abide by these rules. Skillsoft is not responsible for the violation of any rule of any API provider.

Follow these steps to consume the Edamam API and complete the **index.html** page of the website.

1. Copy the folder labeled Day02-JS-04 and rename it to Day02-JS-05.
2. Open the index.html file in a text editor and remove all the anchor tags from the second section of the aside part of the file
3. Replace the anchor tags with just a single div tag as shown below

|  |
| --- |
| **</section>**  **<section>**  **<h4>Healthy Recipes</h4>**  **<div id="recipes"></div>**  **</section>**  **</aside>**  **</div>** |

We will use the innerHTML of this div tag to replace the static urls with live ones.

1. (Optional) Create a new .js file in the scripts folder and call it env.js or something similar. Create a const variable here and pass it the value of the key you received from Edamam, the code below is just a sample:

|  |
| --- |
| **const pKey = "k4l8fsaoiouiue4l589hfsd6520fb3ec";** |

Note: you do not have to do this if you are on your personal computer, the instructor is doing this in order to not expose his key. You can insert the key directly into the API calls, or create a variable in the scrips.js file itself.

1. If you did do step 5, in index.html, add that file above the scripts.js file so that it is available to scripts.js.

|  |
| --- |
| **<footer>**  **<hr />**  **Copyright &copy; 2019. All rights reserved**  **</footer>**  **<script src="scripts/env.js"></script>**  **<script src="scripts/scripts.js"></script>**  **</body>** |

1. In scripts.js remove all content except the first and third lines, so this is what your scripts.js file should look like now

|  |
| --- |
| **let xmlhttp = new XMLHttpRequest();**  **let url = "http://localhost:8000/getallrecords";** |

1. Based on the documentation on the Edmam.com website, complete the URL in order to get their fish recipes

|  |
| --- |
| **let xmlhttp = new XMLHttpRequest();**  **let url = "https://api.edamam.com/search?q=fish&app\_id=c56976b0&app\_key="+pKey+"&from=0&to=4";** |

The pKey value will be inserted from the env.js file. If you did not do step 5, just insert your own key here. Note, depending on which code editor you are using, it may be better to have this as one line, do not break up the line as Word has done here.

1. We can now call the open() method of the xmlhttp object, pass it the type of request, the url and a value to indicate weather we are making the request synchronously or not, we will be requesting asynchronously, so pass true in the end

|  |
| --- |
| **let xmlhttp = new XMLHttpRequest();**  **let url = "https://api.edamam.com/search?q=fish&app\_id=c56976b0&app\_key="+pKey+"&from=0&to=4";**  **xmlhttp.open("GET", url, true);** |

1. Send the request

|  |
| --- |
| **let xmlhttp = new XMLHttpRequest();**  **let url = "https://api.edamam.com/search?q=fish&app\_id=c56976b0&app\_key="+pKey+"&from=0&to=4";**  **xmlhttp.open("GET", url, true);**  **xmlhttp.send();** |

1. The api will respond with lots of data, we only need the hits part of that huge structure (refer to the Edamam.com’s website for this kind of detail). We can access this data via the onload event of the xmlhttp object

|  |
| --- |
| **let xmlhttp = new XMLHttpRequest();**  **let url = "https://api.edamam.com/search?q=fish&app\_id=c56976b0&app\_key="+pKey+"&from=0&to=4";**  **xmlhttp.onload = function () {**  **let data = JSON.parse(this.response);**    **}**  **xmlhttp.open("GET", url, true);**  **xmlhttp.send();** |

You may show this in the console window if you want to take a look at what is being returned.

11. Now that we can get the data, we can extract only the parts of that huge response object that we want to include in our development. In this case we may be interested in the links to recipes along with their images and description. Lets iterate through the data object to get these details

|  |
| --- |
| **let xmlhttp = new XMLHttpRequest();**  **let url = "https://api.edamam.com/search?q=fish&app\_id=c56976b0&app\_key="+pKey+"&from=0&to=4";**  **xmlhttp.onload = function () {**  **let data = JSON.parse(this.response).hits;**  **for(let i=0; i<data.length; ++i){**  **console.log(data[i].recipe.url +" "+**  **data[i].recipe.image +" "+**  **data[i].recipe.label);**  **}**  **}**  **xmlhttp.open("GET", url, true);**  **xmlhttp.send();** |

12. If this works, then we would need to construct html with these values and pass it to the innerHTML of the recipes div. also lets make sure that we did not receive a 404 or a 500 or any other type of error codes from the request.

|  |
| --- |
| **let xmlhttp = new XMLHttpRequest();**  **let url = "https://api.edamam.com/search?q=fish&app\_id=c56976b0&app\_key="+pKey+"&from=0&to=4";**  **xmlhttp.onload = function () {**  **let data = JSON.parse(this.response).hits;**  **if (xmlhttp.status >= 200 && xmlhttp.status < 400) {**  **for(let i=0; i<data.length; ++i){**  **document.getElementById("recipes").innerHTML +=**  **"<div class='recipeText'><a href='"+**  **data[i].recipe.url +"'><img src='"+**  **data[i].recipe.image +"' class='recipeImage'/ >"+**  **data[i].recipe.label+"</a></div>";**  **};**  **}**  **}**  **xmlhttp.open("GET", url, true);**  **xmlhttp.send();** |

1. Although this will work, we should make the images smaller and have the description next to the images, so add some css to the styles.css file

|  |
| --- |
| **#logo{**  **float:right;**  **width:160px;**  **height:88px;**  **}**  **.recipeImage{**  **width: 50px;**  **height: 50px;**  **display:inline-block;**  **vertical-align: middle;**  **}**  **.recipeText{**  **font-size: .8em;**  **}** |

1. Final image  
   