

## COM5961 DATA DRIVEN PRODUCTS & SERVICES DESIGN: LESSON 12 - DATA VISUALISATION IN JAVASCRIPT

Bernard Suen  
Center for Entrepreneurship  
Chinese University of Hong Kong

# **Today's Agenda**

1. **Use cases of front-end JavaScript**
2. **What is Key Performance Index (KPI)?**
3. **How to design a dashboard to communicate KPI?**
4. **Use cases of dashboard for your final project**
5. **Data visualization in JavaScript**

# **General Comments on Assignment#4**

1. Solution in search of a need instead of a need in search of a solution.
2. A need has to be possessed by someone. That ‘someone’, the target persona, will be the ultimate judge on all the assumptions you made about the problem and its solution.
3. Always start small with a narrow and well-defined scope and expand from there given the feedback received favours expansion.
4. Nice Figma work but don’t assume things you can do in Figma will carry over into your web design done in HTML/CSS/JS.

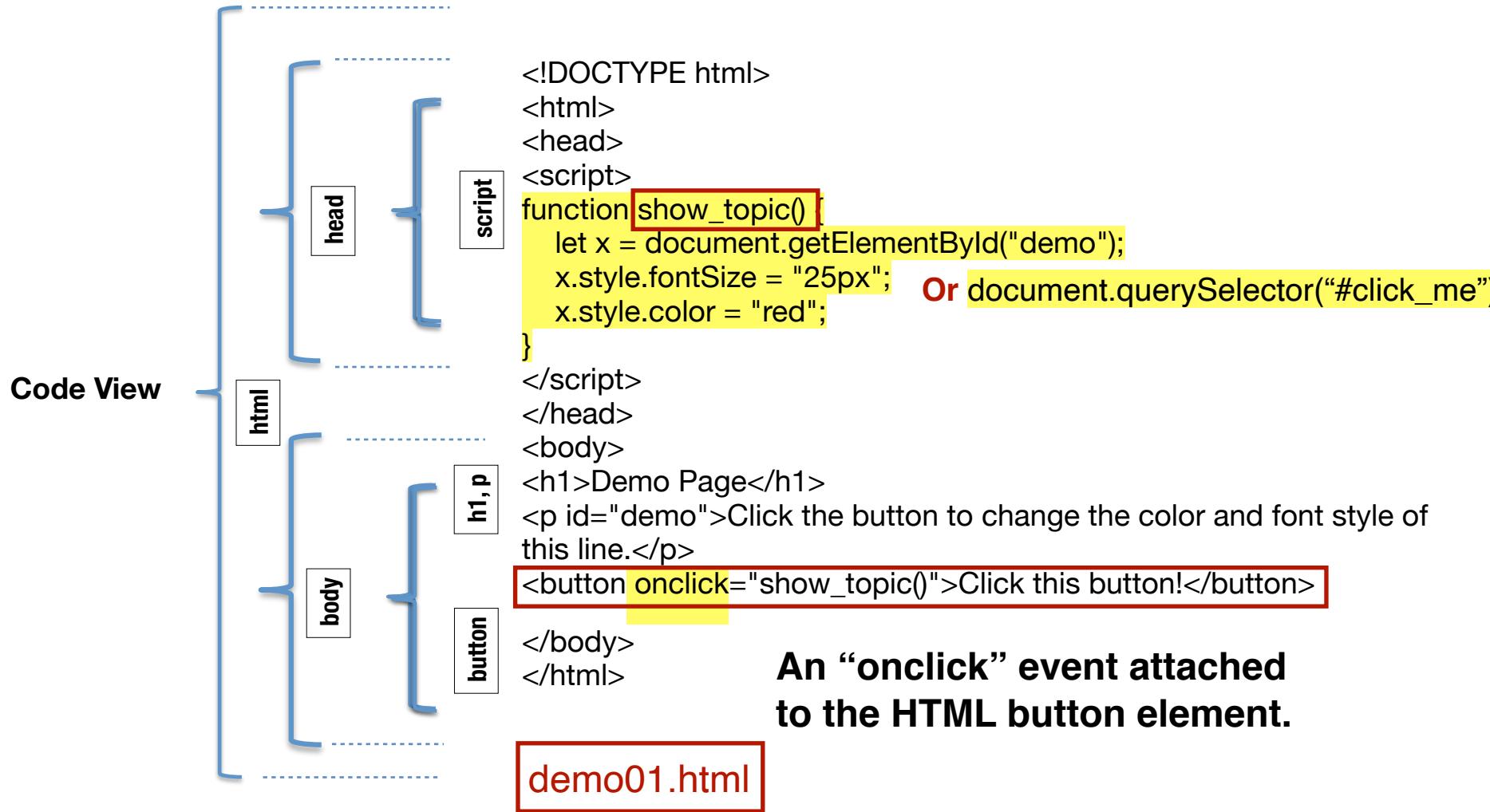
5. Transformation of the Figma prototype into front-end/back-end codes depends on the combined strengths of the product team. As a PM, you need to know the full picture so you can put together the right team for executing the appropriate codes.
6. Adding and counting comments or views entails one-to-many data entities relationship, which is not that hard to understand. However, adding and tracking bookmarks or followers involve many-to-many relationship, which is more complicated.
7. Afraid to ask for money in usability test, which is a fear PM has to overcome to get to the heart of the matter when it comes to user validation.



Source: <https://www.youtube.com/watch?v=WAXLTG9n7Kw>

# **Quick Review of Basic JavaScript Event Listener and Form Handler**

# The **onclick** Method



# The **addEventListener** Method

```
var var_name = document.querySelector("selector")  
var_name.addEventListener("event type", function_name)
```

### **Example:**

```
<body>  
<div id="click_me">Click me</div>
```

```
<script>      Or document.getElementById("click_me")  
let click_name = document.querySelector("#click_me")  
click_me.addEventListener("click", alert_click)
```

```
function alert_click() {  
    alert("Hi!")  
}  
</script>
```

**demo02.html**

```
var var_name = document.querySelector("selector")  
var_name.addEventListener("event type", function_name)
```

### **Example:**

```
<body>  
<div id="click_me">Click me</div>
```

```
<script>      Or document.getElementById("click_me")  
let click_name = document.querySelector("#click_me")  
click_me.addEventListener("click", alert_click)
```

```
function alert_click() {  
    window.location.href='https://www.cuhk.edu.hk';}  
</script>  
</body>
```

# **HTML Form Handling in JavaScript**

**User Name:**

**Password:**

**User Name:** bernard

**Password:** .....

Hello bernard! Your password is password.

```
<div class='container center_container'>
    <form id='login'>
        <div class='form-group'>
            <label>
                User Name:
            </label>
            <input type='text' name='username'>
        </div>
        <div class='form-group'>
            <label>
                Password:
            </label>
            <input type='password' name='pwd'>
        </div>
        <button id='submitBtn' type='button'>Send</button>
    </form>
    <div id='answer'></div>
    <div id="demo"></div>
</div>

<script>
document.getElementById('submitBtn').addEventListener('click', sayHello);

function sayHello() {
    var username = document.forms[0].username.value;
    var password = document.forms[0].pwd.value;
    if (username == '') {alert('username cannot be blank!');}
    if (password == '') {alert('password cannot be blank!');}
    if (username !== '' && password !== '') {
        alert("Hello " + username + "! Your password is " + password + '.');
        var answer = "Hello " + username + "! Your password is " + password + '.';
        document.getElementById('answer').innerHTML = answer;
    }
}
</script>
```

demo04.html

# **Using JS to handle form in Flask**

## **html template**

```
{% block content %}  
    <!-- Main Content-->  
    <div class="container">  
        <form action="/login" method="POST" name="form">  
            <div class="form-group">  
                <label for="username">User name</label>  
                <input type="text" class="form-control" id="username" placeholder="Enter username" name="username">  
            </div>  
            <div class="form-group">  
                <label for="email">Email address</label>  
                <input type="email" class="form-control" id="email" placeholder="Enter email" name="email">  
            </div>  
            <div class="form-group">  
                <label for="password">Password</label>  
                <input type="password" class="form-control" id="password" placeholder="Enter password" name="password">  
            </div>  
            <button type="submit" id="submitBtn" class="btn btn-primary">Submit</button>  
        </form>  
        <br>  
        <center>Don't have an account? <a href="/register">Sign up here.</a></center>  
    </div>  
    <hr />
```

Flask route

Pre-process “checkForm” before submitting to “/login”. Submit only when the “return” condition is true.

id=“submitBtn” is the trigger for activating the function.

login.html

```
document.getElementById('submitBtn').addEventListener('click', checkForm);

function checkForm(event) {
    if (form.password.value.length == 0) {
        alert("Error: Password is empty!");
        form.password.focus();
        event.preventDefault();
        return false;
    }

    let re = /^[a-zA-Z0-9]+$/; // test alpha_numeric characters
    if (form.username.value == "") {
        alert("Error: Username is empty!");
        form.username.focus();
        event.preventDefault();
        return false;
    }

    if (!re.test(form.username.value)) {
        alert("Error: Username contains invalid characters!");
        form.username.focus();
        event.preventDefault();
        return false;
    }

    if (form.email.value == "") {
        alert("Error: Email is empty!");
        form.email.focus();
        event.preventDefault();
        return false;
    }

    if (!re.test(form.email.value)) {
        alert("Error: Incorrect email format!");
        form.email.focus();
        event.preventDefault();
        return false;
    }
}

return true;

```

Add Event listener to detect the ‘click’ event trigger for form submission button (i.e. submitBtn)

When the submitBtn click event is triggered, evoke the “checkForm” function.

The checkForm function tests the entries with conditionals to ensure there is no blank field and the data format (i.e. alphanumeric characters) is correctly entered for the username and email fields.

login.html

**Define display area through selector.**

Home page

## Home Page

### Simple Demo of JavaScript and JQuery

Output area (i.e. id="demo")

Welcome to HTML/CSS/JS play

p#demo 1100 x 20

This is a simple demonstration of JavaScript and JQuery providing the simple CSS styling.

X + Y = ?

Another X + Y = ?

Change Text Color of Welcome Message

Enter first number:

Enter second number:

Elements    Console    Sources    Performance insights    Network    Performance    Memory    Application    >

Styles    Computed    Layout    >

Filter    :hover .cls +   

ent.style {

p { margin: 0 0 10px; } \* { -webkit-box-sizing: border-box; -moz-box-sizing: border-box; box-sizing: border-box; }

bootstrap.min.css:5  
bootstrap.min.css:5  
100  
00  
10

Define output area id="demo".

```
::before
<div class="col-sm-12">
  <h1>Home Page</h1>
  <h2>Simple Demo of JavaScript and JQuery</h2>
  <p id="welcome">Welcome to HTML/CSS/JS play</p>
<div class="well">
  <p id="demo"> == $0
    "This is a simple demonstration of JavaScript and JQuery providing the simple CSS
    styling."
  </p>
</div>
<!-- Sign and date the page, it's only polite! --&gt;
html body div.container div.row div.col-sm-12 div.well p#demo</pre>
```

Home page

# Home Page

## Simple Demo of JavaScript and JQuery

Welcome to HTML/CSS/JS play

The answer is 50

X + Y = ?

Another X + Y = ?

Change Text Color of Welcome Message

Enter first number:

20

Enter second number:

30

Form receiving the input values.

```
<button type="button" class="btn btn-success" onclick="show_topic();>Change Text  
Color of Welcome Message</button>
```

```
<form id="myForm"> == $0  
  <div class="form-group">  
    <label for="1st">Enter first number:</label>  
    <input type="text" class="form-control" id="1st_num">  
  </div>  
  <div class="form-group">...</div>  
</form>
```

Storing 1st number value into input box with id="1st\_num".

```
Filter :hov .cls + , □ □  
element.style {  
}  
* {  
  box-sizing: border-box;  
}  
form {  
  user agent stylesheet  
  display: block;
```



Elements

Console

Sources

Performance insights

Network

Performance

Memory

Application

»

✖ 1

☰ 1



...

html body div.container div.row div.col-sm-12 form#myForm

Home page

## Home Page

### Simple Demo of JavaScript and JQuery

Welcome to HTML/CSS/JS play

The answer is 50

X + Y = ?

Another X + Y = ?

Change Text Color of Welcome Message

Enter first number:

20

Enter second number:

30

Elements

Console

Sources

Performance insights

Network

Performance

Memory

Application

>

x 1

✉ 1

⚙

```
function ask_xy() {  
    var x = document.getElementById("1st_num").value;  
    var y = document.getElementById("2nd_num").value;  
    var z = compute_x_plus_y(x,y);  
    var x = document.getElementById("demo");  
    x.style.fontSize = "25px";  
    x.style.color = "red";  
    document.getElementById("demo").innerHTML = "The answer is " + z;  
}
```

Assign value stored in input box (id="1st\_num") to x and  
value stored in input box (id="2nd\_num") to y.

Pass x and y values to function  
compute\_x\_plus\_y and store result in z.

matching selector or style

# Home Page

## Simple Demo of JavaScript and JQuery

Welcome to HTML/CSS/JS play

The answer is 50

X + Y = ?    Another X + Y = ?    Change Text Color of Welcome Message

Enter first number:  
20

Enter second number:  
30

Elements

Console

Sources

Performance insights

Network

Performance

Memory

Application

»

x 1 F 1 G 1

```
function compute_x_plus_y(x,y) {  
    var z = 0;  
    x = parseInt(x);  
    y = parseInt(y);  
    z = x + y;  
    return z;  
}
```

Pass x and y values to function  
compute\_x\_plus\_y and store result in z.

```
function ask_for_x_n_y() {  
    var x = prompt("Enter x value");  
    var y = prompt("Enter y value");  
}
```

Styles Computed Layout »

Filter :hov .cls +

element.style {  
}

\* { bootstrap.min.css:5  
 -webkit-box-sizing: border-box;  
 -moz-box-sizing: border-box;  
 box-sizing: border-box;  
}

form { user agent stylesheet  
 display: block;

html body div.container div.row div.col-sm-12 form#myForm

# **Key Performance Index**

# A Good KPI Must:

- Be a calculated number; RAPR
- Be comparable over time
- Track Events which *Cause* performance, not just the *Effect*
- Be related to business goals and objectives

## **Key Performance Indicators**

- a. Set goals and quantify outcome for meeting the goals
- b. Outcome should be actionable behaviour that can be observed by humans or machines
- c. Track causes not only effects to understand the “why” as well as the “how”
- d. Determine measurement period
- e. Determine unit of measurement
- f. Derive metrics and KPI from units of measurement

**Dashboard Design Principle:**  
**To *delight* and to *inform*.**

## **1. General Dashboard Design Principles**

- a. **Determine the user** (**who**--target audience) and his/her needs (**why**--purpose) in using the product/service (**what** medium and content)
- b. **Make sure you have the right metrics** (i.e. KPI) including its data preparation before you begin designing the dashboard (basic units of measure vs. ratios/formula)
- c. **Choose the right chart type and encoding principles**
- d. **Provide context suitable for the medium** (e.g. desktop vs. mobile)
- e. **Makes every pixel counts. Less is more.**

## **2. Dashboard Types (By Management Level)**

- a. **Operational** – How often (hourly, daily, weekly, monthly) to see historical and current figures such as page views, average session duration time, bounce rates.
- b. **Tactical** – Explore with interactive and drill-down capabilities to see the details such as user acquisition by channels, content and behaviour flow through website
- c. **Strategic** – Conversion and financial data such as conversion rate and financial ratios.

#### **4. Make Data Relevant to the User**

- a. **Provide context** (e.g. medium and background) to understand the pre-condition of the data source
- b. **Profile user** (e.g. demographics data)

#### **5. Simplify the Dashboard**

- a. Group and show the data with the purpose to make it easy for the user to take action with the knowledge from the data
- b. Simplify but not distort relationship (remove unnecessary noise in the data that may cause confusion)
- c. Use of interactivity to drill down to details instead of showing too much information up front

#### **6. Frequency of Update**

- a. Real-time vs. Occasional
- b. Hourly, Daily, Weekly, Monthly

# **Data-Ink Ratio**

# Design Principles

5. Consider your medium.

Real estate is even more precious. Hide secondary data.



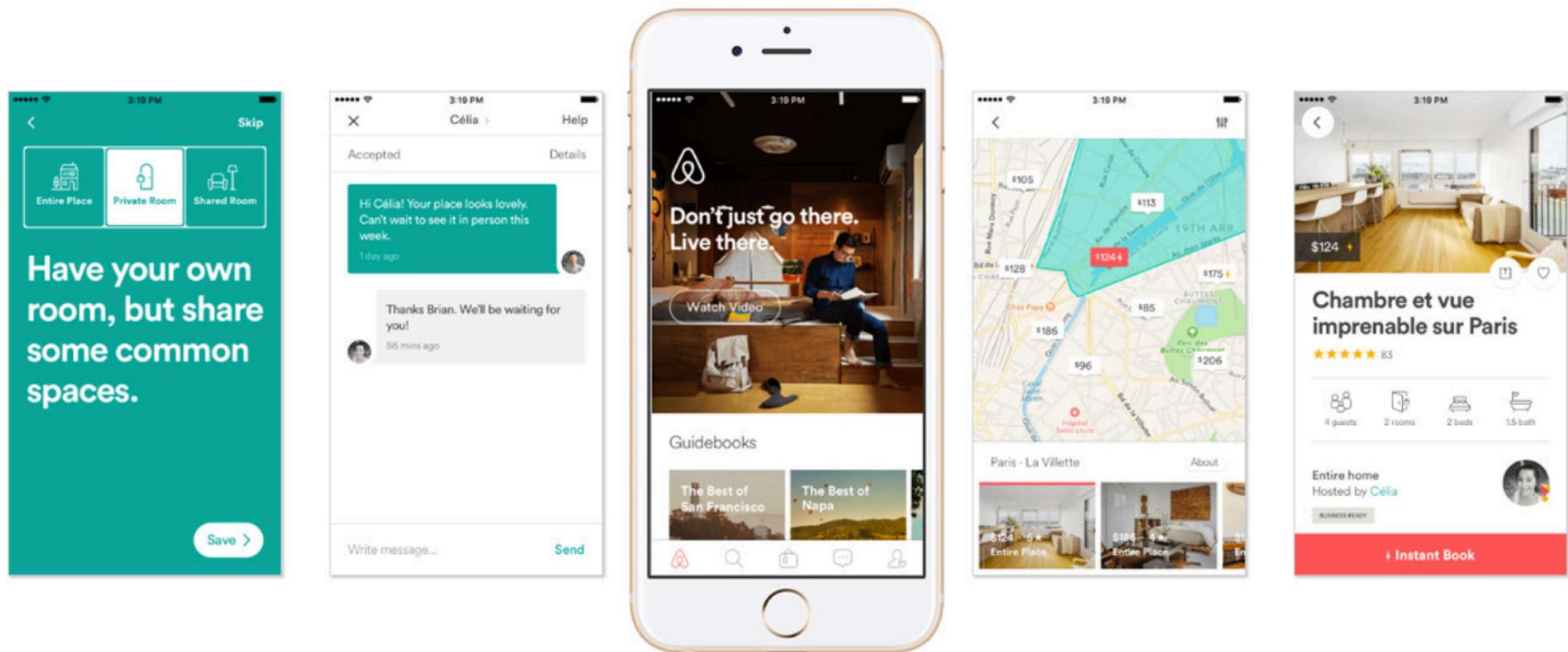
<https://www.youtube.com/watch?v=RtKDSfWFQIA>

## Lesson learned:

1. Minimalism at play — every pixel counts. Nothing more, nothing less.
2. Consider the medium, which provides the context and constraint for communicating the key message and supporting the interactions. Make sure your dashboard works on mobile phone as well as the desktop.
3. Help to declutter the screen so the audience can focus on the most important elements on the page without distractions.

# **Implementation of Dashboard**





Source: Airbnb



# Data-Driven Documents

A small orange rectangular button with white text and a white stitched border. The text reads "Fork me on GitHub".



Like visualization and creative coding? Try interactive JavaScript notebooks in [Observable](#)!

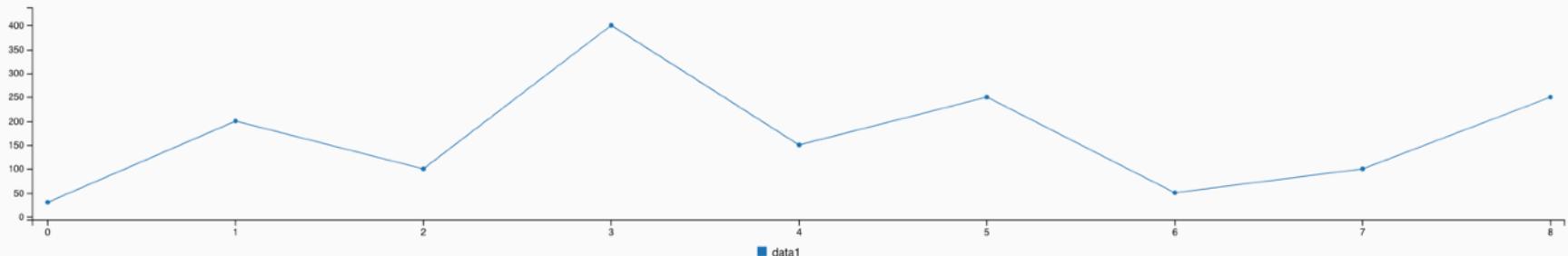
**D3.js** is a JavaScript library for manipulating documents based on data. **D3** helps you bring data to life using HTML, SVG, and CSS. D3's emphasis on web standards gives you the full capabilities of modern browsers without tying yourself to a proprietary framework, combining powerful visualization components and a data-driven approach to DOM manipulation.

[Download the latest version \(7.6.1\) here.](#)

-  See more examples
  -  Chat with the community
  -  Follow announcements
  -  Report a bug
  -  Ask for help

Source: <https://d3js.org/>

# C3.js D3-based reusable chart library

[Start Demo](#)

## Why C3?

### Comfortable

C3 makes it easy to generate D3-based charts by wrapping the code required to construct the entire chart. We don't need to write D3 code any more.

### Customizable

C3 gives some classes to each element when generating, so you can define a custom style by the class and it's possible to extend the structure directly by D3.

### Controllable

C3 provides a variety of APIs and callbacks to access the state of the chart. By using them, you can update the chart even after it's rendered.

<https://c3js.org/>

# Musings of a Strange Loop

Vainolo's Blog



Home  
All Posts  
Tutorials  
My Notebooks  
Research  
Reading List  
About

Buy Me a Coffee

All content on this site is provided as-is and is in no way the opinion of my previous, current, or future employers. Take a look at my [privacy policy](#) for more information.

When I learn something I like to write about it because it helps me remember things better, it creates an archive where I can go back and see how things are done (my way :-)), and it's a good way to share what I learn. So here it is. The code for this lesson (and all future lessons) is located here: <https://github.com/vainolo/learning-c3>

One of the things I really like about JavaScript is that you can start with a text editor and a browser and create incredibly complex things, like in this case. I'll start by creating a simple line chart from data embedded in the same `HTML` page where the chart is shown. The code is very simple:

```
1 <html>
2   <head>
3     <script src="https://unpkg.com/d3@5.12.0/dist/d3.min.js"></script>
4     <link href="https://unpkg.com/c3@0.7.11/c3.min.css" rel="stylesheet">
5     <script src="https://unpkg.com/c3@0.7.11/c3.min.js"></script>
6   </head>
7   <body>
8     <div id="chart"></div>
9     <script>
10    var chart = c3.generate({
11      bindto: '#chart',
12      data: {
13        columns: [
14          ['data1', 50, 20, 10, 40, 15, 25]
15        ]
16      }
17    });
18    </script>
19  </body>
20 </html>
```

The code is based on the C3.js getting started with a small twist – no need to download code and build dependencies as the code is referenced directly from [unpkg.com](https://unpkg.com). Probably not what you would do in production but for learning it's very powerful.

Next I create a `div` that will contain the chart, and lastly generate the chart using `c3.generate`. Here's the resulting chart:

Just like this,  
out of the box,  
we get



Source: <https://vainolo.com/2019/10/12/learning-c3-js-lesson-1-getting-started/>

```
<html>
  <head>
    <script src="https://unpkg.com/d3@5.12.0/dist/d3.min.js"></script>
    <link href="https://unpkg.com/c3@0.7.11/c3.min.css" rel="stylesheet">
    <script src="https://unpkg.com/c3@0.7.11/c3.min.js"></script>
  </head>
  <body>
    <div id="chart"></div>
    <script>
      var chart = c3.generate({
        bindto: '#chart',
        data: {
          columns: [
            ['data1', 50, 20, 10, 40, 15, 25]
          ]
        }
      });
    </script>
  </body>
</html>
```

**Id Selector “chart” is used here to define location for displaying the chart.**

**C3 chart is a data structure presented by object of objects to be linked to selector.**

**Define data structure as objects having array based value pairs (e.g. columns as key to array)**

**chart01.html**

**Default chart type is line chart.**

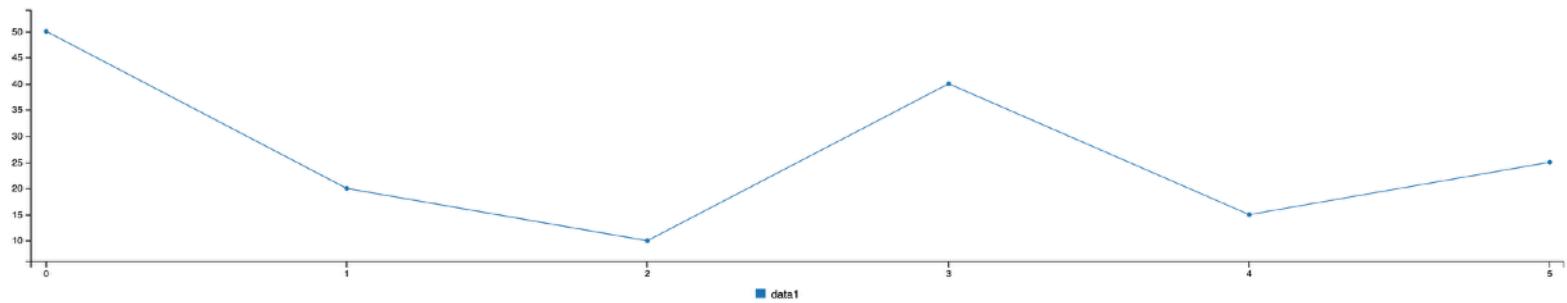


chart01.html

# # Chart

## Line Chart

Line chart with sequential data.

[View details »](#)

## Simple XY Line Chart

Simple line chart with custom x.

[View details »](#)

## Step Chart

Display as Step Chart.

[View details »](#)

## Bar Chart

Display as Bar Chart

[View details »](#)

## Timeseries Chart

Simple line chart with timeseries data.

[View details »](#)

## Multiple XY Line Chart

Multiple line chart with multiple custom x.

[View details »](#)

## Area Chart

Display as Area Chart.

[View details »](#)

## Stacked Bar Chart

Display as Stacked Bar Chart.

[View details »](#)

## Spline Chart

Display as Spline Chart.

[View details »](#)

## Line Chart with Regions

Set regions for each data with style.

[View details »](#)

## Stacked Area Chart

Display as Stacked Area Chart.

[View details »](#)

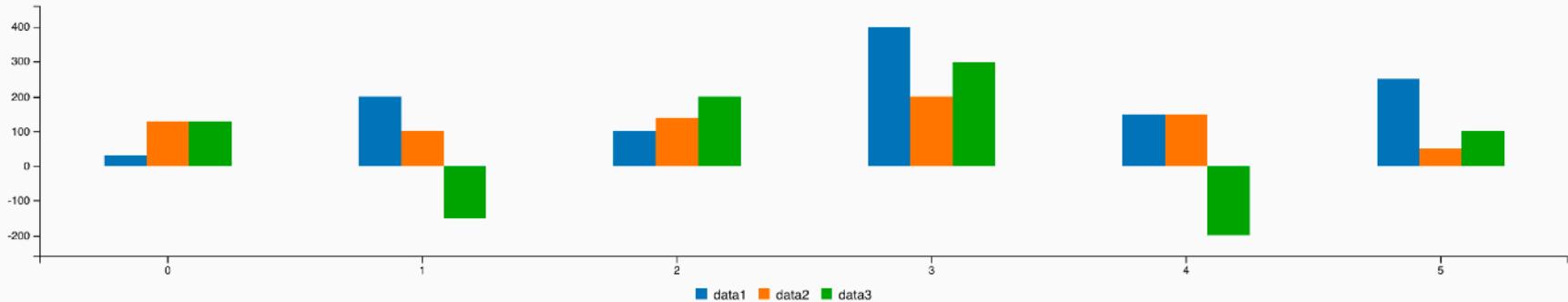
## Scatter Plot

Display as Scatter Plot.

[View details »](#)

<https://c3js.org/examples.html#chart>

## Bar Chart



```
# chart_bar.js
```

```
var chart = c3.generate({
  data: {
    columns: [
      ['data1', 30, 200, 100, 400, 150, 250],
      ['data2', 130, 100, 140, 200, 150, 50]
    ],
    type: 'bar'
  },
  bar: {
    width: {
      ratio: 0.5 // this makes bar width 50% of length between ticks
    }
  }
});
```

Define data structure as objects having array based value pairs (e.g. data1 and data2).

Define chart type as bar chart and set bar width.

[https://c3js.org/samples/chart\\_bar.html](https://c3js.org/samples/chart_bar.html)

# Picking Chart Colors



HTML COLOR CODES



Get HTML color codes, Hex color codes, RGB and HSL values with our color picker, color chart and HTML color names. Let's go!



Choose Wix to Build Your Site

WIX.com

Create a Professional Online Presence for Yourself or Your Business with Wix. Start Today!

OPEN



HEX #FF5733

RGB 255, 87, 51

HSL 11, 80%, 60%



Color 1 Color 2 Color 3 Color 4 Color 5 Color 6



#### HELPFUL TIPS



Picking

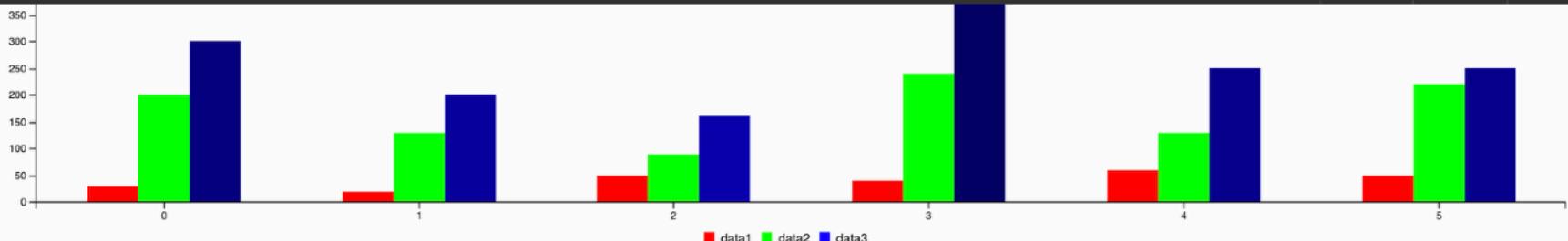


Saving



Exporting

<https://htmlcolorcodes.com/>



```
# data_color.js
```

```
var chart = c3.generate({
  data: {
    columns: [
      ['data1', 30, 20, 50, 40, 60, 50],
      ['data2', 200, 130, 90, 240, 130, 220],
      ['data3', 300, 200, 160, 400, 250, 250]
    ],
    type: 'bar',
    colors: {
      data1: '#ff0000',
      data2: '#00ff00',
      data3: '#0000ff'
    },
    color: function (color, d) {
      // d will be 'id' when called for legends
      return d.id && d.id === 'data3' ? d3.rgb(color).darker(d.value / 150) : color;
    }
});
```

Define data structure and chart type.

Overwrite default color.

[https://c3js.org/samples/data\\_color.html](https://c3js.org/samples/data_color.html)

```
var chart = c3.generate({
  data: {
    columns: [
      ['data1', 30, 200, 100, 400, 150, 250],
      ['data2', 130, 100, 140, 200, 150, 50]
    ],
    type: 'bar',
    colors: {
      data1: '#7FB3D5',
      data2: '#598EAE'
    }
  },
  bar: {
    width: {
      ratio: 0.5 // this makes bar width 50% of length between ticks
    },
    // or
    //width: 100 // this makes bar width 100px
  }
});
```

Define data structure and chart type.

Overwrite default color.

chart02.html

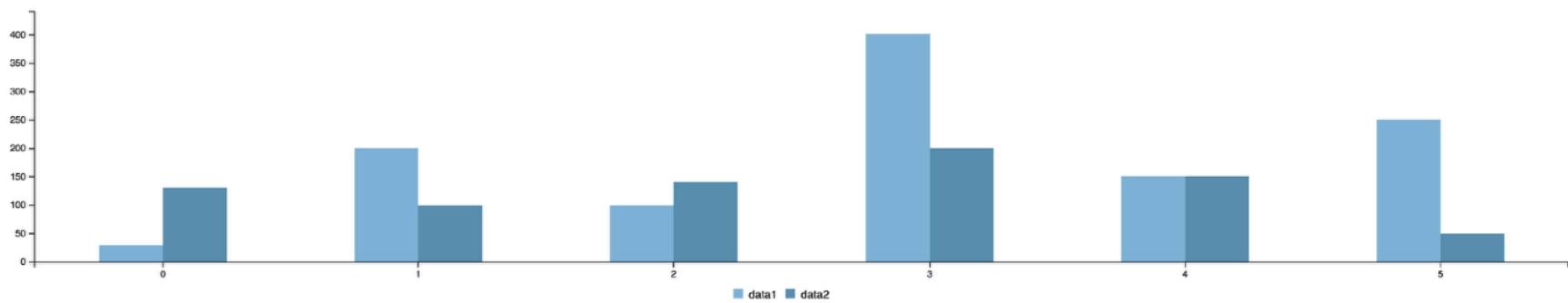


chart02.html

# **Geodata Mapping**

# **Using Leaflet and Open Street Map**



an open-source JavaScript library  
for mobile-friendly interactive maps



[Overview](#) [Tutorials](#) [Docs](#) [Download](#) [Plugins](#) [Blog](#)

May 8, 2019 — [Leaflet 1.5.1](#) has been released!

Leaflet is the leading open-source JavaScript library for mobile-friendly interactive maps. Weighing just about 38 KB of JS, it has all the mapping [features](#) most developers ever need.

Leaflet is designed with *simplicity, performance* and *usability* in mind. It works efficiently across all major desktop and mobile platforms, can be extended with lots of [plugins](#), has a beautiful, easy to use and [well-documented API](#) and a simple, readable [source code](#) that is a joy to [contribute](#) to.

<https://leafletjs.com/>

**Declare the parameters used by the map and the source of the map tiles.**

Search

Where is this?

Go



## Welcome to OpenStreetMap!

OpenStreetMap is a map of the world, created by people like you and free to use under an open licence.

Hosting is supported by [UCL](#), [Bytemark Hosting](#), and other [partners](#).

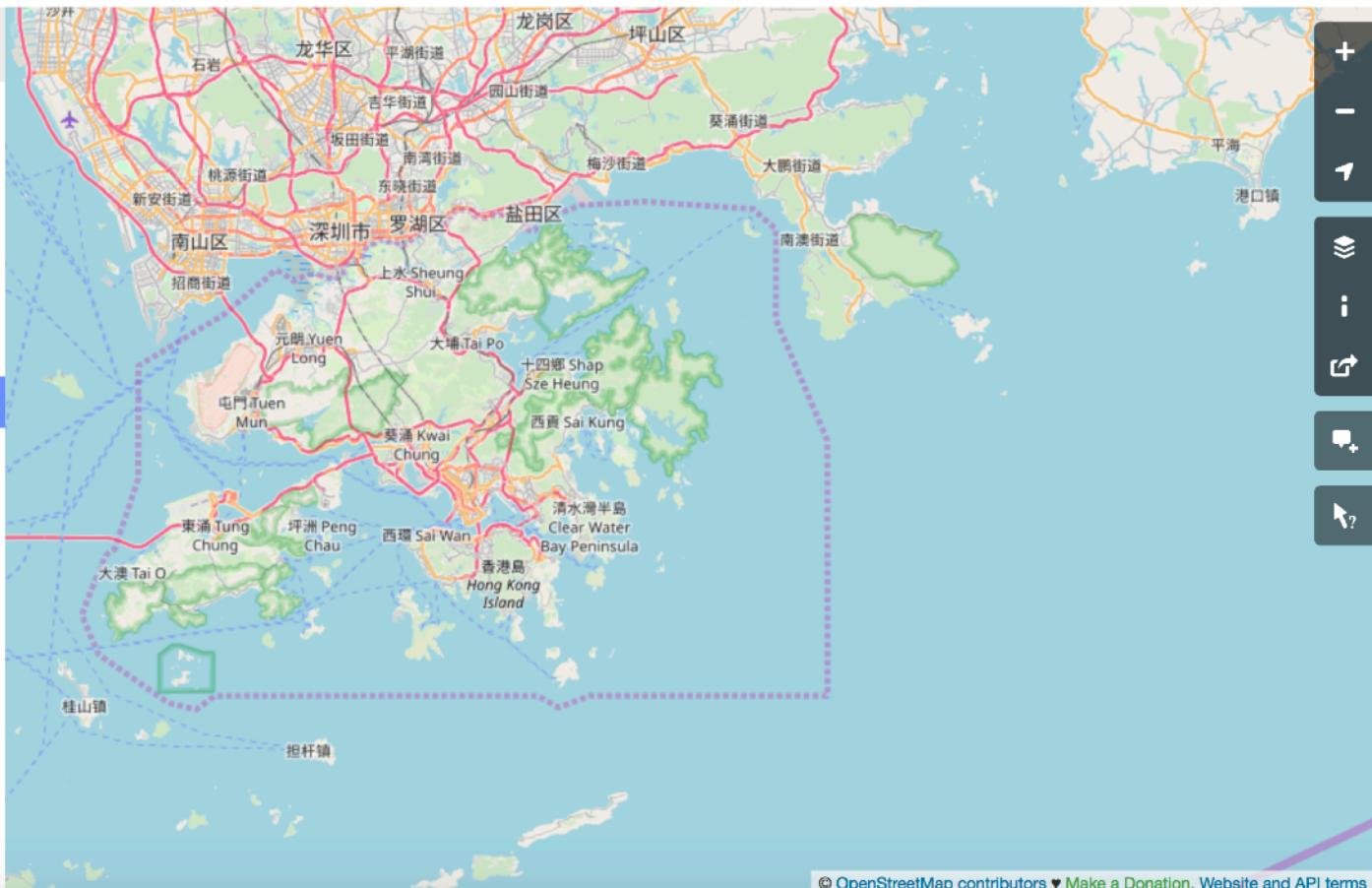
[Learn More](#)[Start Mapping](#)

# SOTM



Dhaka, Bangladesh 2019

10 km  
5 mi



© OpenStreetMap contributors • [Make a Donation](#). [Website](#) and [API terms](#)

<https://www.openstreetmap.org>



## Leaflet Tips and Tricks

[Buy on Leanpub](#)

## Table of Contents

### Acknowledgements

Make sure you get the most up to date copy of Leaflet Tips and Tricks

### Introduction

### What is leaflet.js?

### What do you need to get started?

## Start With a Simple Map

We will walk through the building of a simple web page using an HTML file. The page will host a Leaflet map so that we can understand all the different portions of the file and the processes that needs to be gone through to make it happen.

- This wont be an *exact* template for building on, since there are some liberties that are taken, but it will do the job and provides a base for explanation of the process and it will demonstrate how easy it can be.

The following is the full code listing of the file `simple-map.html` that we will be examining:

```
<!DOCTYPE html>
<html>
<head>
  <title>Simple Leaflet Map</title>
  <meta charset="utf-8" />
  <link
    rel="stylesheet"
    href="http://cdn.leafletjs.com/leaflet-0.7/leaflet.css"
  />
</head>
```

<https://unpkg.com/leaflet@1.9.3/dist/leaflet.css>

[https://leanpub.com/leaflet-tips-and-tricks/  
read#leanpub-auto-start-with-a-simple-map](https://leanpub.com/leaflet-tips-and-tricks/read#leanpub-auto-start-with-a-simple-map)

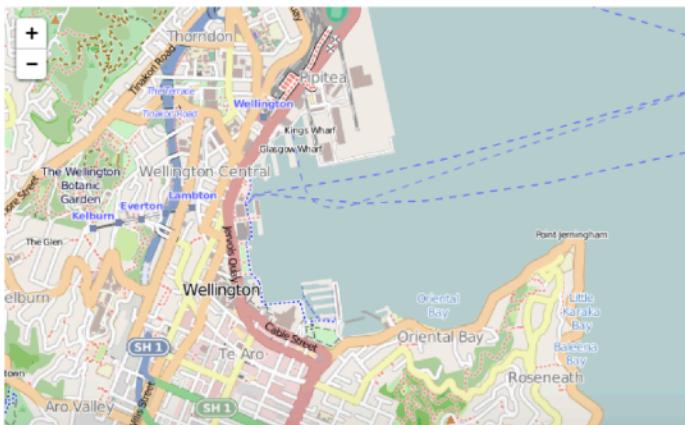
```

<!DOCTYPE html>
<html>
<head>
    <title>Simple Leaflet Map</title>
    <meta charset="utf-8" />
    <link
        rel="stylesheet"
        href="https://unpkg.com/leaflet@1.9.3/dist/leaflet.css" <-- 1
    />
</head>
<body>
    <div id="map" style="width: 600px; height: 400px"></div>
    <script
        src="https://unpkg.com/leaflet@1.9.3/dist/leaflet.js" <-- 2
    </script>

    <script>
        var map = L.map('map').setView([-41.2858, 174.78682], 14);
        mapLink =
            '<a href="http://openstreetmap.org">OpenStreetMap</a>';
        L.tileLayer(
            'http://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
                attribution: 'Map data © ' + mapLink,
                maxZoom: 18,
            }).addTo(map);
    </script>
</body>
</html>

```

The output that it will produce on a web page will look like this;



1. Define the selector (i.e. `div id="map"`) for inserting the map into the DOM.
2. Initialize map view and tile layer.
3. Source of the map tiles (e.g. Open Street Map).

## Adding a marker to our map

At some stage we will most likely want to add a marker to our map to pinpoint something. Leaflet makes the process nice and easy by including a [marker function with several options](#);

In its most simple form the following is the full code to show a map with a marker;

```
<!DOCTYPE html>
<html>
<head>
  <title>Marker Leaflet Map</title>
  <meta charset="utf-8" />
  <link
    rel="stylesheet"
    href="http://cdn.leafletjs.com/leaflet-0.7/leaflet.css"
  />
</head>
<body>
  <div id="map" style="width: 600px; height: 400px"></div>

  <script
    src="http://cdn.leafletjs.com/leaflet-0.7/leaflet.js">
  </script>

  <script>
    var map = L.map('map').setView([-41.2858, 174.78682], 14);
    mapLink =
      '<a href="http://openstreetmap.org">OpenStreetMap</a>';
    L.tileLayer(
      'http://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
        attribution: 'Map data © ' + mapLink,
        maxZoom: 18,
      }).addTo(map);
    var marker = L.marker([-41.29042, 174.78219])
      .addTo(map);
  </script>
</body>
</html>
```

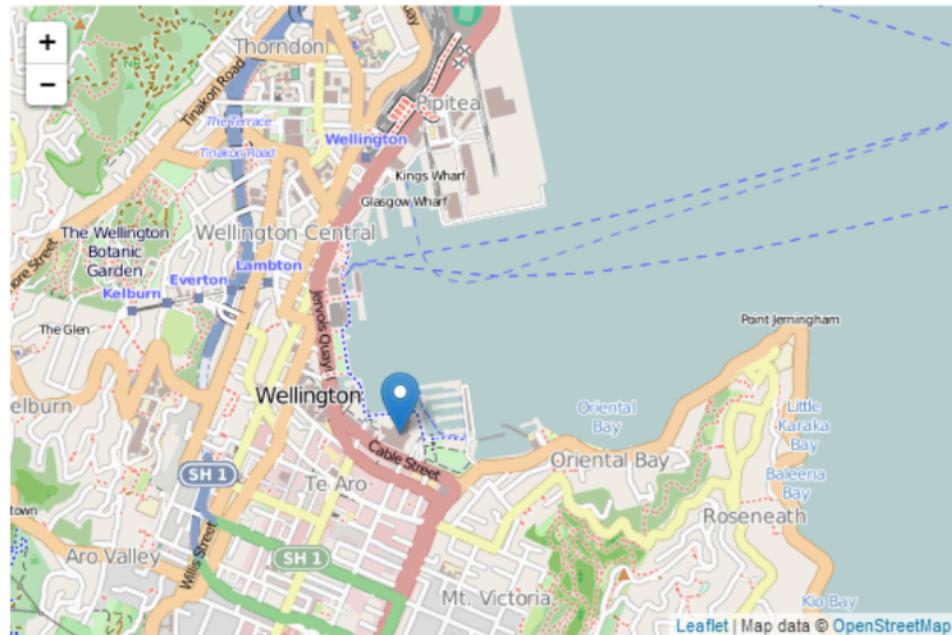
4

1. Define the selector (i.e. div id="map") for inserting the map into the DOM.
2. Initialize view and tile layer.
3. Source of the map tiles (e.g. Open Street Map).
4. Add a marker to the map view.

map01.html

[https://leanpub.com/leaflet-tips-and-tricks/  
read#leanpub-auto-adding-a-marker-to-our-map](https://leanpub.com/leaflet-tips-and-tricks/read#leanpub-auto-adding-a-marker-to-our-map)

```
var map = L.map('map').setView([-41.2858, 174.78682], 14);
mapLink =
  '<a href="http://openstreetmap.org">OpenStreetMap</a>';
L.tileLayer(
  'http://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
    attribution: '&copy; ' + mapLink + ' Contributors',
    maxZoom: 18,
  }).addTo(map);
```



Map with marker

```
var marker = L.marker([-41.29042, 174.78219])
  .addTo(map);
```

Change the latitude and longitude codes of the map to your home town and place a marker with latitude and longitude codes of a notable landmark nearby.

# Finding the **Latitude** and **Longitude** Coordinates

Google

SafeSearch on

All Maps Images News Videos More Settings Tools

About 7,200,000,000 results (0.58 seconds)

**Google Maps** Click the link to start the web app.

<https://maps.google.com>

Find a place. Your location. Trails. Dedicated lanes. Bicycle-friendly roads. Dirt/unpaved trails. Live traffic. Fast. Slow. 2000 km. Set depart & arrive time.

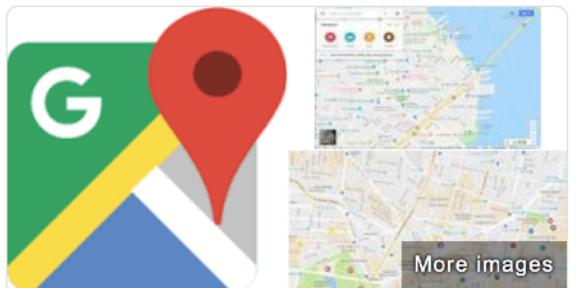
**Maps**  
Your location. Trails. Dedicated lanes. Bicycle-friendly roads ...

**Map**  
Find local businesses, view maps and get driving directions in ...

**My Maps**  
Discover the world with Google Maps. Experience Street View ...  
[More results from google.com »](#)

**Google Maps**  
<https://www.google.com.hk/maps>

Find local businesses, view maps and get driving directions in **Google Maps**.

 [More images](#)

**Google Maps**

Website

Google Maps is a web mapping service developed by Google. It offers satellite imagery, aerial photography, street maps, 360° panoramic views of streets, real-time traffic conditions, and route planning for traveling by foot, car, bicycle and air, or public transportation.

[Wikipedia](#)

**Date launched:** February 8, 2005

**Owner:** Google

Enter the address or name of your destination.

SearchX

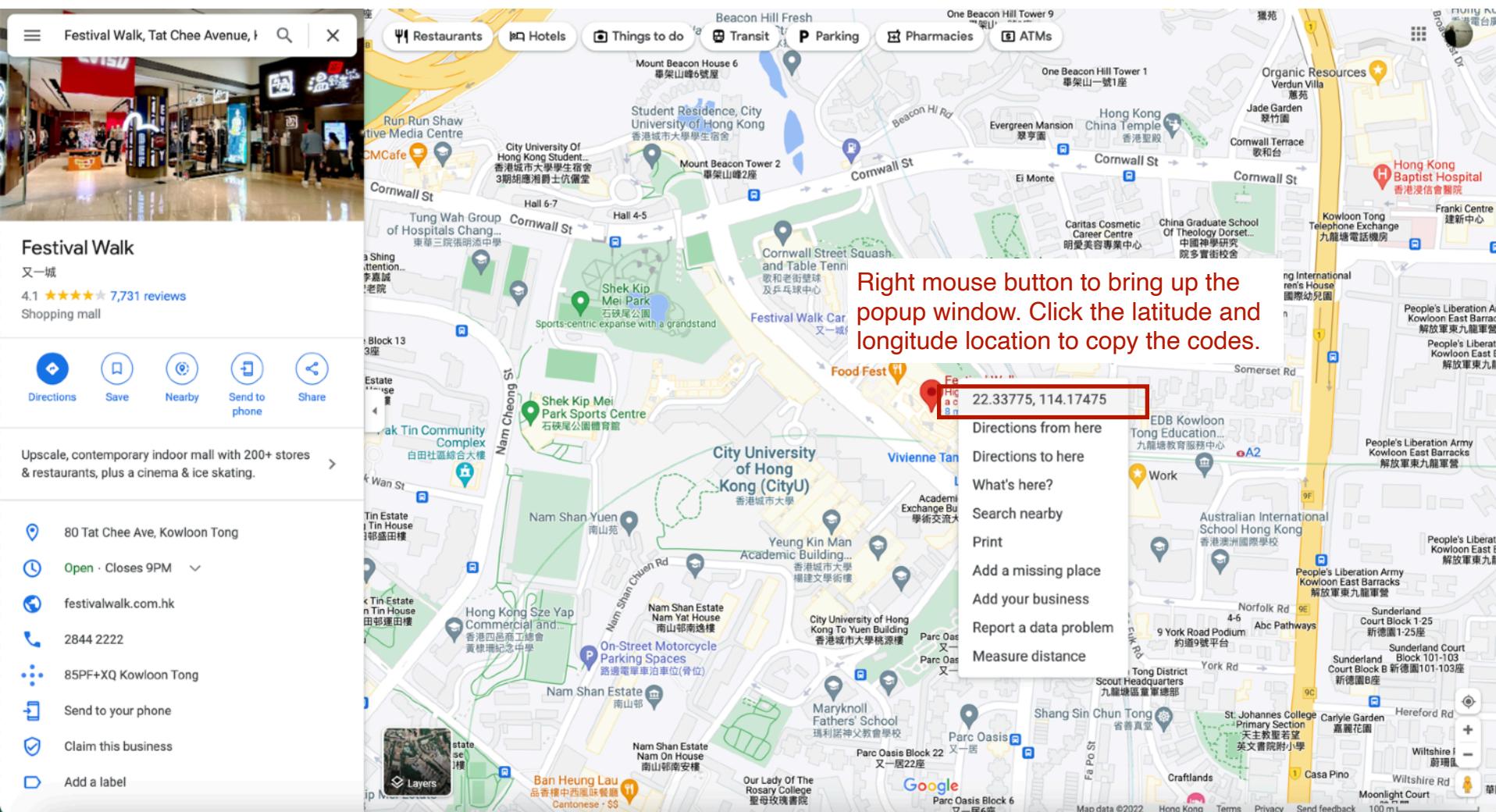


**Festival Walk**  
又一城  
4.1 ★★★★☆ 7,731 reviews  
Shopping mall

[Directions](#) [Save](#) [Nearby](#) [Send to phone](#) [Share](#)

Upscale, contemporary indoor mall with 200+ stores & restaurants, plus a cinema & ice skating.

80 Tat Chee Ave, Kowloon Tong  
Open · Closes 9PM  
[festivalwalk.com.hk](http://festivalwalk.com.hk)  
2844 2222  
85PF+XQ Kowloon Tong  
Send to your phone  
Claim this business  
Add a label



```

var map = L.map('map').setView([-41.2858, 174.78682], 14);
mapLink =
  '<a href="http://openstreetmap.org">OpenStreetMap</a>';
L.tileLayer(
  'http://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
    attribution: '&copy; ' + mapLink + ' Contributors',
    maxZoom: 18,
  }).addTo(map);

```



```

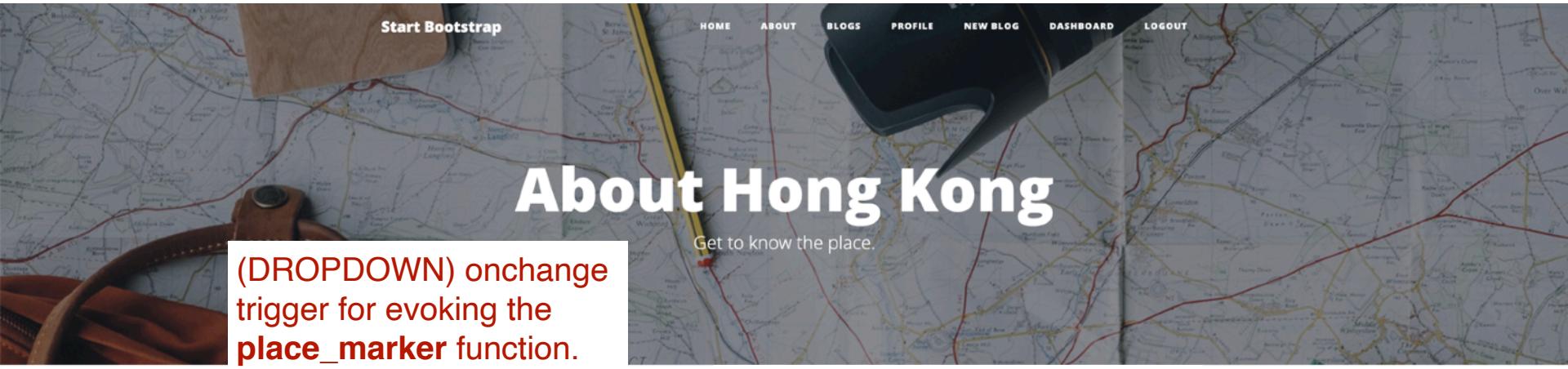
var marker = L.marker([-41.29042, 174.78219])
  .addTo(map);

```

Change the latitude and longitude codes of the map to your home town and place a marker with latitude and longitude codes of a notable landmark nearby.

map01.html

# **Integrating data visualisation features into a Flask project.**



(DROPDOWN) onchange trigger for evoking the **place\_marker** function.

▼  
**DESTINATIONS** **RESET**

(RESET) onclick trigger for evoking the **clear\_map** function.



```

1  {% extends "map_base.html" %} 
2  {% block header %} 
3      <!-- Page Header-->
4      <header class="masthead" style="background-image: url({{ url_for('static', filename='assets/img/about-bg.jpg') }})">
5          <div class="overlay"></div>
6          <div class="container">
7              <div class="row">
8                  <div class="col-lg-8 col-md-10 mx-auto">
9                      <div class="page-heading">
10                         <h1>About Hong Kong</h1>
11                         <span class="subheading">Get to know the place.</span>
12                     </div>
13                 </div>
14             </div>
15         </header>
16     {% endblock %}
17     {% block content %} 
18         <!-- Main Content-->
19         <div class="content">

```

(DROPDOWN) onchange trigger for evoking the place\_marker function.

```

26     <option value="" selected disabled>Please select</option>
27     {% for row in entries %}
28         <option value= "{{ loop.index }}">{{ row["title"] }}</option>
29     {% endfor %}
30     </select>
31     <button type="button" class="btn btn-outline-primary btn-sm" id="roll_up">Destinations</button>
32     <button type="button" class="btn btn-outline-primary btn-sm" id="roll_up" onclick="clear_map();">
33

```

(DESTINATIONS) onclick trigger for evoking the show\_districts function.

```

41             </div>
42         </div>
43     </div>
44     <hr />
45     {% endblock %}

```

Option list for drop down selection of destinations using entries data passed from the /about route.

Selector (i.e. id="map") for inserting the map into the DOM.

(RESET) onclick trigger for evoking the clear\_map function.

about.html

```
@app.route("/about")
def about():
    sqlalchemyObj = db.engine.execute('select title,lat,lng,url,img_url,body from blogs ORDER BY title')
    venues = []
    for i in sqlalchemyObj:
        venues.append(i)
    # print(venues)
    dataset = []
    dict={}
    for i in venues:
        dict['title'] = i[0]
        dict['lat'] = i[1]
        dict['lng'] = i[2]
        dict['url'] = i[3]
        dict['img_url'] = i[4]
        dict['body'] = i[5]
        #print(i)
        #print(dict)
        dataset.append(dict.copy())
    print(dataset)

    #markers.append(fld.copy())
    return render_template('about.html' entries = dataset)
```

Select all the blog entries with, img\_url, url, latitude and longitude codes for mapping and marker creation.

Create a list of dictionaries for housing the dataset to be passed to the template as entries.

```

3   <head>
4     <meta charset="utf-8" />
5     <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
6     <meta name="author" content="" />
7     <title>{& if title %}{&title}&{& else %}Suentze blogs{& endif %}</title>
8     {& if meta_desc %}<meta name="DESCRIPTION" content="{{meta_desc}}"/>{& endif %}
9     {& if meta_keywords %}<meta name="KEYWORDS" content="{{meta_keywords}}"/>{& endif %}
10    <!--link rel="icon" type="image/x-icon" href="{{ url_for('static', filename='assets/favicon.ico') }}" /-->
11    <!-- Font Awesome icons (free version)-->
12    <script src="https://use.fontawesome.com/releases/v5.15.3/js/all.js" crossorigin="anonymous"></script>
13    <!-- Google fonts-->
14    <link href="https://fonts.googleapis.com/css?family=Lora:400,700,400italic,700italic" rel="stylesheet" type="text/css" />
15    <link href="https://fonts.googleapis.com/css?family=Open+Sans:300italic,400italic,600italic,700italic,800italic,400,300,600,700,800" rel="stylesheet" type="text/css" />
16    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/leaflet/1.3.4/leaflet.css">
17    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.11.0/umd/popper.min.js"></script>
18    <script src="https://cdn.jsdelivr.net/npm/jquery@3.5.1/dist/jquery.slim.min.js"></script>
19    <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/js/bootstrap.bundle.min.js"></script>
20    <script src="https://cdnjs.cloudflare.com/ajax/libs/leaflet/1.3.4/leaflet.js"></script>
21    <script src="https://api.tiles.mapbox.com/mapbox.js/plugins/leaflet-markercluster/v0.4.0/leaflet.markercluster.js">
22    </script>
23    <script type="text/javascript" src="{{url_for('static', filename='js/TileLayer.Grayscale.js')}}"></script>
24    <!-- Core theme CSS (includes Bootstrap)-->
25    <link rel="stylesheet" href="{{url_for('static', filename='css/style.css')}}" />
26    <link rel="stylesheet" href="{{url_for('static', filename='css/custom.css')}}" />
27  </head>

```

CSS and JS files required for map support with some external and some internal.

map\_base.html

```

103     <script>
104         var mapboxTiles = L.tileLayer.grayscale('https://s.tile.openstreetmap.org/{z}/{x}/{y}.png', {attribution: 'Map
data &copy; <a href="https://openstreetmap.org">OpenStreetMap</a>', maxZoom: 18,});
105
106         var map = L.map('map',{ fadeAnimation: false })
107             .addLayer(mapboxTiles)
108             .setView([22.287111, 114.191667], 13);
109
110         layerGroup = L.layerGroup().addTo(map);
111
112         var items = {};
113         var data = [];
114
115         {% for row in entries %}
116             items = {};
117             items["title"] = "{{ row['title'] }}";
118             items["url"] = "{{ row['url'] }}";
119             items["img_url"] = "{{ row['img_url'] }}";
120             items["lat"] = {{ row['lat'] }};
121             items["lng"] = {{ row['lng'] }};
122             data.push(items);
123         {% endfor %}
124
125         function show_districts(){
126             for (var i in data) {
127                 var latlng = L.latLng({ lat: data[i].lat, lng: data[i].lng });
128                 L.marker( latlng )
129                     .bindPopup( '<a href="' + data[i].url + '" target="_blank">' + '<img src=' + data[i].img_url+'" width =
'80px"><br>' + data[i].title + '</a>' )
130                         .addTo(layerGroup);
131             }
132         }
133     </script>

```

Loop through entries to create  
data[] array for holding the markers.

Loop through data to place all  
destination markers onto the map  
tiles.

map\_base.html

```

134     function clear_map() {
135         $('#map').show();
136         layerGroup.clearLayers();
137         var mapboxTiles = L.tileLayer.grayscale('https://s.tile.openstreetmap.org/{z}/{x}/{y}.png', {attribution: 'Map
data &copy; <a href="https://openstreetmap.org">OpenStreetMap</a>', maxZoom: 18,});
138
139         var map = L.map('map', { fadeAnimation: false })
140             .addLayer(mapboxTiles)
141             .setView([22.287111, 114.191667], 13);
142
143         layerGroup = L.layerGroup().addTo(map);
144
145         var items = [];
146         var data = [];
147         {% for row in entries %}
148             items = {};
149             items["title"] = "{{ row['title'] }}";
150             items["url"] = "{{ row['url'] }}";
151             items["img_url"] = "{{ row['img_url'] }}";
152             items["lat"] = {{ row['lat'] }};
153             items["lng"] = {{ row['lng'] }};
154             data.push(items);
155         {% endfor %}
156     }
157
158     function place_marker(place) {
159         let filter = document.getElementById("filter");
160         place = filter.selectedIndex-1;
161         var latlng = L.latLng({ lat: data[place].lat, lng: data[place].lng });
162         L.marker( latlng )
163             .bindPopup( '<a href=' + data[place].url + '" target=_blank>' + '<img src=' + data[place].img_url+'" width
= "80px"><br>' +data[place].title + '</a>' )
164             .addTo(layerGroup);
165             /*
166             var marker = L.marker( dat
167                 .bindPopup( '<a href='
168                     .addTo(map);
169             */
170     }
171
172     function clear_markers () {
173         // map.removeLayer(marker);
174         layerGroup.clearLayers();
175     }
176
177 </script>

```

Place specific marker on the map tiles based on destination chosen from drop-down list.

map\_base.html

当您选中了心仪的展览，可以在右侧的列表中查看对应展馆中正在展出的其他展览

点击地图中的图片，可查看对应展馆的官网



Pop-up link  
control problem.

1 hidden field					Filter	Grouped by 1 field	Sort	...	Search
	Name	展览	场馆介绍	Pic					
所在区域									
	徐汇区	Count 4							
1	油罐艺术中心	高伟刚《回头》，当代艺术群展《Mo... 上海油罐艺术中心是...							
2	杜梦堂	动物狂欢节	杜梦堂 (Galerie DUM...)						
3	龙美术馆西岸馆	叶茂中画展, 松山智一个展“自然可解”...	龙美术馆是由中国收...						
4	余耀德美术馆	安德烈·布特兹：光，色彩与希望, 摩...	余德耀美术馆坐落于“...						
所在区域									
	浦东新区	Count 2							
5	震旦博物馆	【民·潮】特展   张信哲和他的收藏, ...	震旦博物馆是上海陆...						
6	艺仓美术馆	WAVELENGTH:新黄金时代	艺仓美术馆 (Modern ...)						
所在区域									
	虹口区	Count 1							
15 records									
Airtable									
<a href="#">Download CSV</a>									
<a href="#">View larger version</a>									

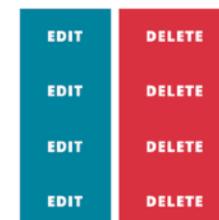


# Dashboard

**SEARCH**

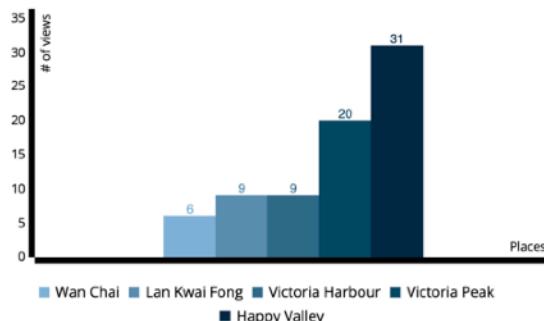
ID	Title
1	North Point
2	Mong Kok
3	Happy Valley
4	Victoria Peak

Author	Date
admin	05-16-2021

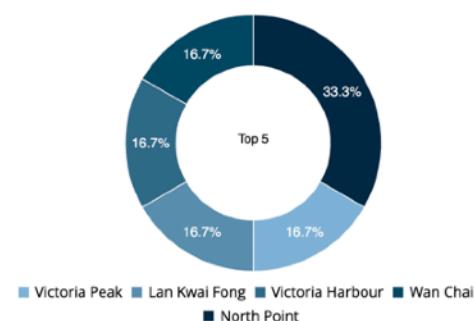


Page: 1 2

**Top 5 Blog Posts by Number of Views**



**Top 5 Blog Posts by Number of Comments**



```

@app.route("/dashboard")
@login_required
def dashboard():
    page_num = 1
    if session['admin'] > 0:
        blog_list = Blog.query.paginate(per_page=4, page=page_num, error_out=True)
        top5 = []
        result = db.engine.execute('select title, views from blogs order by views limit 5')
        for i in result:
            top5.append(i)
        dataset = []
        top5_rec={}
        for i in top5:
            top5_rec[ 'title' ] = i[0]
            top5_rec[ 'views' ] = i[1]
        dataset.append(top5_rec.copy())
    top5=[]
    result = db.engine.execute('select title, comments from blogs order by comments limit 5')
    for i in result:
        top5.append(i)
    dataset2 = []
    total = 0
    top5_rec={}
    for i in top5:
        top5_rec[ 'title' ] = i[0]
        top5_rec[ 'comments' ] = i[1]
        total = total + i[1]
    dataset2.append(top5_rec.copy())

```

Select top 5 entries with highest # of views and pass to template as entries with dataset assigned.

Select top 5 entries with highest # of comments and pass to template as entries2 with dataset2 assigned. The total variable is used to compute the total value for calculating the donut chart proportion.

```

top5=[]
result = db.engine.execute('select title, comments from blogs order by comments limit 5')
for i in result:
    top5.append(i)
dataset2 = []
total = 0
top5_rec={}
for i in top5:
    top5_rec['title'] = i[0]
    top5_rec['comments'] = i[1]
    total = total + i[1]
    dataset2.append(top5_rec.copy())
#print("dataset2=",dataset2)

else:
    blog_list = Blog.query.filter_by(author=session['username']).paginate(per_page=4, page=page_num, error_out=True)
    top5 = []
    result = db.engine.execute('select title, views from blogs order by views limit 5')
    for i in result:
        top5.append(i)
    dataset = []
    top5_rec={}
    for i in top5:
        top5_rec['title'] = i[0]
        top5_rec['views'] = i[1]
        dataset.append(top5_rec.copy())

    top5=[]
    result = db.engine.execute('select title, comments from blogs order by comments limit 5')
    for i in result:
        top5.append(i)
    dataset2 = []
    total = 0
    top5_rec={}
    for i in top5:
        top5_rec['title'] = i[0]
        top5_rec['comments'] = i[1]
        total = total + i[1]
        dataset2.append(top5_rec.copy())
    print("dataset2=",dataset2)

return render_template('dashboard.html', blogs=blog_list, entries=dataset, entries2=dataset2, total=total)

```

top 5 entries with highest # of views (i.e. entries) and highest # of comments (i.e. entries2) and total value to be passed to the dashboard.html template.

```

3 <head>
4   <meta charset="utf-8" />
5   <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
6   <meta name="author" content="" />
7   <title>{% if title %}{{title}}{% else %}Suentze blogs{% endif %}</title>
8   {% if meta_desc %}<meta name="DESCRIPTION" content="{{meta_desc}}"/>{% endif %}
9   {% if meta_keywords %}<meta name="KEYWORDS" content="{{meta_keywords}}"/>{% endif %}
10  <!--link rel="icon" type="image/x-icon" href="{{ url_for('static', filename='assets/favicon.ico') }}" /-->
11  <!-- Font Awesome icons (free version)-->
12  <script src="https://use.fontawesome.com/releases/v5.15.3/js/all.js" crossorigin="anonymous"></script>
13  <!-- Google fonts-->
14  <link href="https://fonts.googleapis.com/css?family=Lora:400,700,400italic,700italic" rel="stylesheet" type="text/css" />
15  <link href="https://fonts.googleapis.com/css?
family=Open+Sans:300italic,400italic,600italic,700italic,800italic,400,300,600,700,800" rel="stylesheet" type="text/css" />
16  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.11.0/umd/popper.min.js"></script>
17  <script src="https://cdn.jsdelivr.net/npm/jquery@3.5.1/dist/jquery.slim.min.js"></script>
18  <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/js/bootstrap.bundle.min.js"></script>
19  <script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/d3/5.12.0/d3.min.js"></script>
20  <script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/c3/0.7.11/c3.min.js"></script>
21  <script type="text/javascript" src="https://suentze2021.pythonanywhere.com/static/js/TileLayer.Grayscale.js"></script>
22  <!-- Core theme CSS (includes Bootstrap)-->
23  <link rel="stylesheet" href="{{url_for('static', filename='css/style.css')}}">
24  <link rel="stylesheet" href="{{url_for('static', filename='css/custom.css')}}">
25 </head>

```

CSS and JS files required for chart support with some external and some internal.

chart\_base.html

```

63 <div class="row" style="margin-top:5%;">
64   <div class="col-sm-6">
65     <h4>Top 5 Blog Posts by Number of Views</h4>
66     <div id="chart"></div>
67   </div>
68   <div class="col-sm-6">
69     <h4>Top 5 Blog Posts by Number of Comments</h4>
70     <div id="chart2"></div>
71   </div>
72 </div><!--end row-->
73 </div>
74 <script>
75   var chart = c3.generate({
76     data: {
77       columns:
78         [% for i in entries %]
79           ['{{i.title}}', {{i.views}}],
80         {% endfor %}
81       ],
82       type : 'bar',
83       labels:true,
84       colors: {
85         [% for i in entries %]
86           {% if loop.index == 1 %} '{{i.title}}': '#7FB3D5', {% endif %}
87           {% if loop.index == 2 %} '{{i.title}}': '#598EAE', {% endif %}
88           {% if loop.index == 3 %} '{{i.title}}': '#326A89', {% endif %}
89           {% if loop.index == 4 %} '{{i.title}}': '#004865', {% endif %}
90           {% if loop.index == 5 %} '{{i.title}}': '#002943' {% endif %}
91         {% endfor %}
92       },
93       bar: {
94         title: "Top 5 in views:",
95         width: {
96           ratio: 0.5
97         }
98       },
99       tooltip: {
100         show: false,
101       },
102       axis: {
103         x: {
104           label: 'Places',
105           type: 'category',
106           categories:['','','','','','']
107         },
108         y: {label: '# of views'}
109       }
110     });
111   // end c3.generate
112 });

```

Selectors (i.e. id="chart" and id="chart2") for inserting the bar chart and donut chart into the DOM with a Bootstrap row holding two Bootstrap columns.

Loop through entries to place top 5 # of views.

Loop through entries and test which one is the current entry for matching the appropriate color to the bar.

Trick to remove the “0” label from the x axis.

dashboard.html

```

114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137

var chart = c3.generate({
  data: {
    columns: [
      {for i in entries2}
        ['{{i.title}}', {{i.comments}}/{{total}}],
      {endfor}
    ],
    type : 'donut',
    colors: {
      for i in entries2
        if loop.index == 1 '{{i.title}}': '#7FB3D5', endif
        if loop.index == 2 '{{i.title}}': '#598EAE', endif
        if loop.index == 3 '{{i.title}}': '#326A89', endif
        if loop.index == 4 '{{i.title}}': '#004865', endif
        if loop.index == 5 '{{i.title}}': '#002943' endif
      endfor
    },
    donut: {
      title: "Top 5",
    },
    bindto: '#chart2'
  });
}

</script>

```

Loop through entries to place top 5 # of comments over the total value to correctly depict the donut proportion.

Loop through entries and test which one is the current entry for matching the appropriate color to the bar.

dashboard.html

**Integration of websites through  
Iframe and CSS touch-up.**

局

[返回主页](#)



● ● ●

## 戏剧信息

类型: 话剧

时间: 11月27日-28日 20 : 00  
11月28日-29日 14 : 30

地点: 福田区福田文化馆

票价: 100-280元

[购票](#)

剧情简介:

《局》这部戏从我们身边的“局”出发，从最初的“攒局”到最后的“局散”，随着“局”的推进，人与人之间的关系愈加微妙，究竟是旁观者，局外人，还是不知不觉已然成为故事主角？时间在流逝，人们想从局中脱身，可当时机一而再再而三错过时，仿佛从此陷入一种破罐破摔的氛围里，甘愿被困在“局”中。

演职员:

廖书艺 / 萧竞 / 苗春雨 / Luka / 等

[Tutorials ▾](#)[References ▾](#)[Exercises ▾](#)[Videos](#)[Get Certified](#)[Free Website](#)[HTML](#)[CSS](#)[JAVASCRIPT](#)[SQL](#)[PYTHON](#)[JAVA](#)[PHP](#)[BOOTSTRAP](#)[HOW TO](#)[HTML Styles](#)[HTML Formatting](#)[HTML Quotations](#)[HTML Comments](#)[HTML Colors](#)[HTML CSS](#)[HTML Links](#)[HTML Images](#)[HTML Favicon](#)[HTML Tables](#)[HTML Lists](#)[HTML Block & Inline](#)[HTML Classes](#)[HTML Id](#)[HTML Iframes](#)[HTML JavaScript](#)[HTML File Paths](#)[HTML Head](#)[HTML Layout](#)

黑色星期五，享3折優惠！  
學生享Creative Cloud全套應用程式折扣，  
優惠至11月25日。

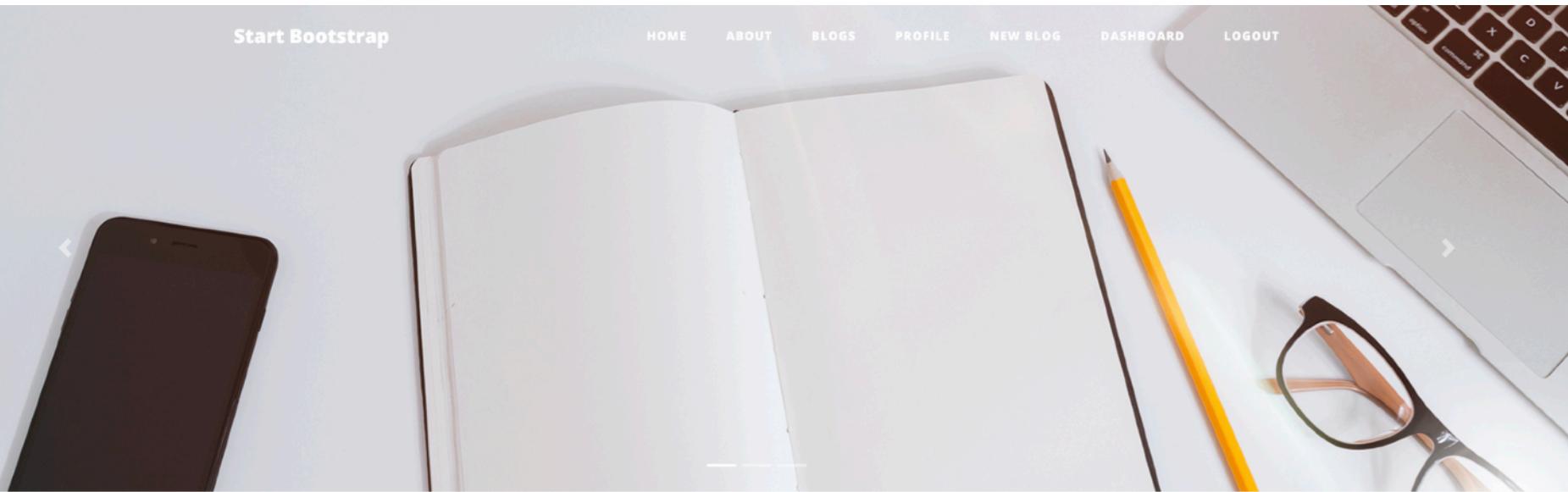
[立即購買](#)

# HTML Iframes

[◀ Previous](#)[Next ▶](#)

An HTML iframe is used to display a web page within a web page.

[Tutorials ▾](#)[References ▾](#)[Log in](#)[HTML](#)[CSS](#)[JAVASCRIPT](#)[SQL](#)



## Introducing Hong Kong

civitatis HongKong

English ▾

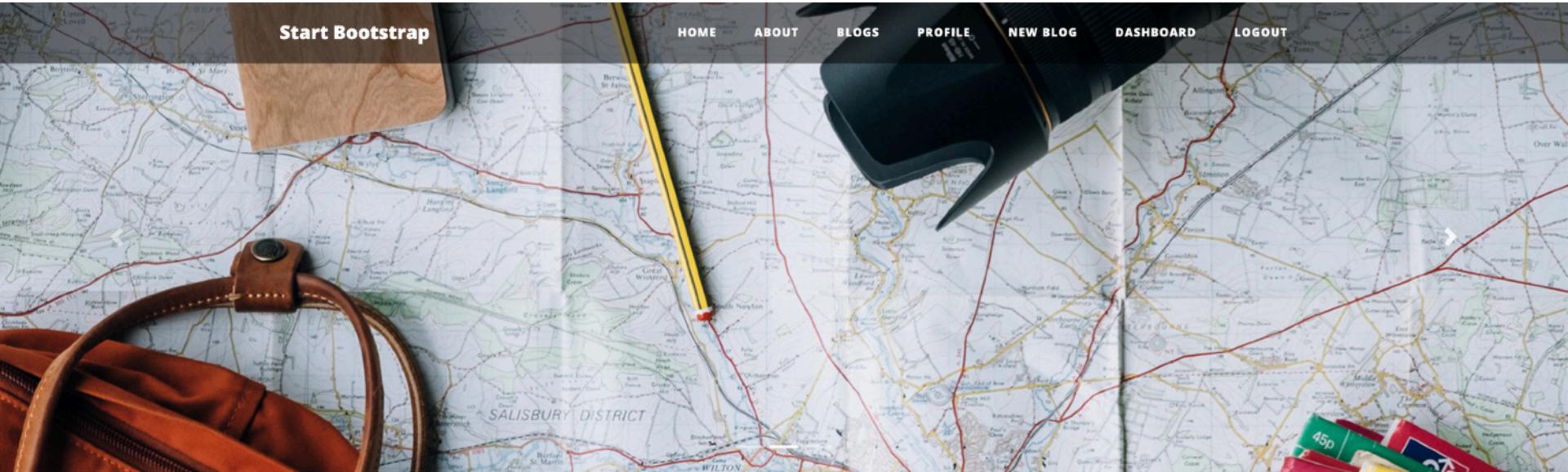
£ ▾



Travel Guide

Day trips and guided tours · Hotel deals · Flights





## Introducing Hong Kong

civitatis HongKong

English ▾

£ ▾



Travel Guide

Day trips and guided tours · Hotel deals · Flights



# Clean Blog

A Blog Theme by Start Bootstrap

Search

SEARCH

**MAKING REMOTE WORKS  
COLLABORATIVE AND  
ACCOUNTABLE**

30 2

By admin on 05-16-2021

**LESSON FROM THE  
TIKTOK DEAL:  
NAVIGATING THE  
PARALLEL INTERNET**

48 0

By admin on 05-16-2021

**COVID-19: THE GREAT  
ACCELERATOR OF WORK  
AND LEARNING**

27 2

By admin on 05-16-2021

# Clean Blog

A Blog Theme by Start Bootstrap

SEARCH

**North Point**

37  2



**Mong Kok**

60  2



**Happy Valley**

31  3

## **Major JavaScript libraries and frameworks used in the class presentation.**

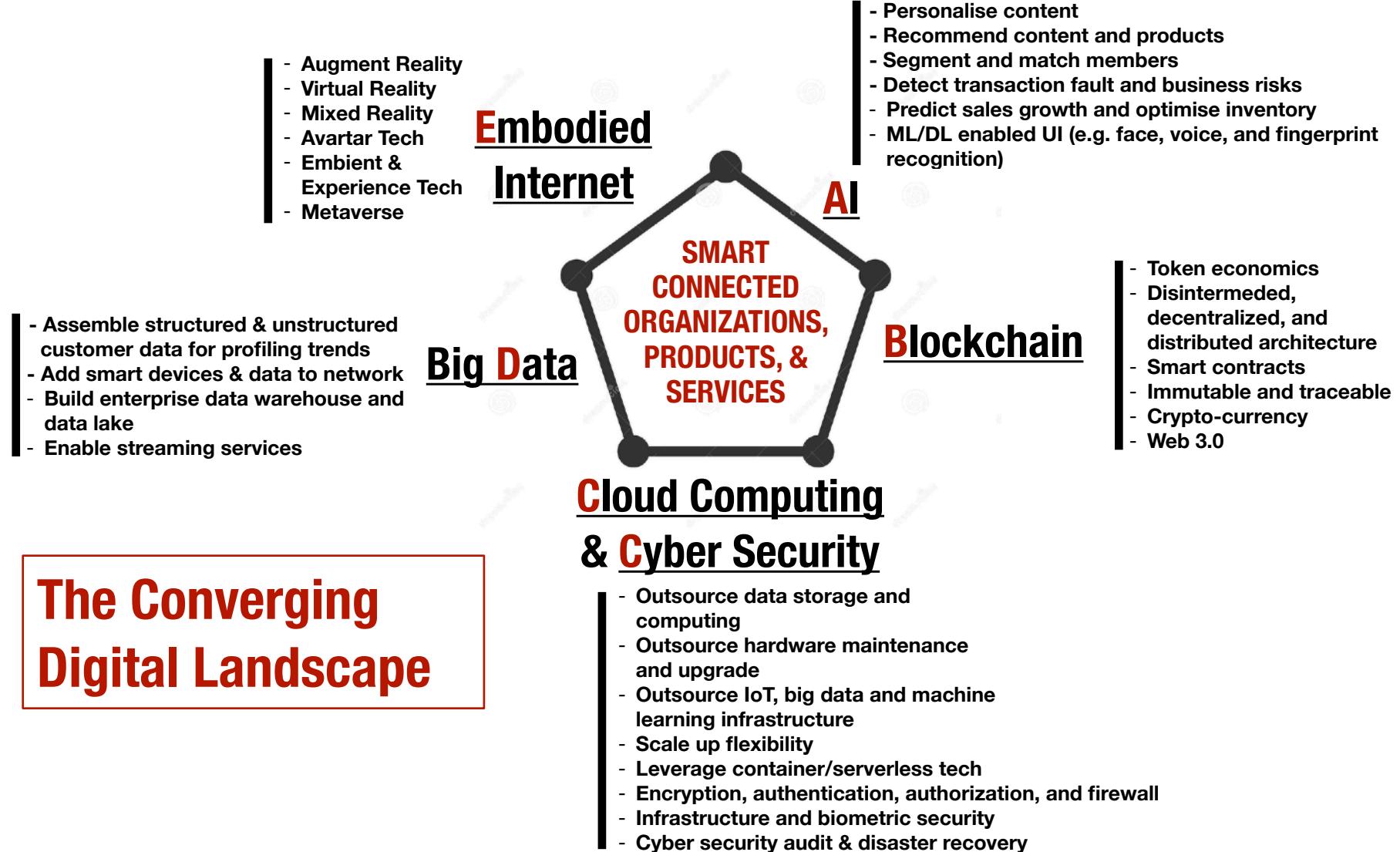
- 1. D3 (<https://d3js.org/>)**
- 2. C3 (<https://c3js.org/>)**
- 3. Leaflet (<https://leafletjs.com/>)**
- 4. Open Street Map (<https://openstreetmap.org>)**

**UX/UI trend to watch amidst the converging digital landscape.**

# The Converging Digital Landscape

- Assemble structured & unstructured customer data for profiling trends
- Add smart devices & data to network
- Build enterprise data warehouse and data lake
- Enable streaming services

- Augment Reality
- Virtual Reality
- Mixed Reality
- Avatar Tech
- Embient & Experience Tech
- Metaverse



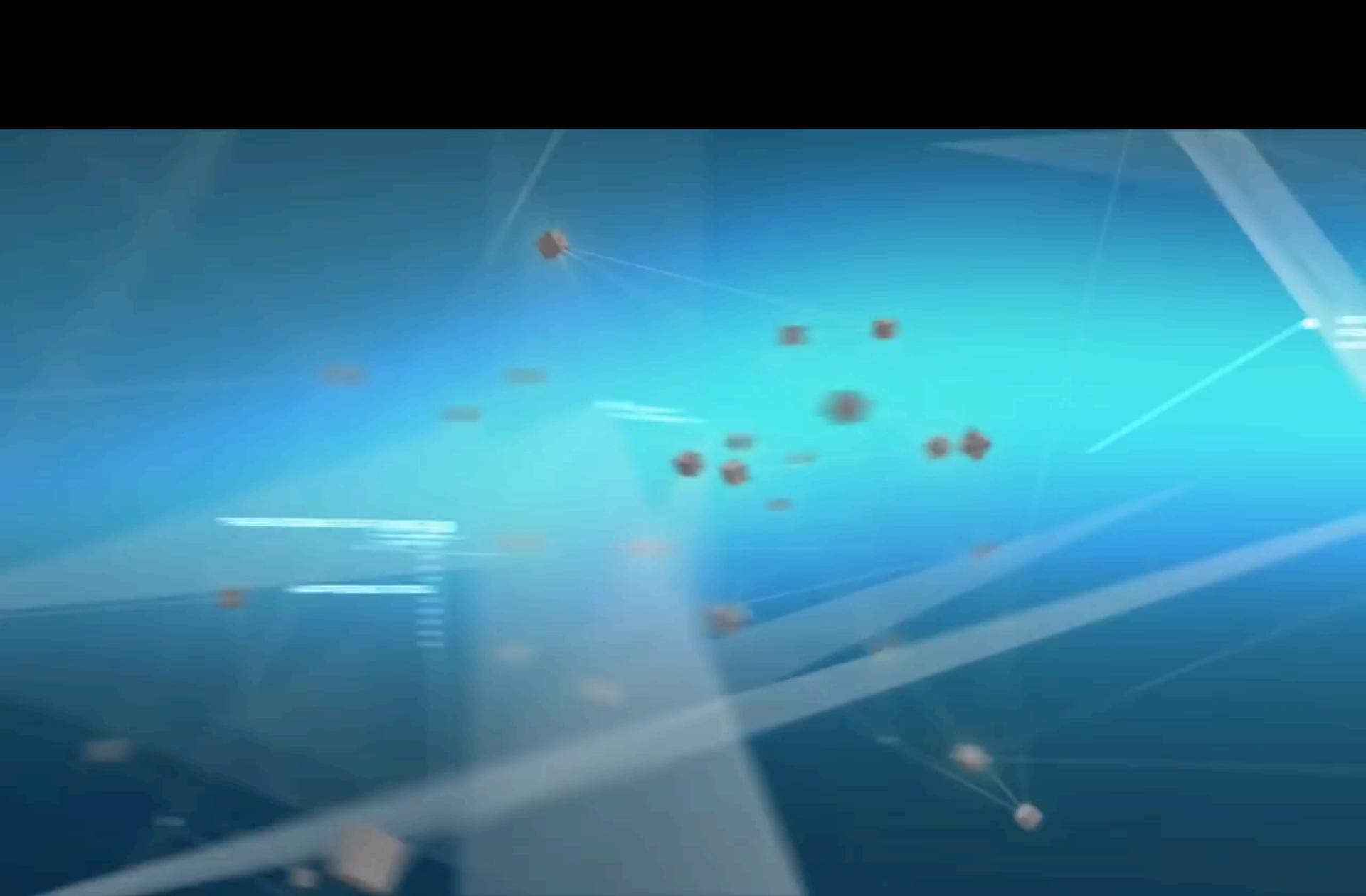
- Outsource data storage and computing
- Outsource hardware maintenance and upgrade
- Outsource IoT, big data and machine learning infrastructure
- Scale up flexibility
- Leverage container/serverless tech
- Encryption, authentication, authorization, and firewall
- Infrastructure and biometric security
- Cyber security audit & disaster recovery

- Personalise content
- Recommend content and products
- Segment and match members
- Detect transaction fault and business risks
- Predict sales growth and optimise inventory
- ML/DL enabled UI (e.g. face, voice, and fingerprint recognition)

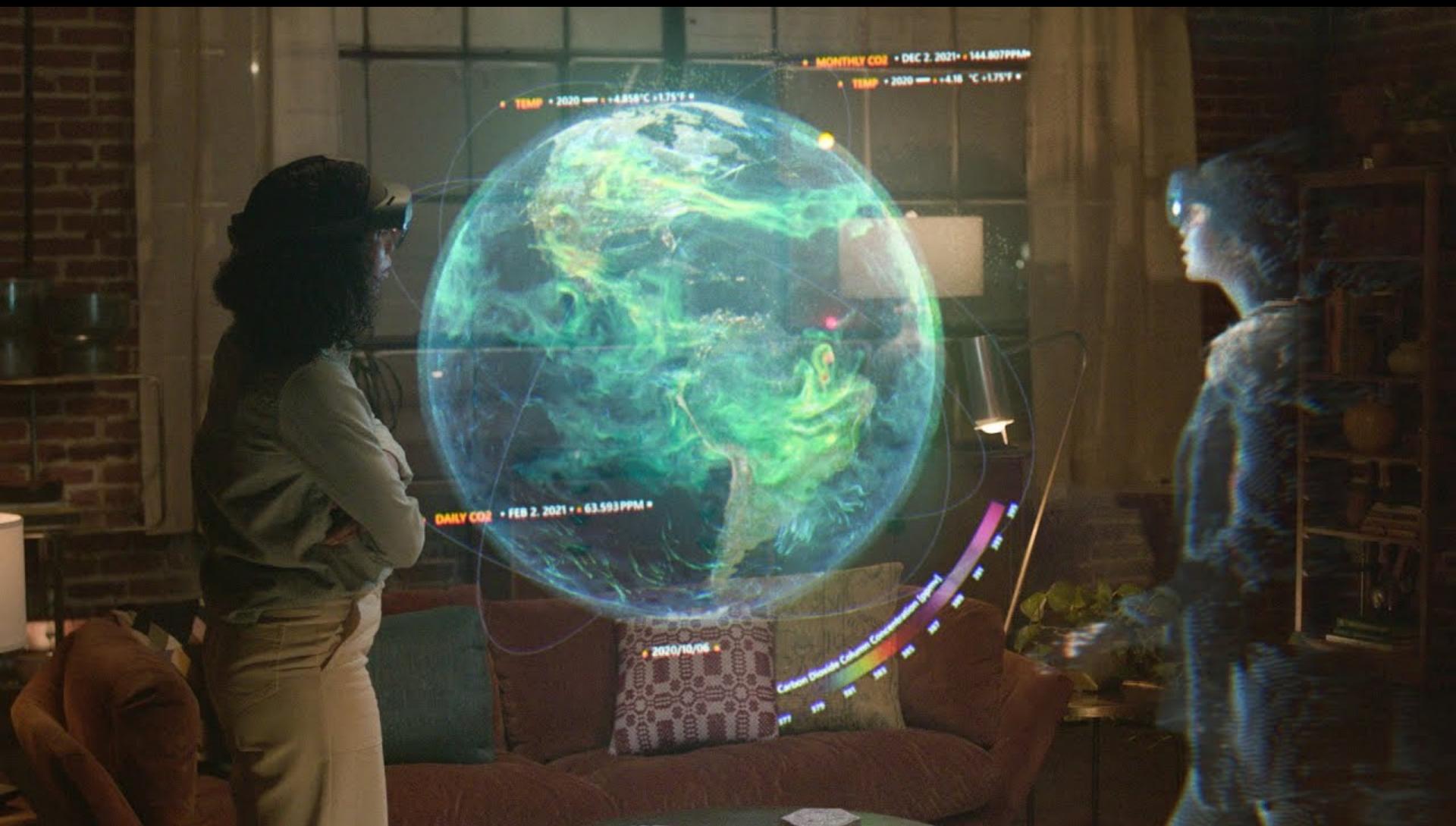
- Token economics
- Disintermediated, decentralized, and distributed architecture
- Smart contracts
- Immutable and traceable
- Crypto-currency
- Web 3.0



```
1 01 01 00 10 10 01 00 10 01 10 00 10 11 00 11 00 0  
1 01 11 11 10 11 00 11 10 10 01 10 10 11 01 10 11 10 0  
0 10 10 10 10 10 00 11 00 00 00 00 01 01 11 10 10 0  
0 11 01 11 00 00 01 00 11 11 00 11 11 01 10 00 10 00 1  
0 00 11 00 01 11 01 10 10 01 00 11 11 01 01 11 01 11 0  
1 11 10 01 01 01 00 01 01 00 01 00 00 11 00 00 01 10 0  
1 10 01 01 01 11 01 00 10 00 00 10 00 01 10 11 00 10 1  
1 00 10 00 01 00 10 11 01 00 00 10 10 10 01 10 01 10 0  
1 10 11 00 01 11 00 10 01 11 11 11 11 10 11 01 01 1  
0 00 11 00 11 10 10 00 00 11 01 11 10 11 01 10 01 01 0  
1 01 00 01 01 00 10 10 00 00 11 11 00 11 00 01 10 01 0  
1 10 10 00 10 10 10 10 11 10 01 10 10 01 11 00 01 00 1  
0 00 01 10 01 00 00 11 00 11 10 00 10 10 00 11 00 10 1  
1 01 11 01 01 10 11 00 00 11 10 01 10 00 00 01 10 01 1  
0 11 00 11 01 00 11 11 10 10 11 11 10 11 11 00 00 10 0  
1 11 01 10 11 11 01 00 01 01 00 00 01 11 00 00 00 10 0  
1 11 01 01 00 10 11 01 11 11 10 11 00 11 10 01 01 1  
0 01 01 00 10 11 11 00 11 01 00 10 10 00 00 00 11 11 1  
0 10 11 01 00 01 11 10 11 10 11 01 01 11 10 00 00 01 00 0  
0 01 11 01 10 10 01 11 00 11 00 00 01 01 10 01 00 01 10 1  
1 00 01 00 11 11 10 10 11 11 11 01 11 10 11 10 11 11 1  
1 10 11 11 10 00 10 01 01 00 11 11 00 11 01 10 10 00 0  
1 11 10 11 10 00 10 10 00 10 01 11 01 11 10 00 11 00 1  
1 10 01 00 10 01 11 00 10 00 10 01 01 01 11 10 00 01 00 1  
0 10 00 11 10 10 11 01 00 10 10 11 00 10 01 11 11 01 1  
0 00 01 11 00 01 11 00 10 00 10 00 01 01 11 01 11 00 0  
1 11 11 01 00 10 00 11 10 10 00 11 00 00 10 11 10 10 0  
1 01 11 11 11 01 00 11 01 11 00 00 11 01 10 00 00 01 1  
0 11 00 00 11 01 01 11 01 11 01 00 11 10 11 01 10 10 1  
1 10 10 11 00 00 00 10 00 10 10 00 11 00 11 10 11 10 1  
1 10 01 10 11 00 01 01 01 10 10 10 11 11 10 01 01 01 0  
1 11 10 01 01 00 11 01 11 00 10 11 00 11 01 10 00 01 00 1  
0 01 10 00 01 10 11 10 01 00 00 00 11 00 11 01 00 01 0  
1 11 10 11 10 00 00 11 00 11 00 00 11 11 01 11 01 11 0  
0 00 00 01 10 01 01 10 11 11 10 10 01 01 00 11 11 00 0  
0 11 00 00 00 11 10 01 00 00 00 10 11 11 11 10 01 01 0  
0 11 01 10 11 11 10 00 10 00 01 11 10 11 00 00 01 00 11 0  
0 11 01 00 00 01 11 10 10 10 00 10 01 11 11 00 00 01 11 1
```



Source: <https://www.youtube.com/watch?v=ibdm1zU6G3Q>



Source: <https://www.youtube.com/watch?v=Jd2GK0qDtRg>

**Dashboard**



**Digital  
Twin**



**Augment  
Reality**



**Metaverse**



# **Inspiration of COM5961**



<https://www.youtube.com/watch?v=5dFxQRcsKpw>



**THIS IS CS50.**  
CS50.TV

<https://www.youtube.com/watch?v=59pfsj4nvl8>

# CS50

From Wikipedia, the free encyclopedia

*For the sailboat, see [CS 50](#).*

**CS50 (Computer Science 50)** is an on-campus and online introductory course on computer science taught at Harvard University and Yale University. In 2016, CS50 became available to high school students as an Advanced Placement Computer Science course. The on-campus version is Harvard's largest class with 800 students, 102 staff and up to 2,200 participants in their regular hackathons.<sup>[1][2]</sup>

The course material is available online for free on the EdX platform, with a range of certificates available for a fee. As of 2022, this online version, called CS50x, teaches the languages C, Python, SQL, HTML, CSS, and JavaScript. It also teaches fundamental computer-science concepts including arrays and data structures, and the Flask web framework.<sup>[3]</sup> The 2021 iteration of the course introduced three new additional lectures on computer security, artificial Intelligence, and the ethics of technology.<sup>[4]</sup>

The lead instructor is Harvard professor David Malan.<sup>[5]</sup> Doug Lloyd and Brian Yu<sup>[6]</sup> is also a former member of CS50 and was one of the lecturers.<sup>[7]</sup> Facebook co-founder Mark Zuckerberg and former Microsoft Chief Executive Officer Steve Ballmer have given guest lectures.<sup>[8][9]</sup> The CS50 course first appeared on-campus in 1989.

Several follow-on programs exist, focusing on web programming, artificial intelligence, game development, and mobile apps.

## About this course

This is **CS50x**, Harvard University's introduction to the intellectual enterprises of computer science and the art of programming for majors and non-majors alike, with or without prior programming experience. An entry-level course taught by David J. Malan, **CS50x** teaches students how to think algorithmically and solve problems efficiently. Topics include abstraction, algorithms, data structures, encapsulation, resource management, security, software engineering, and web development. Languages include C, Python, SQL, and JavaScript plus CSS and HTML. Problem sets inspired by real-world domains of biology, cryptography, finance, forensics, and gaming. The on-campus version of **CS50x**, **CS50**, is Harvard's largest course.

Students who earn a satisfactory score on 9 problem sets (i.e., programming assignments) and a final project are eligible for a certificate. This is a self-paced course—you may take **CS50x** on your own schedule.

HarvardX requires individuals who enroll in its courses on edX to abide by the terms of the edX honor code. HarvardX will take appropriate corrective action in response to violations of the [edX Honor Code](#).

# This was CS50

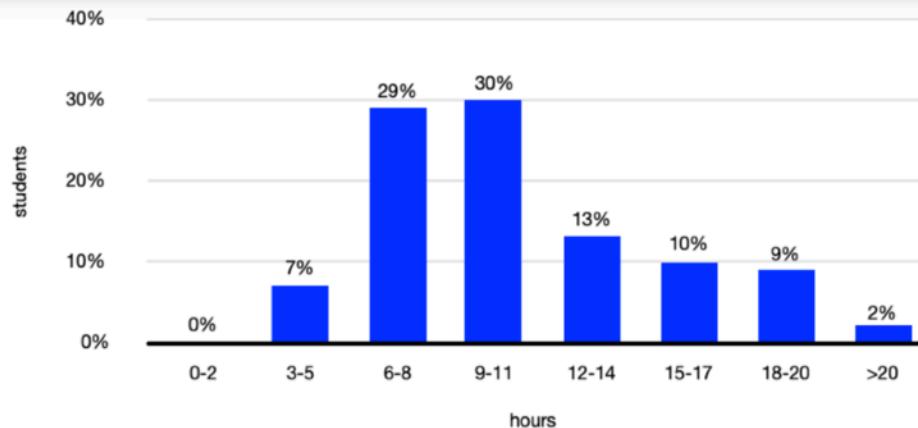
Harvard College  
Fall 2021

Search

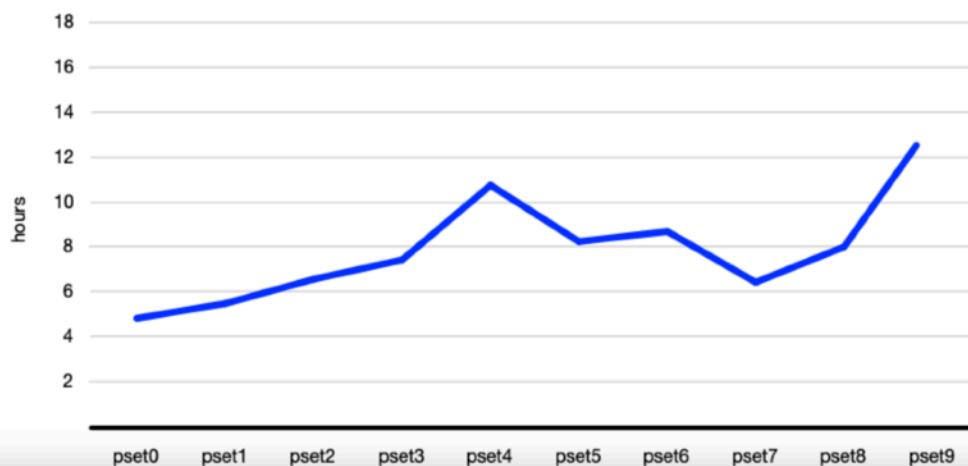
Week 0 Scratch  
Week 1 C  
Week 2 Arrays  
Week 3 Algorithms  
Week 4 Memory  
Week 5 Data Structures  
Week 6 Python  
Week 7 SQL  
Week 8 HTML, CSS, JavaScript  
Week 9 Flask  
Week 10 Emoji  
Cybersecurity

Ed Discussion for Q&A

Announcements  
FAQs  
Final Projects  
Labs  
Music



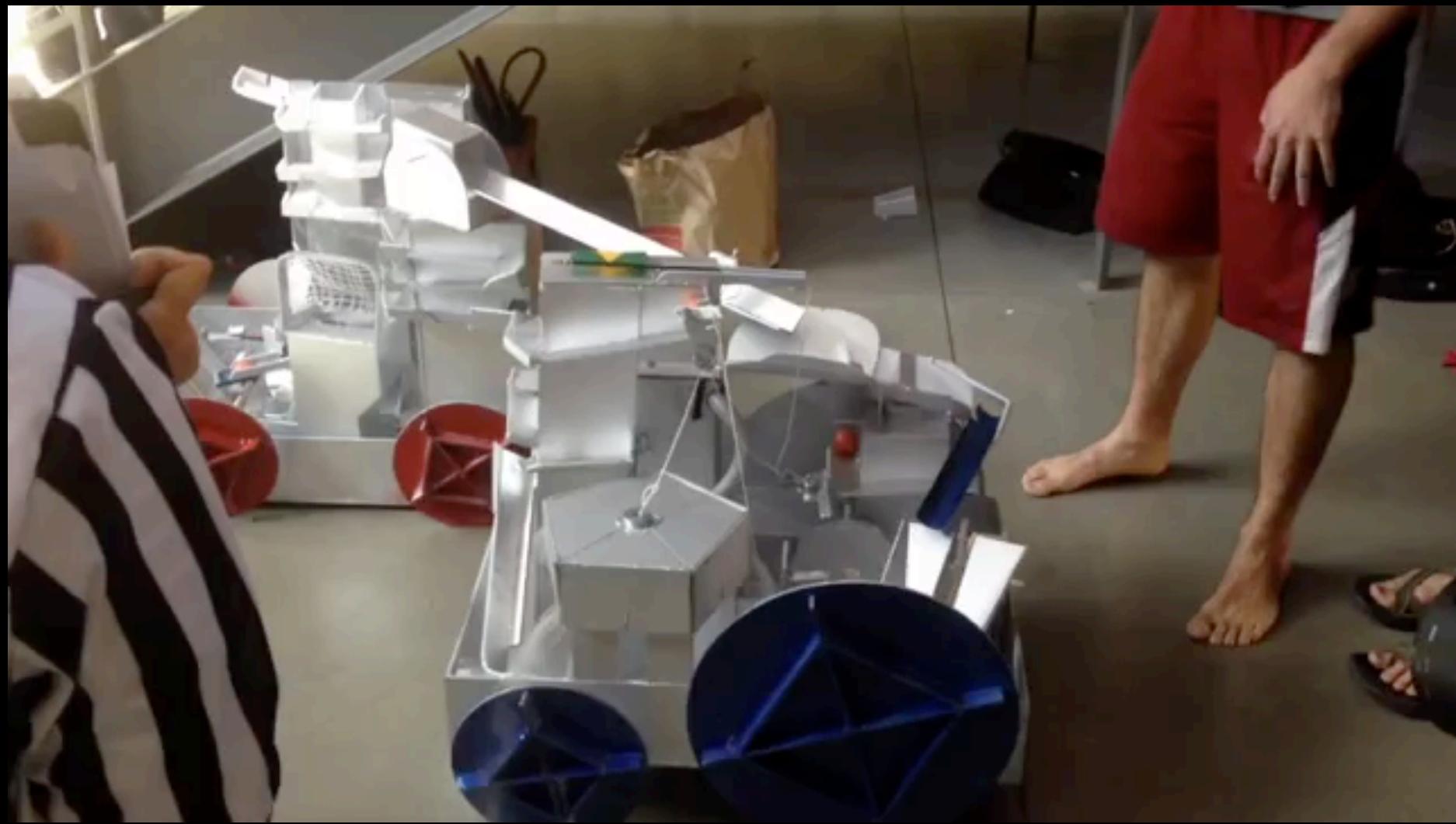
By mid-semester, most students spend 10+ hours per week on the course's problem sets, but it definitely varies by problem set, per the below, and student.



# A place for explorers & experimenters at Stanford University.

[What We Do](#)[How We Do It](#)[Our Impact](#)[The Home Team](#)[How to start a d.school](#)

<https://www.youtube.com/watch?v=XgpQXSVdm2w>



<https://www.youtube.com/watch?v=-rnGn4cPjhA>



<https://www.youtube.com/watch?v=KII1MR-qNt8>

# **Inspiration of COM5940**



# 2.009

Product Engineering  
Processes

Hello!

- **Fall 2022 Notice**

For more information about the Fall 2022 offering of 2.009, please contact Professors Ellen Roche and Josh Wiesman.

This website is for the Fall 2021 offering of 2.009 taught by Professor David Wallace.

- **Fall 2021 Website**

Team Presentations:

- Blue Team - [Eclipse](#)
- Orange Team - [Escalate](#)
- Purple Team - [Vulcan](#)
- Red Team - [Dextra](#)
- Green Team - [deltaTherapy](#)
- Silver Team - [Aisle Assist](#)
- Pink Team - [ReVise](#)

- **of interest ([archive](#))**

[Team Teasers](#)

[2.009 returns to the stage](#)

[Scheduling Jam](#)

<https://web.mit.edu/2.009/www/index.html>



Source: [https://www.youtube.com/results?search\\_query=MIT+2.009](https://www.youtube.com/results?search_query=MIT+2.009)

# Comparison Between COM5961 and COM5940

	<b>COM5961</b>	<b>COM5940</b>
<b>Competency</b>	Individual mastery	Team project management
<b>Methodology</b>	DT, Elements of UX, MVC, usability study and AB Test	Business model design, lean & agile product development, and CRISP-DM
<b>Requirement Analysis</b>	Product	Market, Product, Business
<b>Front-end/ Back-end</b>	Front-end: HTML/CSS/JS Back-end: Python Flask/SQL	Front-end: Web/Mobile Back-end: Flask/SQL and API
<b>Technical Focus</b>	Web app prototyping	Mobile, cloud and API services with use of AI and AR in business model and product innovation
<b>Application Focus</b>	Content acquisition and content management	Community formation and E-Commerce (Conversion)

# **About the final project**

## **(Due Dec 16th, 2022)**

- a. Integrate relevant works from your previous assignments into your final project (**report + prototype+usability report+A/B test + appendix — no limit on # of pages**).
- b. Separate your report into **four major sections (problem background, requirements, design and test/evaluation)** with sub-sections in each for elaborating the development of your ideas from problem definition to solution provision and evaluation.
- c. The evaluation sub-section should include usability and A/B tests results and **future recommendations (e.g. change or expand scope)** about conducting further product development, market studies, and business model innovation (i.e. subjects for next term) to turn the prototype into a sustainable product.
- d. Bonus points — 1 page self-reflection journal on what you've learned from this course.

**Thanks for joining me today!**