

**Monash University**  
**FIT5147 Data Exploration and Visualisation**  
**Semester 1, 2024**

## **Programming Exercise 3: D3 (5%)**

### **Programming Exercise 3: D3**

Please carefully review all the requirements below to ensure you have a good understanding of what is required for your assessment.

1. **Instructions & Brief**
  2. **Assessment Resources**
  3. **Assessment Criteria**
  4. **How to Submit**
  5. **Late Penalty**
- 

#### **1. Instructions & Brief**

In this assignment you are required to demonstrate the ability to complete the source code of an interactive visualisation built using D3.

#### ***Relevant learning outcomes for FIT5147:***

6. Implement interactive data visualisations using R and other tools

#### **The final visualisation:**

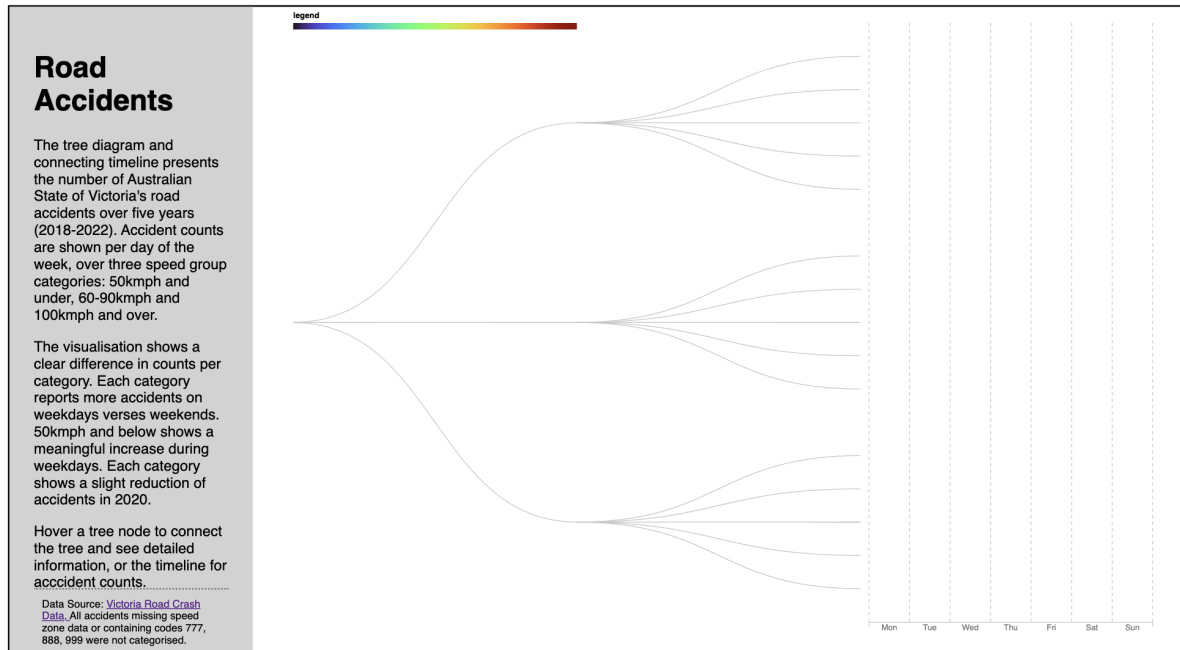
The expected final visualisation is within the same topic of Victoria's accidents data (similar to PE1 and PE2). The outcome is a web page containing two complementary visualisations. The first visualisation is a tree diagram showing groups of speed zones and the years when accidents occurred. A second connected visualisation shows a timeline displaying the trend of accident counts over each day of the week. The two visualisations are presented side-by-side. There is also a legend to describe the colour and text narration to help describe the visualisation insights.

#### **The template.**

You will be given an incomplete source code of the final visualisation. Your task is to complete the code by following the provided instructions. We refer to this incomplete source code as the template. **You are not expected to remove any aspect of this template.**

The template is a single HTML document. All references to external libraries, datasets, CSS, and the layout are already provided in the template. The layout of the

page is a side-bar layout. The left side-bar contains basic information of the visualisation, some data insights and information relating to the data source. The main canvas on the right is where the visualisation is placed. The HTML elements of the tooltip are also provided in the template.



To help you better understand the template structure, comprehensive comments are provided in the HTML elements as well as the javascript. Most of the javascript codes required to finish the visualisations are provided in the template, and are accompanied by descriptions. Most of the descriptions are represented as the `#region` and `#endregion` block of comments in the code.

There are two dataset files used in this project.

1. *PE3\_roads\_hierarchy.json*. This data set is used to create the tree diagram.
2. *PE3\_accident\_data.csv*. This data set is used to create the lines.

## Detailed tasks:

You will only need to complete the javascript parts. These parts are indicated by a comment line: **“ADD YOUR CODE HERE FOLLOWING THE INSTRUCTION”** and a list of **instructions**. There are 6 tasks that you need to complete. These tasks are based on the criteria described in Section 4. Assessment Criteria. The following is the list of these tasks.

### #Task 1: Add nodes to the network diagram

- Take the node variable specified in the code and use it to create a symbol representing each node in the tree diagram.
- The symbol should be an appropriate design choice.

### #Task 2: Label the nodes

In this

- Take the node variable specified above and use it to create a label showing the name of the node.
- The label should be placed on the left side of the node, or in the centre of the node if appropriate for your symbol choice.
- Set the class attribute of the label to *"label"*.

### #Task 3: Create the timeline bars

- The timeline bars are horizontal lines or rectangles with the same length and thickness. There should be 7 in a row representing the 7 days of the week for each year. Create the 7 timeline bars for each year.

*Hint: to create and position the timeline bars, you can append lines (or rectangles) with a stroke width of about 10px. Use the y position of the respective year node, and the x position of respective day for y and x positions of each line respectively. The length of each line should span the distance between the vertical dashed lines that separate the days. You might need to make some adjustments by adding offsets for optimal alignment.*

### #Task 4: Encode the accident count per day using colour in the timeline bars

Only the colour of the bar will encode the total number of accidents occurring on that day of week per speed zone per year. Do not hard code the accident count in your code.

- Choose a colour scheme and scale that is appropriate for the encoded variable, and enables you to differentiate the patterns in the timeline explained in the text description on the left side-bar.
- Add or modify the legend for the colour scale and any other visual variables you might add to match how you have encoded them.

### #Task 5: Show the tooltip on mouse hover, when:

1) on the timeline bars

- You will need to write code to show and hide the tooltip when the mouse pointer is on or leaving the bar.
- The tooltip should show all attributes and values in the “PE3\_accident\_data” dataset that is associated with the bar being hovered.
- These attributes are SpeedZone, Year, DayofWeek, and AccidentCount

2) on the nodes

- You will need to write code to show and hide the tooltip when the mouse pointer is

- on or leaving the node.
- The tooltip should show the year and total number of accidents when hovering a year node, and the number of child nodes if hovering a non-leaf node.

**#Task 6: Highlight the timeline bar, links, and nodes when the mouse pointer is on a node**

1. Using the node object, write code to make the following interactive features:
  - a. When the mouse pointer is hovering on the speed zone node, the bars of that speed zone are focused.
2. When the mouse pointer is hovering on a year node, the bars of that year in **all** speed zones are focused.
  - a. The focus visually stands-out by changing the opacity of other bars to a value between 0.1 and 0.5 and keeping the opacity of the current bar or branch of bars that is being focused. You can use *timeline-bar-transparent* class for this purpose.
  - b. The paths/links from the root to the nodes of the focused bars are highlighted with a thicker and a different colour. You can use *link-highlighted* class for this purpose.
  - c. An outline colour is added to the node being hovered.
3. When the mouse pointer exits, the colour and thickness of the bars, links, and nodes must come back to the original

**Remember**, you are only asked to complete the tasks above. We provided hints of the starting points for a task or a combination of the tasks with this comment: “ADD YOUR CODE HERE FOLLOWING THE INSTRUCTION”.

**General Help Notes:**

- The template uses D3 v7. This is important when referencing online resources which may use older versions of D3.
- You are allowed to make changes to the template, such as to add your own CSS rules, and adjust fonts and styling. However, any issues which arise from this (e.g., network graph or timeline no longer working, poor visual changes to the visualisation) may result in marks being deducted, even if not part of the given task.
- The datasets are stored on an online repository and are referenced via URL in the template (suffixed with “...DataPath”). You must NOT change these paths, nor should you download and use a local copy in your submission.
- Your code will be run through similarity detection software during the marking process. Be sure to acknowledge any reference sources in your code comments to avoid breaching academic integrity.
- Generative Artificial Intelligence (Generative AI) software or systems cannot be used for any part of this assessment task, including (but not limited to) generating written or visual components of your submitted work.

**3. Assessment Resources**

- The template is provided in the assessment page on Moodle.

## 4. Assessment Criteria

The following outlines the criteria which you will be assessed against.

- Ability to create simple static visualisations using D3 [3%]
  - Demonstrated ability to create basic SVG elements using D3 [1%]
  - Demonstrated ability to link data to visual properties [1%]
  - Demonstrated ability to choose appropriate visual variables to encode data [1%]
- Ability to create interactive visualisations using D3 [2%]
  - Demonstrated ability to create an interactive visualisation in D3: tooltip [1%]
  - Demonstrated ability to create an interactive visualisation in D3: highlight on mouse over [1%]

Please note that marks will be deducted if the provided template is not used, or if your submission uses modified and/or local copies of the datasets. Remember that you are allowed to improve on the template with improved aesthetics (e.g., CSS) and more advanced interactions and/or layouts if you wish, but this is not required for full marks.

## 5. How to Submit

The code for the final interactive visualisation should be submitted through Moodle. Ensure that all necessary files for the visualisation are included. The code's components (i.e., HTML, CSS, JS) can either be in the same file or separated into individual files. Data files **should not be included** as these should be accessed via the URLs provided in the template.

Submit a zip file containing all files required to run your work. Name the zip file in this format: **[STUDENT ID]\_[FIRST NAME]\_[LAST NAME].zip**

## 6. Late penalty

See the late penalty guidelines in the Assessments section on Moodle.