Department of
COMPUTER SCIENCE
HONG KONG BAPTIST UNIVERSITY

COMP7980 – Dynamic Web and Mobile Programming
COMP7270 – Web and Mobile Programming

Chapter 5 Data Visualization

Course Instructors: Dr. Ma Shichao, Dr. Zhang Ce, and Mr. Jiang Jintian

Announcement

- Assignment 1 - Online Survey System (Non-Ajax)
  - See the guidelines on Moodle
  - Deadline: March 9, 24:00
  - Email submission is not allowed, and late submissions will not receive marks

Preparations

- Choose Mac OS, account: guest
- Download lab05-materials from Moodle
- Extract it to a folder
  - Rename the folder name to ‘lab05’
- Get into the folder and install the libraries
  - `cd lab05`
  - `npm install`
- Setup the MongoDB database
  - In the file `utils/db.js`, please assign the variable `process.env.MONGODB_URI` as your own connection string

The most advanced chart package

AMCHARTS

Classics with some new twists

XY charts are now so powerful and flexible, you can plot any data on them. Number, date, duration, or category axes are supported, in all directions.

The axes can now contain interactive breaks, that expand on hover and actually look awesome.

New geo maps

Maps now use **GeoJSON** format! Being open and widely accepted standard it opens up a lot of possibilities and sources for ready-made and custom maps.

Furthermore, maps are now very flexible, with multi-series support, configurable down to the nut and bolt.

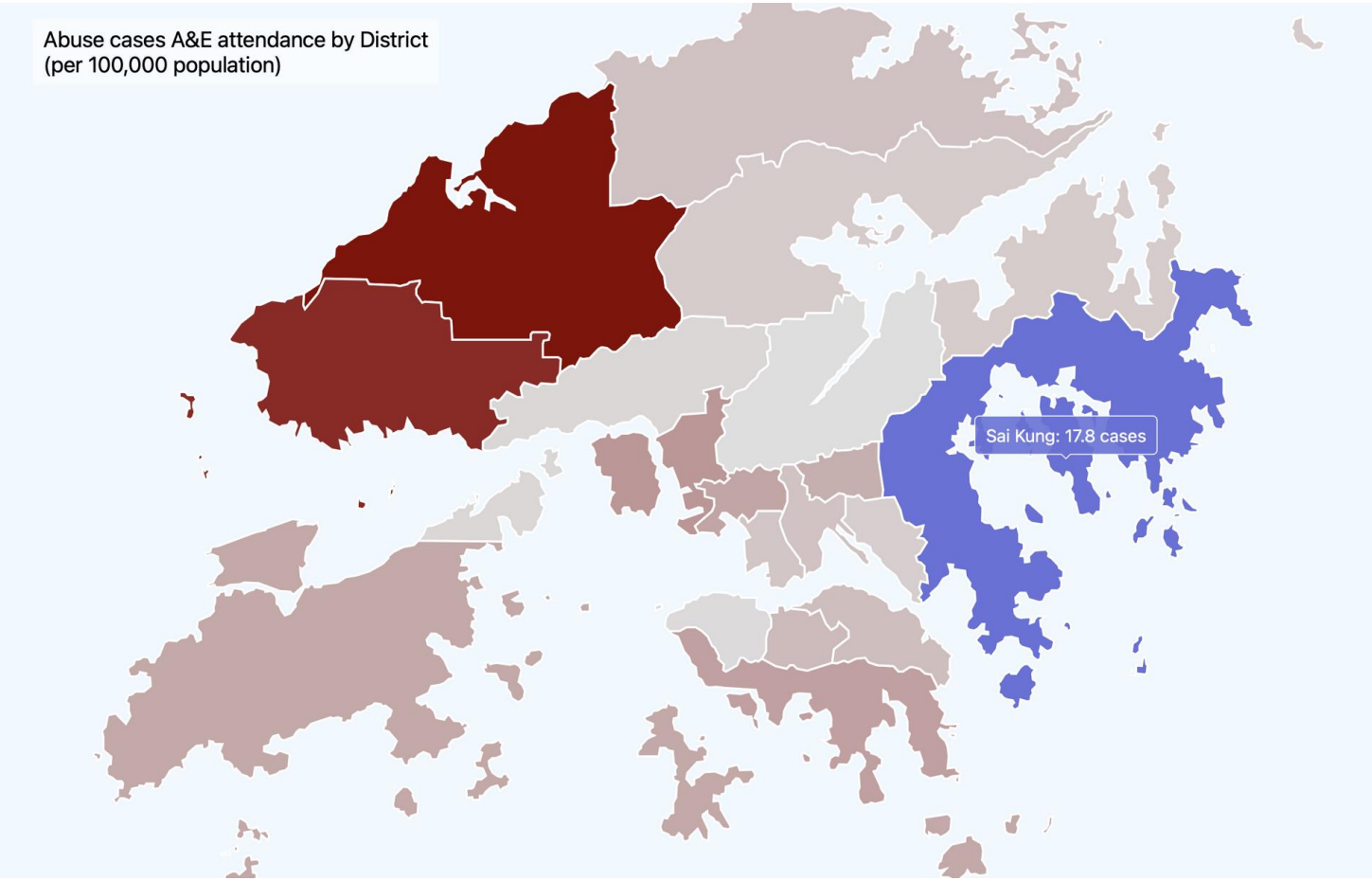
More about amCharts 4 maps

Pictorials

Create multi-layer, multi-series pictorial charts. Any SVG path can be used as a shape for your chart.

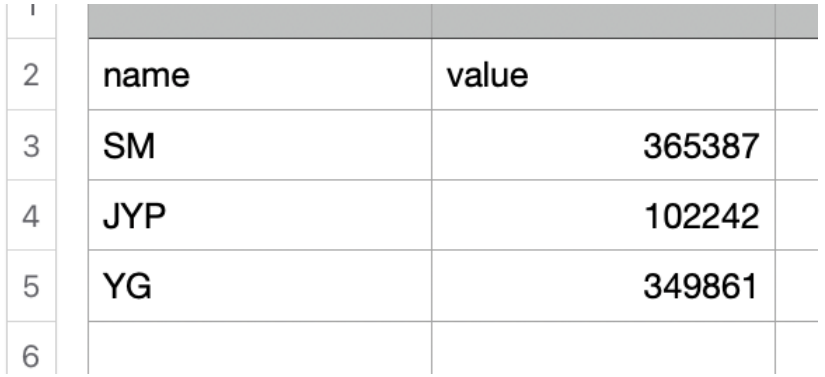
https://www.amcharts.com

- While the **amCharts chart** drawing frontend syntax is **not necessary for the examination**, the code responsible for populating data from the route handler to the EJS template is required.



Object Array

- An array of plain objects
- It can represent **tubular data**, like an excel worksheet.
- It is a very important data structure in this course.



	name	value
2		
3	SM	365387
4	JYP	102242
5	YG	349861
6		

```
a = [{
      name: "SM",
      value: 365387
    }, {
      name: "JYP",
      value: 102242
    }, {
      name: "YG",
      value: 349861
    }]
```

```
a[0] = {
      name: "SM",
   value: 365387
 }
```

```
a[2].value = 349861
```

JSON.Stringify

- JSON.stringify() is a method in JavaScript that **converts a JavaScript** object or value to **a JSON string**.
- It takes an object or value as a parameter and **returns a string representation of that object** in JSON format.

```
const obj = {
  name: 'John',
  age: 30,
  city: 'New York'
};

const jsonString = JSON.stringify(obj);
console.log(jsonString);
// Output: {"name":"John","age":30,"city":"New York"}
```

Web Template Engine

```
<table>
  <% for (var booking of bookings) { %>

    <tr>
      <td><%= booking.email %></td>
      <td><%= booking.numTickets %></td>
    </tr>

  <% } %>
</table>
```

<% %>	'Scriptlet' tag, for **control-flow**, no output
<%= %>	**Outputs** the value into the template (**HTML escaped**)
<%- %>	**Outputs** the **unescaped** value into the template

```
series.data.setAll(eval(`<%- JSON.stringify(data) %>`));
```

HTML Escape

- HTML escape is a technique used to **represent special characters** and symbols in HTML documents **using their corresponding character entities**. In HTML, certain characters have special meanings, such as **<** and **>** for tags, **&** for entities, and **"**, **'**, and & for attribute values.

Unescaped	Escaped	Meaning
<	<	Less than
>	>	Greater than
"	"	Quotation mark
&	&	Ampersand
(a space)		Non-breaking space

```
Non-escape: {"name":"John","age":30,"city":"New York"}
```

eval()

- eval() is a built-in JavaScript function that allows you to **evaluate a string of code and execute it** as if it were part of the original code.
- It takes **a string parameter** representing a JavaScript expression or statement and returns the result of evaluating that code.

```
const x = 5;
const y = 10;
const code = 'x + y';
const result = eval(code); // Evaluates the expression 'x + y'
console.log(result); // Output: 15
```