**Information Visualisation (M), 2022/23**

**Lab Session 2, Monday 23rd January**

**Vega-Lite and Observable**

For your Information Visualisation group project, you will be required to implement at least two visualisation systems. The focus of the assessment will be on your written report (and in particular the justification and evaluation of your designs), but you will still need to implement working code. You can use JavaScript to implement the systems, if you wish. (An alternative choice is Python+Altair, which were the subject of Lab 1.)

The tutorials below will give you a good starting point for this. They are provided by the University of Washington Interactive Data Lab (**uwdata**) by internationally well-known Information Visualisation researchers. You may soon recognise the pattern of work here – it is basically the same as in Lab 1, but using a different language to create the same kind of functionality.

Even if you prefer Python, and are unfamiliar with JavaScript (JS), we strongly encourage you to have a thorough look at JavaScript as a language, at Observable as a ‘live’ programming environment, and at Vega-Lite as a visualisation toolkit. A quick introduction to JavaScript (and Observable) is [here](https://observablehq.com/@uwdata/a-minimal-introduction-to-javascript-and-observable). Python *may* be stronger for analytics based on machine learning, statistics, and the like, but we suggest that analytics based on advanced interaction methods is better in JS. The integration with web technologies makes JS-based systems very easy to distribute. Also, we remind you that—even though this lab work is not assessed in itself, the material here is potentially examinable (albeit at a basic level, as we don’t assume as much in-depth JS knowledge as Python). In other words, students would be wise to do some basic work here.

You are free to start working through these tutorials whenever you like, although you may want to devote the time allocated to the weekly scheduled lab sessions to doing them – in particular, Monday 23rd January. The lab assistants will try to help if you have any problems, although we know that neither have used JS before (and Matthew has only done a *tiny* bit).

**The Data Visualisation Curriculum**

Developed at the University of Washington by Jeffrey Heer, Dominik Moritz, Jake VanderPlas, and Brock Craft, this is a great resource for anyone wishing to learn more about the technical side of information visualisation. Observable uses a web-based environment, so setup is much easier than Python. Just go to the curriculum web page, for the overview:

* <https://github.com/uwdata/visualization-curriculum>

and then on to the following pages:

**Introduction:**

* <https://observablehq.com/@uwdata/introduction-to-vega-lite>

**Marks and encoding**

* <https://observablehq.com/@uwdata/data-types-graphical-marks-and-visual-encoding-channels>

**Data Transformation**

* <https://observablehq.com/@uwdata/data-transformation>

**Scales, axes and legends**

* <https://observablehq.com/@uwdata/scales-axes-and-legends>

**Multi-view composition**

* [https://observablehq.com/@uwdata/multi-view-composition](•%09https:/observablehq.com/@uwdata/multi-view-composition)

**Interaction**

* <https://observablehq.com/@uwdata/interaction>

# JavaScript

If you are a novice JavaScript user, there a whole range of resources that you can use to enhance your skills; we recommend online tutorials, e.g.:

<https://javascript.info/>

<https://www.w3schools.com/js/>

<https://www.codecademy.com/learn/introduction-to-javascript>

# Lab Assistants and Lab Sessions

The lab sessions are drop-in sessions where you can get technical advice on your project, and where you are expected to work through the tutorials. You are not required to attend any lab sessions, and attendance will not be taken.

Please do not ask the lab assistants to write any of your code for you – they are there to advise you on how you can do it yourself.

A lecturer will usually be available during these lab sessions to answer questions about the theoretical aspects of the project, either by online via Moodle or email, in the Boyd Orr lab (1028), or both.