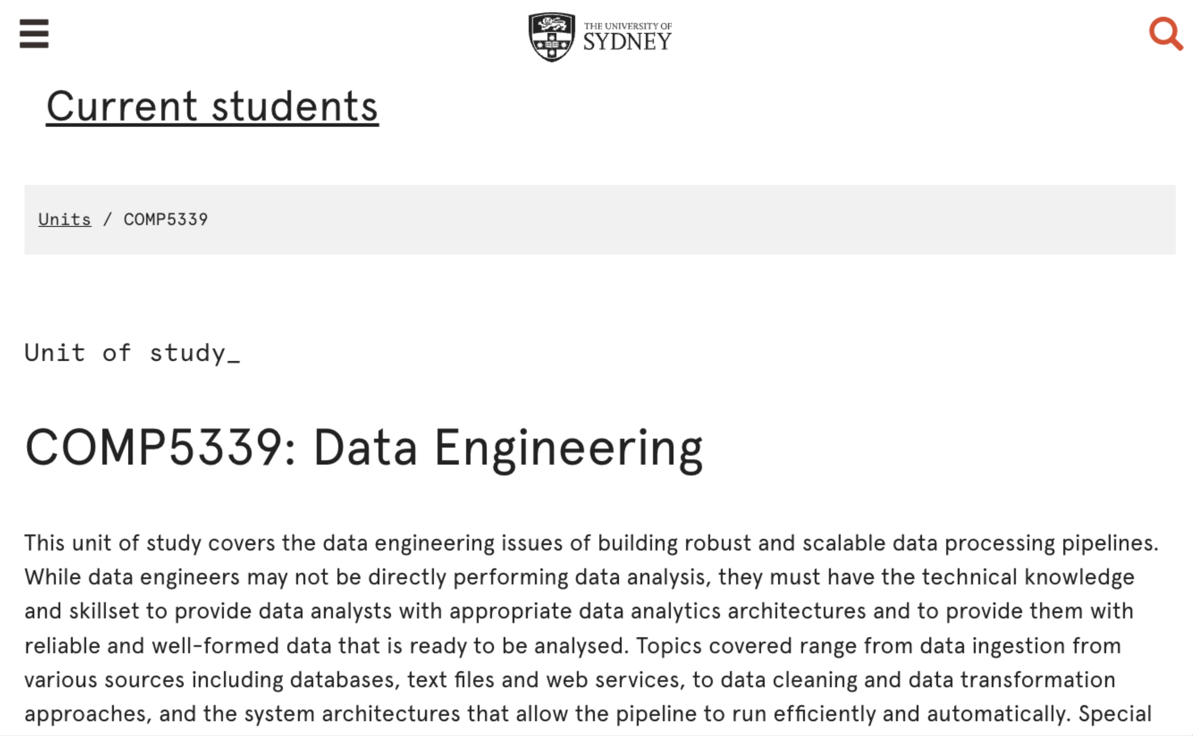
Dataset



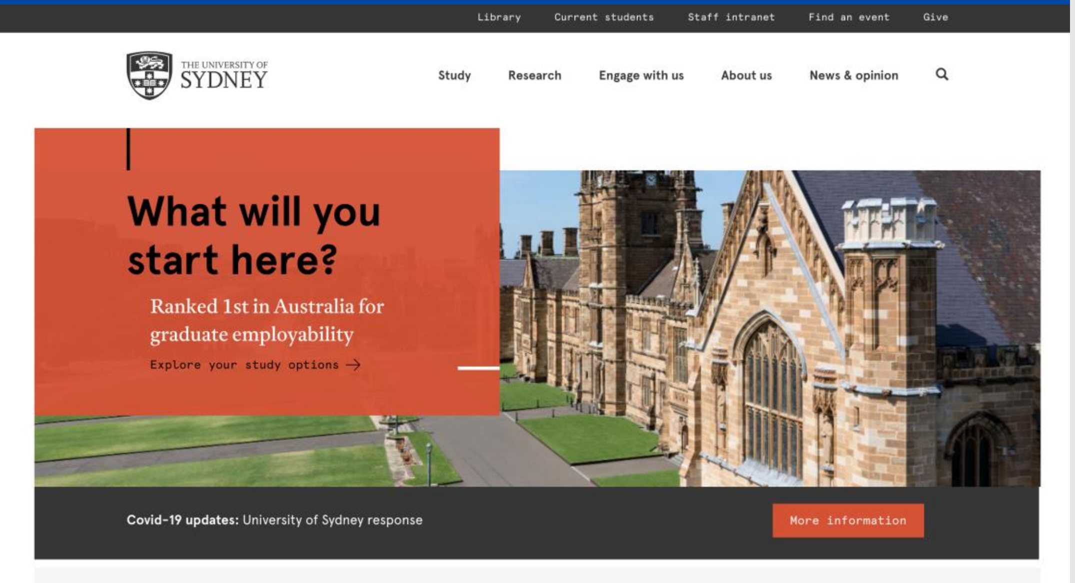
Current students

Units / COMP5339

Unit of study_

COMP5339: Data Engineering

This unit of study covers the data engineering issues of building robust and scalable data processing pipelines. While data engineers may not be directly performing data analysis, they must have the technical knowledge and skillset to provide data analysts with appropriate data analytics architectures and to provide them with reliable and well-formed data that is ready to be analysed. Topics covered range from data ingestion from various sources including databases, text files and web services, to data cleaning and data transformation approaches, and the system architectures that allow the pipeline to run efficiently and automatically. Special



What will you start here?

Ranked 1st in Australia for graduate employability

Explore your study options →

Covid-19 updates: University of Sydney response

More information

Not all data is presented as neatly as a structured dataframe of rows and columns, like the datasets we've used so far. Often, meaningful information exists in unstructured or semi-structured formats.

Today we'll explore how to extract data from **webpages**, by focussing firstly on a familiar example – the online **UoS outline** for this subject – and then investigating how this can be scaled up in application.

Webpages

- Documents intended for web browsers are written in **HTML** (HyperText Markup Language)
    - These are tree-like structures, comprising multiple elements that start and end with tags
    - e.g.

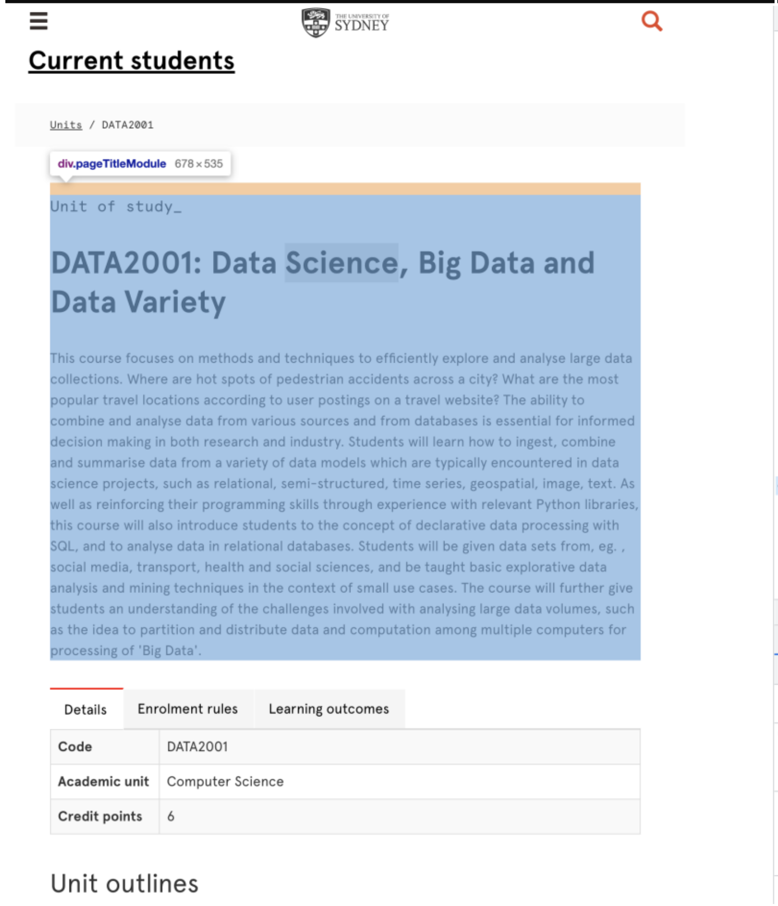
```
<div class="firstSection">
   <h1>This is a heading!</h1>
   <p>This is a paragraph containing text</p>
   <a href="https://bit.ly/3JX9nLM">This is a link</a>
</div>
```

→ this would have a heading element ('h1' tag), a paragraph ('p' tag) and a hyperlink ('a' tag), all contained within a 'div'

→ elements can have classes (not unique), or a single id (unique), which is particularly useful for setting styles with **CSS**

Inspect Element

Webpage



Current students

Units / DATA2001

div.pageTitleModule 678 × 535

Unit of study_

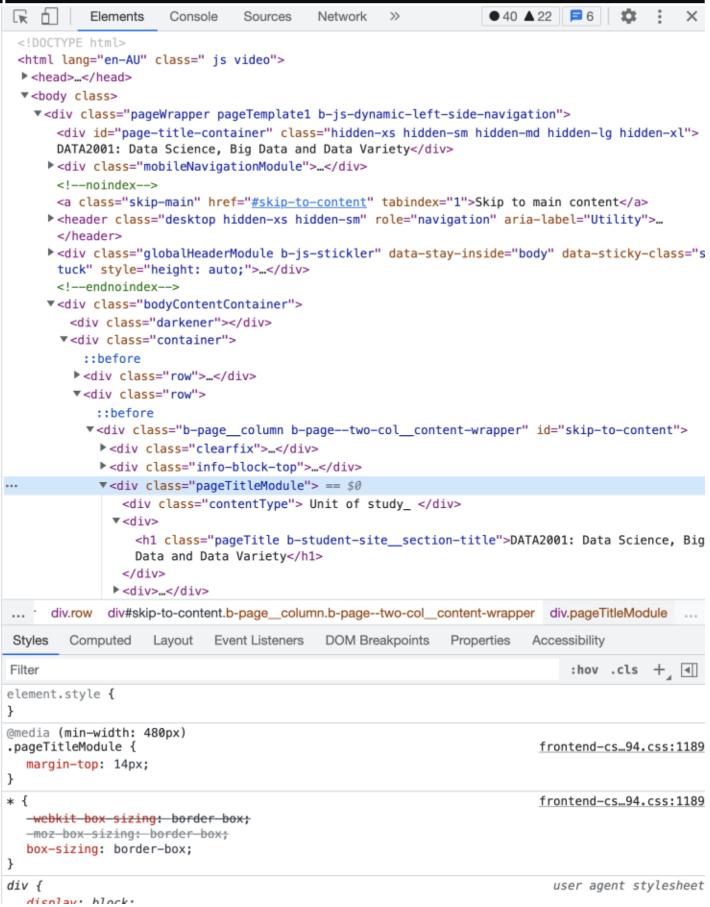
DATA2001: Data Science, Big Data and Data Variety

This course focuses on methods and techniques to efficiently explore and analyse large data collections. Where are hot spots of pedestrian accidents across a city? What are the most popular travel locations according to user postings on a travel website? The ability to combine and analyse data from various sources and from databases is essential for informed decision making in both research and industry. Students will learn how to ingest, combine and summarise data from a variety of data models which are typically encountered in data science projects, such as relational, semi-structured, time series, geospatial, image, text. As well as reinforcing their programming skills through experience with relevant Python libraries, this course will also introduce students to the concept of declarative data processing with SQL, and to analyse data in relational databases. Students will be given data sets from, eg. , social media, transport, health and social sciences, and be taught basic explorative data analysis and mining techniques in the context of small use cases. The course will further give students an understanding of the challenges involved with analysing large data volumes, such as the idea to partition and distribute data and computation among multiple computers for processing of 'Big Data'.

Code	DATA2001
Academic unit	Computer Science
Credit points	6

Unit outlines

Content

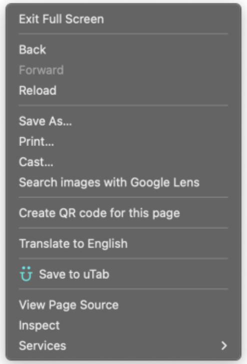


```
<!DOCTYPE html>
<html lang="en-AU" class=" js video">
 ▶<head>…</head>
 ▼<body class>
  ▼<div class="pageWrapper pageTemplate1 b-js-dynamic-left-side-navigation">
     <div id="page-title-container" class="hidden-xs hidden-sm hidden-md hidden-lg hidden-xl">
     DATA2001: Data Science, Big Data and Data Variety</div>
    ▶<div class="mobileNavigationModule">…</div>
     <!--noindex-->
     <a class="skip-main" href="#skip-to-content" tabindex="1">Skip to main content</a>
    ▶<header class="desktop hidden-xs hidden-sm" role="navigation" aria-label="Utility">…
     </header>
    ▶<div class="globalHeaderModule b-js-stickler" data-stay-inside="body" data-sticky-class="s
     tuck" style="height: auto;">…</div>
     <!--endnoindex-->
    ▼<div class="bodyContentContainer">
      <div class="darkener"></div>
     ▼<div class="container">
        ::before
      ▶<div class="row">…</div>
      ▼<div class="row">
         ::before
       ▼<div class="b-page__column b-page--two-col__content-wrapper" id="skip-to-content">
        ▶<div class="clearfix">…</div>
        ▶<div class="info-block-top">…</div>
…       ▼<div class="pageTitleModule"> == $0
           <div class="contentType"> Unit of study_ </div>
          ▼<div>
             <h1 class="pageTitle b-student-site__section-title">DATA2001: Data Science, Big
             Data and Data Variety</h1>
           </div>
          ▶<div>…</div>
```

… div.row div#skip-to-content.b-page__column.b-page--two-col__content-wrapper div.pageTitleModule …

```
element.style {
}
@media (min-width: 480px)
.pageTitleModule {                                    frontend-cs…94.css:1189
  margin-top: 14px;
}
* {                                                   frontend-cs…94.css:1189
  ~~-webkit-box-sizing: border-box;~~
  ~~-moz-box-sizing: border-box;~~
  box-sizing: border-box;
}
div {                                                 user agent stylesheet
  display: block;
```

*Right clicking anywhere on a webpage should give an "**Inspect**" or "Inspect Element" option, which reveals, and allows interaction with, the underlying source code.*

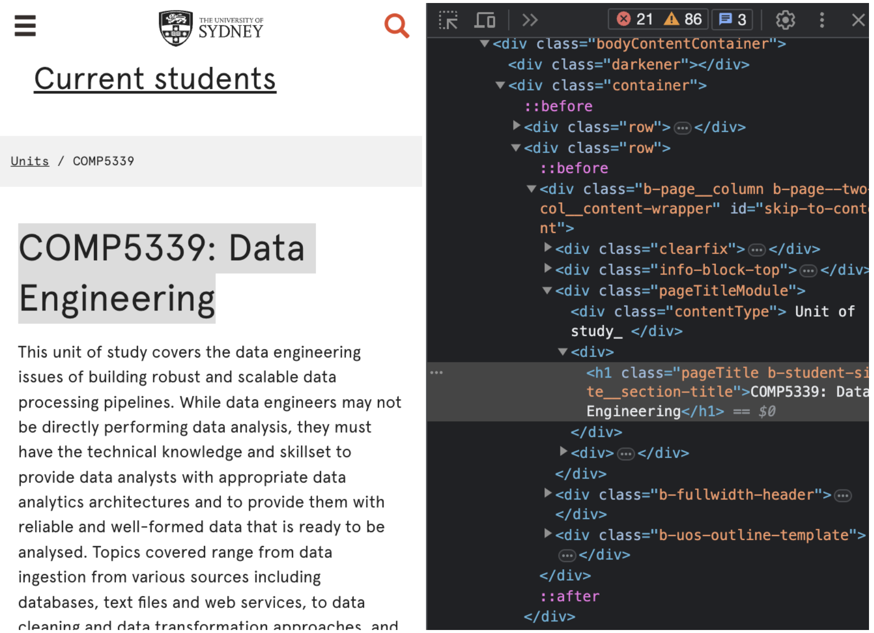


***Hovering** over an element in the HTML code will reveal where it exists on the webpage.*

*The **CSS styling** of each element can also be viewed when a HTML element is clicked. Here for example, we see an element with an upper margin of 14 pixels.*

Webpage Parsing

Jump to the Jupyter Notebook for this week and begin exploring the webpage.



THE UNIVERSITY OF SYDNEY

Current students

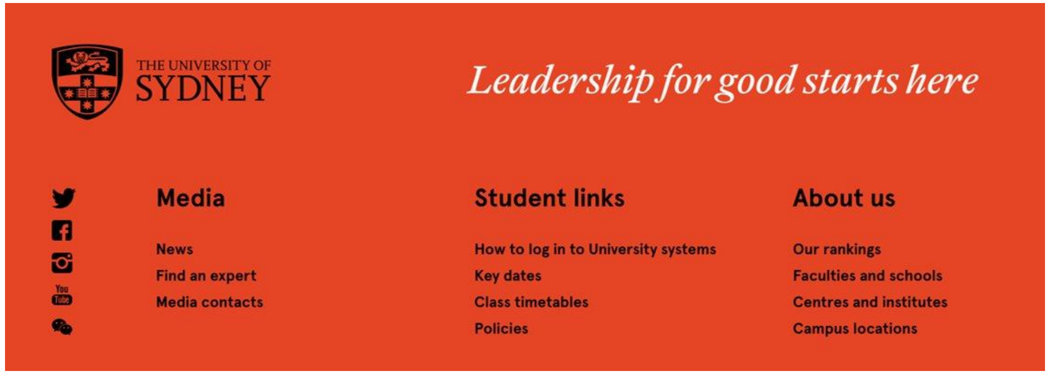
Units / COMP5339

COMP5339: Data Engineering

This unit of study covers the data engineering issues of building robust and scalable data processing pipelines. While data engineers may not be directly performing data analysis, they must have the technical knowledge and skillset to provide data analysts with appropriate data analytics architectures and to provide them with reliable and well-formed data that is ready to be analysed. Topics covered range from data ingestion from various sources including databases, text files and web services, to data cleaning and data transformation approaches, and

```
▼<div class="bodyContentContainer">
   <div class="darkener"></div>
  ▼<div class="container">
     ::before
    ▶<div class="row">⋯</div>
    ▼<div class="row">
       ::before
      ▼<div class="b-page__column b-page--two-
       col__content-wrapper" id="skip-to-conte
       nt">
        ▶<div class="clearfix">⋯</div>
        ▶<div class="info-block-top">⋯</div>
        ▼<div class="pageTitleModule">
           <div class="contentType"> Unit of
           study_ </div>
          ▼<div>
             <h1 class="pageTitle b-student-si
             te__section-title">COMP5339: Data
             Engineering</h1> == $0
           </div>
          ▶<div>⋯</div>
         </div>
        ▶<div class="b-fullwidth-header">⋯
         </div>
        ▶<div class="b-uos-outline-template">
         ⋯</div>
       </div>
       ::after
     </div>
```

Guiding screenshot for the demonstration in Section 1.3
Can we extract some attributes of the header links?



THE UNIVERSITY OF SYDNEY

Leadership for good starts here

Media

News
Find an expert
Media contacts

Student links

How to log in to University systems
Key dates
Class timetables
Policies

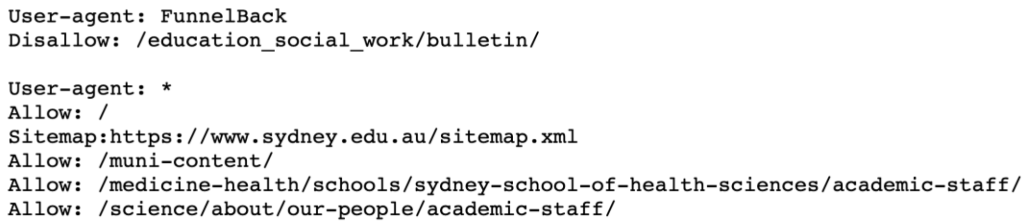
About us

Our rankings
Faculties and schools
Centres and institutes
Campus locations

Guiding screenshot for the task in Section 1.3
Can we extract the text and links for the social media platforms?

Robots.txt

The location for websites to specify what can and can't be scraped is **robots.txt**.
For any given web domain, simply put '/robots.txt' on the end. Below is Sydney University's:



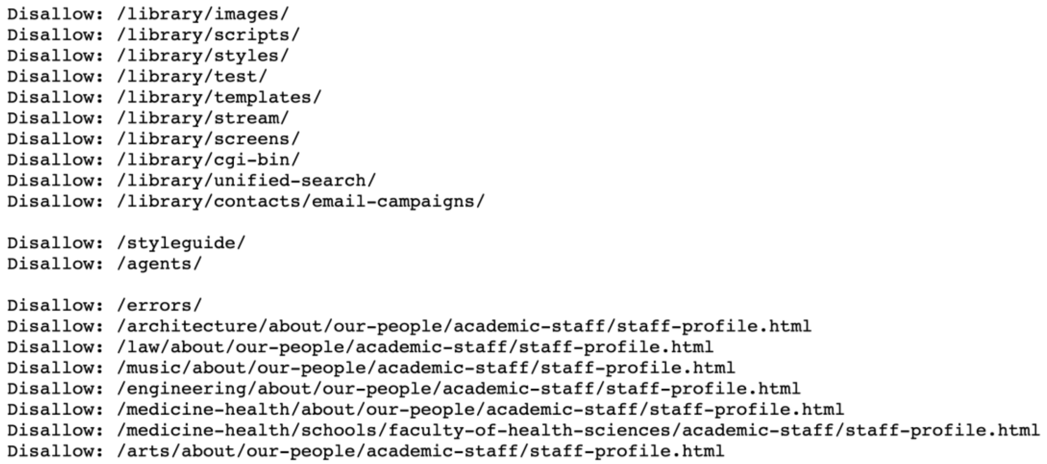
```
User-agent: FunnelBack
Disallow: /education_social_work/bulletin/

User-agent: *
Allow: /
Sitemap:https://www.sydney.edu.au/sitemap.xml
Allow: /muni-content/
Allow: /medicine-health/schools/sydney-school-of-health-sciences/academic-staff/
Allow: /science/about/our-people/academic-staff/
```

User-agent: the group the rules apply to (e.g. 'AdsBot-Google' may be subject to different terms of use, '*' indicates all other users)

It will generally then detail what is allowed/disallowed. Here, the root directory ('/') is listed as "**allow**", so our use here is permissible.

Some may even specify a "**crawl-delay**", which specifies the minimum time that must be left in between requests. Even if this is not specified, **always be sure to add delays** between requests!



```
Disallow: /library/images/
Disallow: /library/scripts/
Disallow: /library/styles/
Disallow: /library/test/
Disallow: /library/templates/
Disallow: /library/stream/
Disallow: /library/screens/
Disallow: /library/cgi-bin/
Disallow: /library/unified-search/
Disallow: /library/contacts/email-campaigns/

Disallow: /styleguide/
Disallow: /agents/

Disallow: /errors/
Disallow: /architecture/about/our-people/academic-staff/staff-profile.html
Disallow: /law/about/our-people/academic-staff/staff-profile.html
Disallow: /music/about/our-people/academic-staff/staff-profile.html
Disallow: /engineering/about/our-people/academic-staff/staff-profile.html
Disallow: /medicine-health/about/our-people/academic-staff/staff-profile.html
Disallow: /medicine-health/schools/faculty-of-health-sciences/academic-staff/staff-profile.html
Disallow: /arts/about/our-people/academic-staff/staff-profile.html
```

There will also often be sites listed as "**disallow**" that should not be visited programmatically. For USYD, this entails the pages of academic staff profiles, for example.

Thought Questions

What applications/benefits could web scraping have?

What challenges are faced when attempting to web scrape?

How do we determine what is legal, and remain a good internet citizen?