CS/INFO 3300 // INFO 5100

Homework 1

Due at 11:59pm Monday, February 4

Goals: Review visual channels, Javascript types, data structures, and JSON.

Your work should be in the form of an HTML file called index.html with one `<p>` element per problem. Wrap any Javascript code for each problem in a `<script>` element nested within the `<p>` element.
For example:

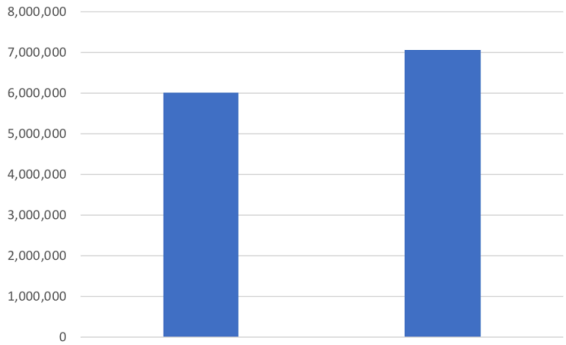
```
<p id="p0">Problem 0: We use the var statement to declare a
variable and set it to a value.
<script>
var x = 100;
</script>
</p>
```

Create a zip archive containing this file and upload it to CMS before the deadline.

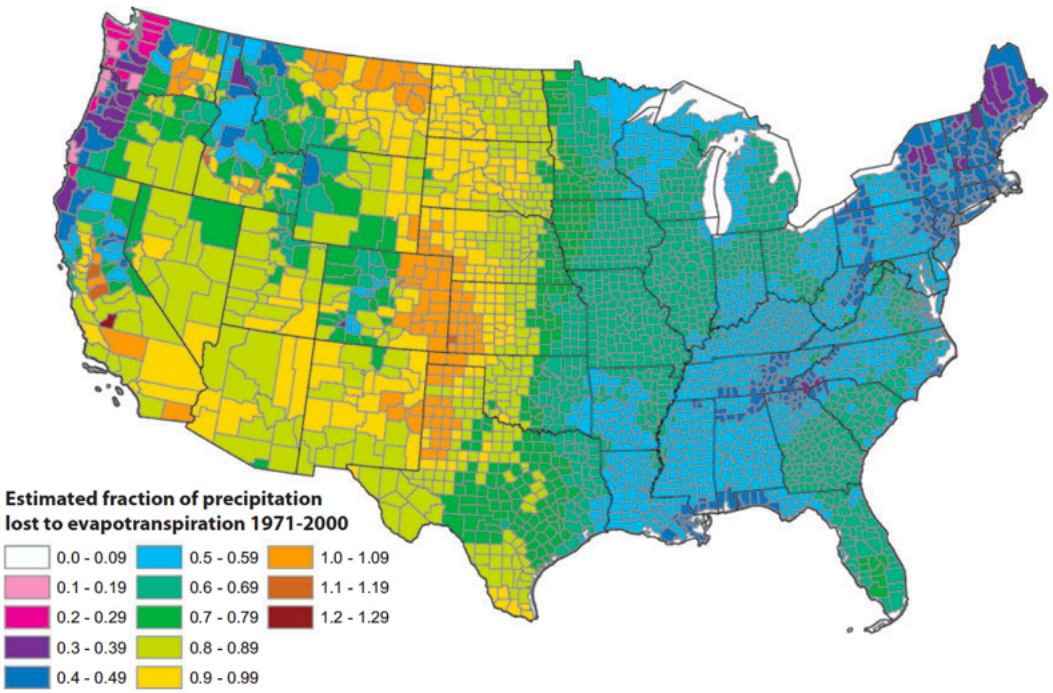
1. For each of the following visualizations, identify the **marks** used and **visual channels** from the list provided. Be as specific as possible. Ignore axes and axis labels. (16 points)

Possible visual channels:
Aligned/Unaligned Position (x,y); Aligned/Unaligned Length; Area; Volume; Color Hue

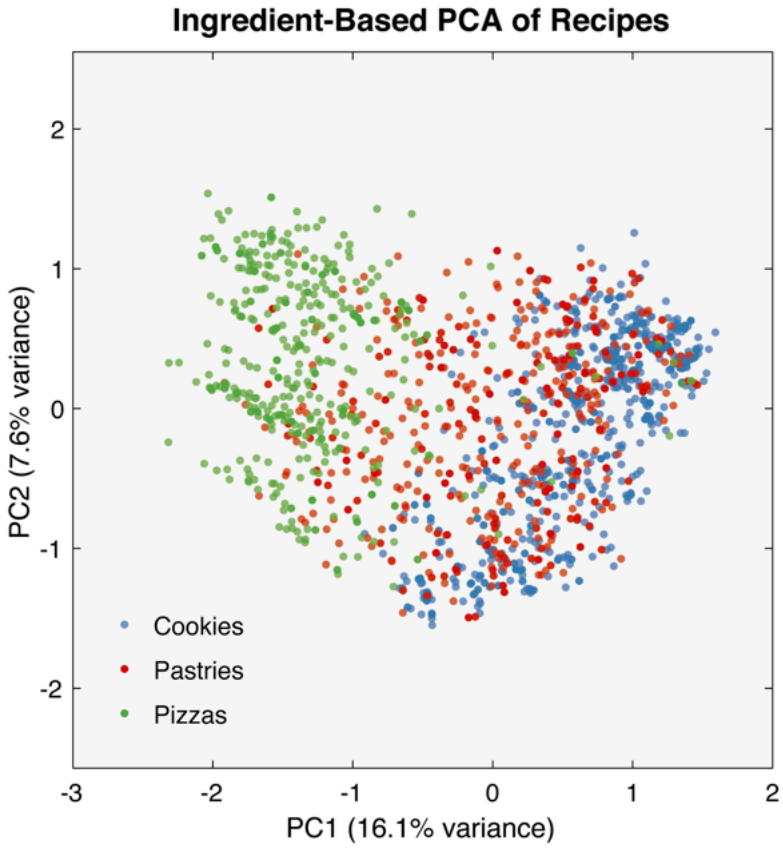
EXAMPLE:



Marks: Blue rectangles
Channels: Varying the vertical aligned length and horizontal position of the rectangles



a)



b)

(see next page)

2. **Identify** which of the following are valid Javascript types (i.e. can be returned as output from the typeof() function). For each valid type **provide an example** using console.log(typeof(...)) in your `<script>` section. If applicable, identify any unusual or unexpected `typeof()` behavior for a given type. (24 pts)

a) `regex`; b) `string`; c) `unicode`; d) `list`; e) `map`; f) `object`;
g) `number`; h) `int`; i) `boolean`; j) `float`; k) `null`; l) `function`

3. Each of the following expressions contains an improper JSON string. Calling JSON.parse(...) on them will result in an error. For each example, **describe** the error and provide a **fixed version**. Use the JSON.parse function to show that it works in your `<script>` section. Some may contain more than one error. (20 pts)

a) { netid: 'jmr395' }
b) { /* choose one of these options */ "number": "INFO3300/CS3300/INFO5100" }
c) [3; 4.3; null];
d) [{ "name": "DDWA", "number": 3300 }{ "name": "Java", "number": 2110 }{ "name": "Python", "number"; 1110 }

4. Write a JSON expression that represents **a list**. This list should contain **at least three objects**. Each object should have **at least three properties**. The value of one property should be a **number**, the value of another should be a **boolean**, and value of a third should be a **list of at least two strings**. You may define the contents in any way you choose and add additional properties as long as we can easily verify that the required elements are present. (30 pts)

5. Describe, in your own words, the course policy on late and unreadable work outlined in the syllabus, which is linked from the course website. (10 pts)