

```

1 <html>
2   <head>
3     <meta charset='utf-8'>
4     <meta http-equiv='X-UA-Compatible' content='IE=edge'>
5     <title>yo</title>
6     <meta name='viewport' content='width=device-width, initial-scale=1'>
7     <link rel='stylesheet' type='text/css' media='screen' href='results.css'>
8     <link rel="preconnect" href="https://fonts.gstatic.com">
9     <link href="https://fonts.googleapis.com/css2?
10 family=Josefin+Sans&display=swap" rel="stylesheet">
11     <script type="text/javascript"
12 src="//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js"></script>
13     <script src='main.js'></script>
14   </head>
15   <body>
16     <div class="topnav">
17       <a href="main.html">Home</a>
18       <a href="select.php">Visualize (profilin)</a>
19       <a href="select_nsltp.php">Visualize (nsLTP)</a>
20       <a class="active" href="select_pr10.php">Visualize (pr10)</a>
21       <a href="personal.php">Inspect</a>
22     </div>
23     <style>
24       body{
25         font-family: 'Josefin Sans', sans-serif;
26       }
27     </style>
28     <?php
29     $servername = "localhost";
30     $username = "root";
31     $password = "Iatwbofm21!";
32     $dbname = "crossreactivity";
33
34     $temp = $_GET['display'];
35     # echo "****" . $temp;
36     #echo "<br>";
37     // Create connection
38     $conn = new mysqli($servername, $username, $password, $dbname);
39     // Check connection
40     if ($conn->connect_error) {
41       die("Connection failed: " . $conn->connect_error);
42     }
43
44     $sql = "SELECT name, origin FROM crossreactivity_pr10";
45     #echo $sql;
46     $result = $conn->query($sql);
47     if ($result->num_rows > 0) {
48       // output data of each row
49       $maps = array();
50       $maps_otn = array();
51       while($row = $result->fetch_assoc()) {
52         $maps[$row['name']] = $row['origin'];
53         $maps_otn[$row['origin']] = $row['name'];
54       }
55     } else {
56       echo "0 results";
57     }
58
59
60     $query = "SELECT * FROM crossreactivity_pr10 WHERE name = '". $temp . "'";
61     #echo "<br>";
62     #echo $query;
63     #echo "<br>";
64     $result = mysqli_query($conn, $query);
65
66     $rows = mysqli_fetch_all ($result, MYSQLI_ASSOC);
67
68     foreach( $rows as &$row){
69       array_multisort($row, SORT_DESC, SORT_NUMERIC);
70

```

```

71 $origin = $row['origin'];
72 $name = $row['name'];
73
74 echo "<h1><center>$origin</center></h1>";
75 echo "<br>";
76 #echo $name;
77
78 unset($row['origin']);
79 unset($row['name']);
80
81 /*
82 foreach($row as $key => $value){
83     echo $value;
84     echo " ";
85 }
86 */
87
88 echo "<br>";
89
90 $last = end($row);
91 $first = reset($row);
92
93 #echo $row;
94
95 foreach($row as $key => $value){
96     #echo $value;
97     #echo $maps[$key]." : ".$key . " : " . $value . "<br>";
98     $last = (float)$last ;
99     $color = numberToColor($value,0,100, ['#8de667', '#fff5f5',
100 '#CC0000']);
101     #echo $color;
102     echo "<div class='flip-card'>
103         <div class='flip-card-inner'>
104             <div class='flip-card-front'>
105                 <div style=' padding:10%;display:inline-block;
106 height:80%;width:75%; font-size:28px; color: black; background-color:" . $color . ";
107 text-align:center'>$maps[$key]</div>
108                 </div>
109                 <div class='flip-card-back'>
110                     <h1>$maps[$key]</h1>
111                     <p>$key</p>
112                     <p>$value</p>
113                 </div>
114             </div>
115             </div>";
116             #echo "<div class = 'result'>$final</div>";
117         }
118         echo "<br>";
119         echo "<br>";
120         echo "<br>";
121         echo "<br>";
122     }
123     $conn->close();
124
125 function numberToColor($value, $min, $max, $gradientColors = null)
126 {
127     // Ensure value is in range
128     if ($value < $min) {
129         $value = $min;
130     }
131     if ($value > $max) {
132         $value = $max;
133     }
134
135     // Normalize min-max range to [0, positive_value]
136     $max -= $min;
137     $value -= $min;
138     $min = 0;
139
140     // Calculate distance from min to max in [0,1]
141     $distFromMin = $value / $max;

```

```

140 // Define start and end color
141 if (count($gradientColors) == 0) {
142     return numberToColor($value, $min, $max, ['#CC0000', '#EEEE00',
143 '#00FF00']);
144 } else if (count($gradientColors) == 2) {
145     $startColor = $gradientColors[0];
146     $endColor = $gradientColors[1];
147 } else if (count($gradientColors) > 2) {
148     $startColor = $gradientColors[floor($distFromMin *
149 (count($gradientColors) - 1))];
150     $endColor = $gradientColors[ceil($distFromMin *
151 (count($gradientColors) - 1))];
152
153     $distFromMin *= count($gradientColors) - 1;
154     while ($distFromMin > 1) {
155         $distFromMin--;
156     }
157 } else {
158     die("Please pass more than one color or null to use default red-green
159 colors.");
160 }
161
162 // Remove hex from string
163 if ($startColor[0] === '#') {
164     $startColor = substr($startColor, 1);
165 }
166 if ($endColor[0] === '#') {
167     $endColor = substr($endColor, 1);
168 }
169
170 // Parse hex
171 list($ra, $ga, $ba) = sscanf("#$startColor", "%02x%02x%02x");
172 list($rz, $gz, $bz) = sscanf("#$endColor", "%02x%02x%02x");
173
174 // Get rgb based on
175 $distFromMin = $distFromMin;
176 $distDiff = 1 - $distFromMin;
177 $r = intval(($rz * $distFromMin) + ($ra * $distDiff));
178 $r = min(max(0, $r), 255);
179 $g = intval(($gz * $distFromMin) + ($ga * $distDiff));
180 $g = min(max(0, $g), 255);
181 $b = intval(($bz * $distFromMin) + ($ba * $distDiff));
182 $b = min(max(0, $b), 255);
183
184 // Convert rgb back to hex
185 $rgbColorAsHex = '#' .
186     str_pad(dechex($r), 2, "0", STR_PAD_LEFT) .
187     str_pad(dechex($g), 2, "0", STR_PAD_LEFT) .
188     str_pad(dechex($b), 2, "0", STR_PAD_LEFT);
189
190 return $rgbColorAsHex;
191 }
192
193 ?>
194 <br>
195 
196 <p>The darker the red, the more you should avoid that substance, the lighter
197 the green, the more ok it is to be around. Flip the cards over to see a risk-score
198 between 0 and 100</p>
199 <br>
200 </body>
201 <script type = "text/javascript">
202     $( "img" ).each( function() {
203         var $img = $( this );
204         $img.width( $img.width() * .5 );
205     });
206 </script>
207 </html>

```