

<image/nation>

Tech Training Series

Exclusive for



HKU  
BUSINESS  
SCHOOL  
港大經管學院

# JavaScript Programming

(2024/2025)

## Intermediate Level

*by*

Sunny NG

<image/nation>

To download the slides

[bit.ly/in-download](https://bit.ly/in-download)

# In this workshop (3 hours)

- Basic JavaScript Syntax
- Understand JavaScript Object and JavaScript Object Notation (JSON)
- `fetch` API
- JSON Processing
- JS Libraries / Tools
  - Apache eCharts
  - Google Charts
  - Observable HQ

# Sunny Ng 吳新陸



Master of Fine Art, CityU (HK)  
Master of Science, HKU  
Bachelor of Science, UH (UK)

- Founder / Master Trainer Image Nation
- Part-time Lecturer HKU, HKUSPACE, City University of Hong Kong
- Developer AI, Web, Mobile, WeChat & IoT
- Content Creator Video producing / Live streaming
- Certified Azure AI Engineer
- AWS Solution Architect - Associate
- Alibaba Cloud Certified Professional
- AWS Academy Educator
- Email: [sunny.ng@imagenation.com.hk](mailto:sunny.ng@imagenation.com.hk)
- [linkedin.com/in/ngsunny/](https://www.linkedin.com/in/ngsunny/)
- [github.com/ngsanluk](https://github.com/ngsanluk)

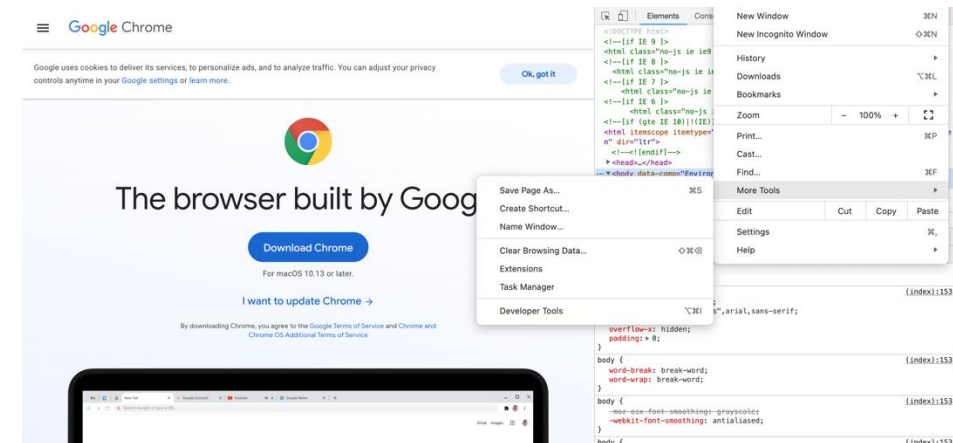
# Required Dev Tools

Since software download/installation takes time, please download and install the following software

1. Google Chrome Browser
2. Visual Studio Code
3. Node JS

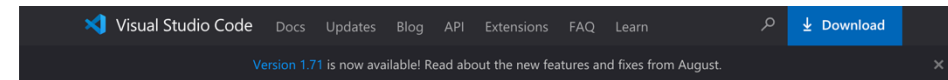
# Google Chrome Browser

- We will need to use the Chrome Developer Tools
- Google “Chrome installer” or
- Download link
- [https://www.google.com/intl/en\\_hk/chrome/](https://www.google.com/intl/en_hk/chrome/)



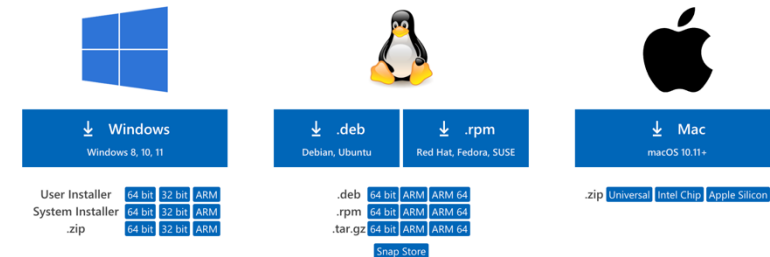
# Visual Studio Code

- Visual Studio Code is one of the most popular modern code editor.
- We will use VS Code for HTML, CSS and JS coding/editing
- Download link
  - <https://code.visualstudio.com/download>




## Download Visual Studio Code

Free and built on open source. Integrated Git, debugging and extensions.



# Node.js Download




- Node.js is a popular development tool and runtime for JavaScript
- Download link
  - <https://nodejs.org/en/download/>



The screenshot shows the Node.js website's download section. At the top is a dark navigation bar with the Node.js logo and links for HOME, ABOUT, DOWNLOADS, DOCS, GET INVOLVED, SECURITY, CERTIFICATION, and NEWS. Below the navigation bar, the 'Downloads' section is highlighted. It states the latest current version is 18.8.0 (including npm 8.18.0) and encourages downloading the source code or a pre-built installer. Two main tabs are visible: 'LTS' (Recommended For Most Users) and 'Current' (Latest Features). Under the 'LTS' tab, there are links for 'Windows Installer' (node-v18.8.0-x86.msi), 'macOS Installer' (node-v18.8.0.pkg), and 'Source Code' (node-v18.8.0.tar.gz). Under the 'Current' tab, there are links for '32-bit', '64-bit', '64-bit / ARM64', 'ARMv7', and 'ARMv8', all leading to the same source code file (node-v18.8.0.tar.gz).

**Downloads**  
Latest Current Version: 18.8.0 (includes npm 8.18.0)

Download the Node.js source code or a pre-built installer for your platform, and start developing today.

LTS Recommended For Most Users	Current Latest Features	
 Windows Installer node-v18.8.0-x86.msi	 macOS Installer node-v18.8.0.pkg	 Source Code node-v18.8.0.tar.gz

Windows Installer (.msi)	32-bit	64-bit
Windows Binary (.zip)	32-bit	64-bit
macOS Installer (.pkg)	64-bit / ARM64	
macOS Binary (.tar.gz)	64-bit	ARM64
Linux Binaries (x64)	64-bit	
Linux Binaries (ARM)	ARMv7	ARMv8
Source Code	node-v18.8.0.tar.gz	



# Source Codes Download

`github.com/ngsanluk/js-inter`

# JavaScript Syntax

- JS Variables and Data Types
- Arithmetic Operators
- Comparison Operators
- Condition Control
- Looping
- Function
- Getting Familiar with JavaScript Object

# JavaScript Object

- Use { } to denote an object
- On the right is an empty object
  - It is empty in-between brace

```
{  
}
```

# JavaScript Object Key/Value

- A JavaScript object with ONE pair of key and value
- The key is `firstName`
- The associate value is `"Peter"` and it's a **string** type
  - String value must be double-quoted

```
{  
  firstName: "Peter"  
}
```

# JavaScript Object Key/Value

- A JavaScript with TWO pair of keys and values
- , is used to separate each pair of key/value from each other
- The extra key is `lastName`
- The associate value is `"Pan"` and it's a string type

```
{  
  firstName: "Peter",  
  lastName: "Pan"  
}
```

# JavaScript Object Key/Value

- A JavaScript with THREE pair of keys and values
- , is used to separate each pair of key/value from each other
- The extra key is `age`
- The associate value is `18`
  - A number is NOT double-quoted

```
{  
  firstName: "Peter",  
  lastName: "Pan",  
  age: 18  
}
```

# JavaScript Array

- Use `[ ]` to denote an array
- On the right is an empty array

`[ ]`

# JavaScript is Nested

- On the right is an array with THREE empty objects

```
[  
  { } ,  
  { } ,  
  { }  
]
```



# JavaScript is Nested

- On the right is a JavaScript object
- One of the children elements is an array
  - tel
- The `tel` child element has two values "111" and "222".
  - Values are separated by ,

```
{  
  firstName: "Peter",  
  lastName: "Pan",  
  age: 18,  
  tel: ["111", "222"]  
}
```

# JavaScript is Nested

- The JavaScript object has a child object named `address`
- The address sub-object has two keys namely `street` and `district`

```
{  
  firstName: "Peter",  
  lastName: "Pan",  
  age: 18,  
  tel: ["111", "222"],  
  address: {  
    street: "Leighton Road",  
    district: "Causeway Bay"  
  }  
}
```

# Referring JavaScript Object

- On the right is an object named `student`
- To refer to the whole object
  - `student`
- To refer to `firstName`
  - `student.firstName`
- To refer to the `tel` number (array)
  - `student.tel`
  - To refer to the first `tel` number
  - `student.tel[0]`
- To refer to the `street` of address object
  - `student.address.street`

```
student = {  
  firstName: "Peter",  
  lastName: "Pan",  
  age: 18,  
  tel: ["111", "222"],  
  address: {  
    street: "Leighton Road",  
    district: "Causeway Bay"  
  }  
}
```

# JSON

# JavaScript Object $\neq$ JSON

## JavaScript object

```
{firstname : "Sam",  
  lastname : "Fernandes"}
```

↑  
Key

↑  
Value

(Need not be enclosed  
within double quotes)

## JSON object

```
{"firstname" : "Sam",  
  "lastname" : "Fernandes"}
```

↑  
Key

↑  
Value

(Must be enclosed  
within double quotes)

# Real-world JSON Data Source

Weather API by Hong Kong Observatory

- HK Weather Data Dictionary

- <https://data.weather.gov.hk/weatherAPI/opendata/weather.php>

- HK Weather Resource Endpoint / Link

- <https://data.weather.gov.hk/weatherAPI/opendata/weather.php?dataType=flw&lang=tc>

# HK Observatory Open API

## Data Dictionary

- [https://www.hko.gov.hk/en/weatherAPI/doc/files/HKO\\_Open\\_Data\\_API\\_Documentation.pdf](https://www.hko.gov.hk/en/weatherAPI/doc/files/HKO_Open_Data_API_Documentation.pdf)

### Request Example

<https://data.weather.gov.hk/weatherAPI/opendata/weather.php?dataType=flw&lang=en>

### Request

Parameter	Accepted values	Description
dataType	flw fnd rhrread warnsum warningInfo swt	flw: Local Weather Forecast fnd: 9-day Weather Forecast rhrread: Current Weather Report warnsum: Weather Warning Summary warningInfo: Weather Warning Information swt: Special Weather Tips
lang	en tc sc	en: English tc: Traditional Chinese sc: Simplified Chinese Default language: en

# 9-day Weather Forecast (Raw Data)

<https://data.weather.gov.hk/weatherAPI/opendata/weather.php?dataType=fnd&lang=en>

JSON Raw Data Headers

Save Copy Pretty Print

```
{
  "generalSituation": "A fresh to strong easterly airstream will affect the coast of southern China in the next couple of days. A cold front is expected to reach the coast of Guangdong on Saturday. Temperatures will fall appreciably over the region. Under the influence of an intense winter monsoon and upper-air disturbance, it will remain cold with some rain over southern China during the weekend to midweek next week.",
  "weatherForecast": [
    {
      "forecastDate": "20220216",
      "week": "Wednesday",
      "forecastWind": "East force 4 to 5, occasionally force 6 offshore later.",
      "forecastWeather": "Mainly cloudy. Sunny periods during the day.",
      "forecastMaxtemp": {
        "value": 19,
        "unit": "C"
      },
      "forecastMintemp": {
        "value": 16,
        "unit": "C"
      },
      "forecastMaxrh": {
        "value": 90,
        "unit": "percent"
      },
      "forecastMinrh": {
        "value": 70,
        "unit": "percent"
      },
      "ForecastIcon": 51,
      "PSR": "Low"
    },
    {
      "forecastDate": "20220217",
      "week": "Thursday",
      "forecastWind": "East force 5, force 6 offshore and on high ground.",
      "forecastWeather": "Cloudy with one or two rain patches.",
      "forecastMaxtemp": {
        "value": 19,
        "unit": "C"
      },
      "forecastMintemp": {
        "value": 17,
        "unit": "C"
      },
      "forecastMaxrh": {
        "value": 95,
        "unit": "percent"
      },
      "forecastMinrh": {
        "value": 80,
        "unit": "percent"
      },
      "ForecastIcon": 60,
      "PSR": "Low"
    },
    {
      "forecastDate": "20220218",
      "week": "Friday",
      "forecastWind": "East force 4 to 5, force 6 offshore and on high ground at first.",
      "forecastWeather": "Cloudy with a few rain and mist patches.",
      "forecastMaxtemp": {
        "value": 19,
        "unit": "C"
      },
      "forecastMintemp": {
        "value": 17,
        "unit": "C"
      },
      "forecastMaxrh": {
        "value": 95,
        "unit": "percent"
      },
      "forecastMinrh": {
        "value": 85,
        "unit": "percent"
      },
      "ForecastIcon": 62,
      "PSR": "Medium Low"
    },
    {
      "forecastDate": "20220219",
      "week": "Saturday",
      "forecastWind": "North to northeast force 4 to 5.",
      "forecastWeather": "Becoming cold appreciably. Cloudy with occasional rain.",
      "forecastMaxtemp": {
        "value": 17,
        "unit": "C"
      },
      "forecastMintemp": {
        "value": 12,
        "unit": "C"
      },
      "forecastMaxrh": {
        "value": 95,
        "unit": "percent"
      },
      "forecastMinrh": {
        "value": 80,
        "unit": "percent"
      },
      "ForecastIcon": 92,
      "PSR": "High"
    },
    {
      "forecastDate": "20220220",
      "week": "Sunday",
      "forecastWind": "North to northeast force 4 to 5, occasionally force 6 offshore.",
      "forecastWeather": "Cloudy to overcast with occasional rain. It will be cold.",
      "forecastMaxtemp": {
        "value": 13,
        "unit": "C"
      },
      "forecastMintemp": {
        "value": 11,
        "unit": "C"
      },
      "forecastMaxrh": {
        "value": 95,
        "unit": "percent"
      },
      "forecastMinrh": {
        "value": 80,
        "unit": "percent"
      },
      "ForecastIcon": 93,
      "PSR": "High"
    }
  ]
}
```



# Use **Firefox** for better JSON viewing

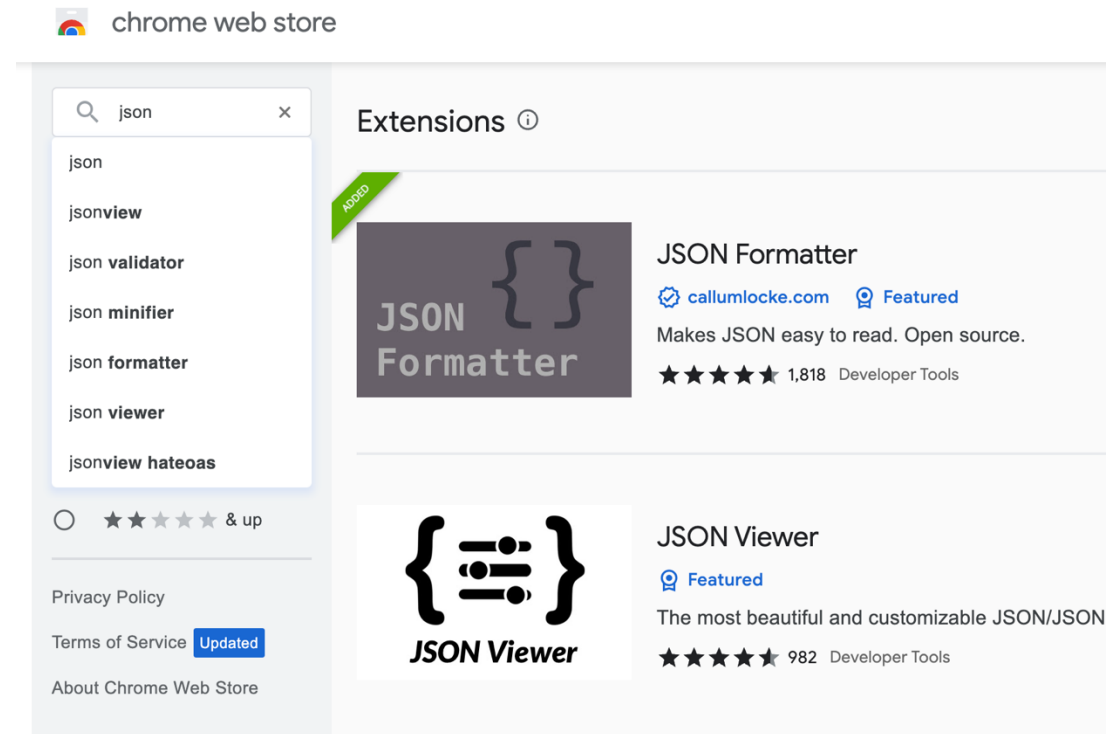
The screenshot displays the Firefox JSON Viewer interface. At the top, there are tabs for 'JSON', 'Raw Data', and 'Headers', with 'JSON' being the active tab. Below the tabs is a toolbar with buttons for 'Save', 'Copy', 'Collapse All', 'Expand All', and a 'Filter JSON' input field. The main content area shows a JSON object with two main properties: 'generalSituation' and 'weatherForecast'. The 'generalSituation' property is expanded, showing a long text description of the weather. The 'weatherForecast' property is also expanded, revealing an array of forecast objects. The first object in the array, indexed '0', is further expanded to show detailed forecast data including date, week, wind, weather, and temperature ranges.

```
{
  "generalSituation": "A fresh to strong easterly airstream will affect the coast of southern China in the next couple of days. A cold front is expected to reach the coast of Guangdong on Saturday. Temperatures will fall appreciably over the region. Under the influence of an intense winter monsoon and upper-air disturbance, it will remain cold with some rain over southern China during the weekend to midweek next week.",
  "weatherForecast": [
    {
      "forecastDate": "20220216",
      "week": "Wednesday",
      "forecastWind": "East force 4 to 5, occasionally force 6 offshore later.",
      "forecastWeather": "Mainly cloudy. Sunny periods during the day.",
      "forecastMaxtemp": {
        "value": 19,
        "unit": "C"
      },
      "forecastMintemp": {
        "value": 16,
        "unit": "C"
      }
    }
  ]
}
```

# Or install Google Extensions

Go to **Chrome Web Store** and search “**JSON**”

<https://chrome.google.com/webstore/search/json?hl=en>



# JavaScript fetch API

# fetch API

- The fetch API provides a JavaScript interface for accessing and manipulating parts of the HTTP pipeline, such as **requests** and **responses**.
- It also provides a global `fetch()` method that provides an easy, logical way to fetch resources (usually in JSON format) asynchronously across the network.
- [https://developer.Mozilla.Org/en-us/docs/web/API/fetch\\_api/using\\_fetch](https://developer.mozilla.org/en-us/docs/web/API/fetch_api/using_fetch)

# fetch Sample Codes

```
fetch('https://data.weather.gov.hk/weatherAPI/opendata/weather.php?dataType=fnd&lang=tc')  
  .then(response => response.json())  
  .then(data => {  
    console.log(data.generalSituation)  
  })  
);
```

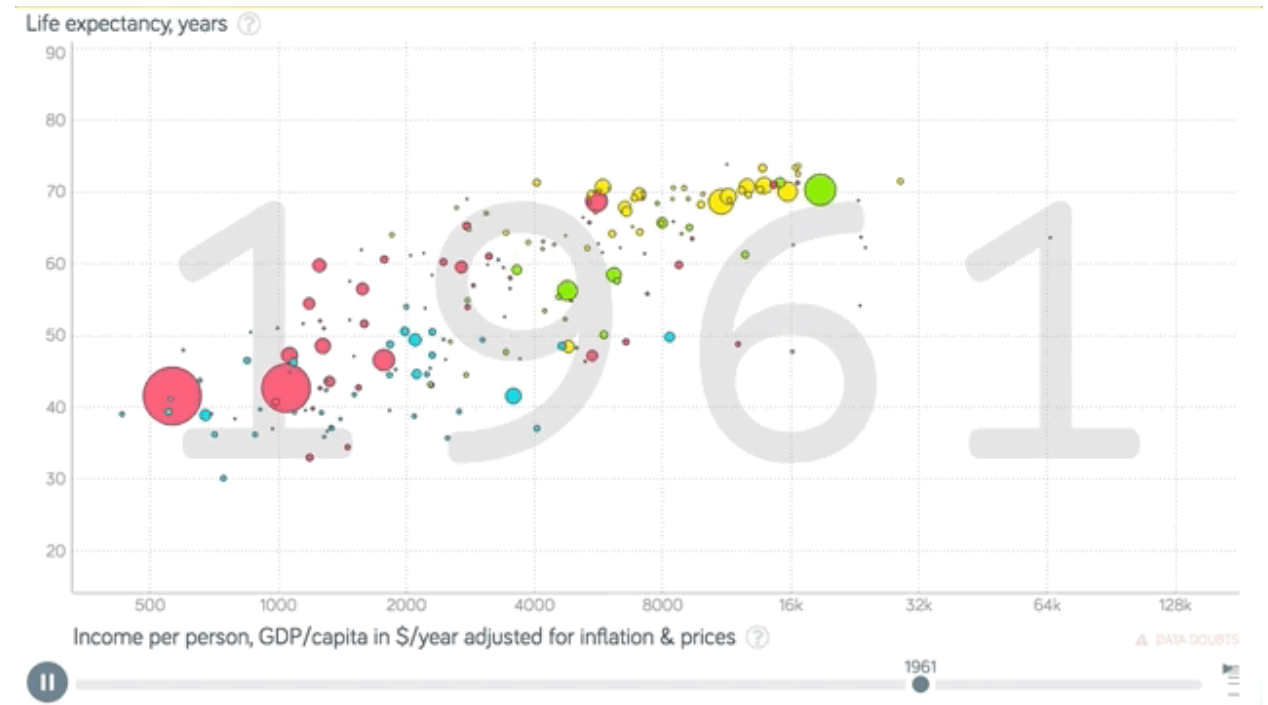
# fetch usage

- We are fetching a JSON file across the internet and printing it to the console (the windows for debug/dev)
- The simplest use of `fetch()` takes one argument — the path to the web resource (URL) you want to fetch
- The Response object, in turn, does not directly contain the actual JSON response body but is instead a representation of the entire HTTP **response**.
- So, to extract the JSON body content from the Response object, we use the `json()` method which returns the response body text as JS object

# Interactive Data Visualization

# Interactive Data Visualization

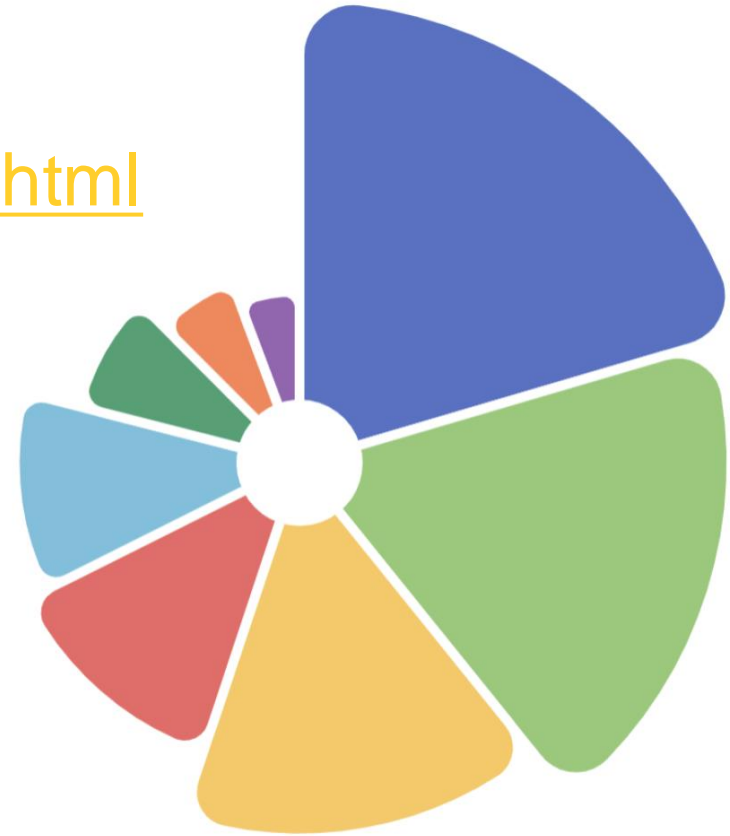
- Web-based
- Powered by JavaScript
- Presentation is dynamic
- Allows user interactions





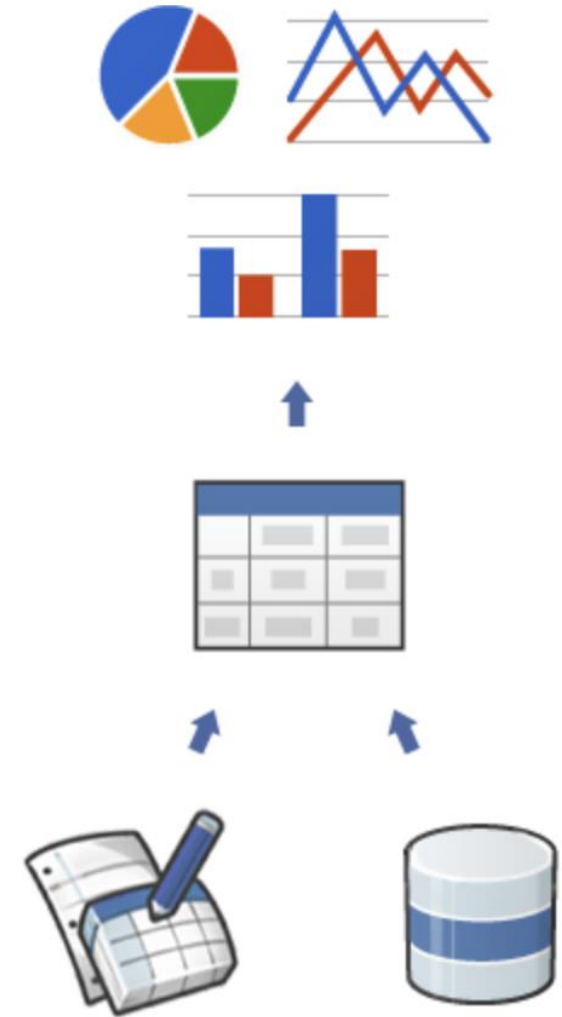
# Apache eCharts

- <http://echarts.apache.org/en/index.html>



# Google Charts

- <https://developers.google.com/chart>





Interactive Programming: JS Notebook

# Observable

[observablehq.com/](https://observablehq.com/)

Explore Learn Community Solutions ... Search

New

## Exploring Olympic athletes data

id	name	nationality	sex	date_of_birth	height	weight	sport	gold
736,041,664	A Jesus Garcia	ESP	male	1969-10-17	1.72	64	athletics	1
532,037,425	A Lam Shin	KOR	female	1986-09-23	1.68	56	fencing	1
435,962,603	Aaron Brown	CAN	male	1992-05-27	1.98	79	athletics	1
521,041,435	Aaron Cook	MDA	male	1991-01-02	1.83	80	taekwondo	1
33,922,579	Aaron Gate	NZL	male	1990-11-26	1.81	71	cycling	1
173,071,782	Aaron Royle	AUS	male	1990-01-26	1.8	67	triathlon	1
266,237,702	Aaron Russell	USA	male	1993-06-04	2.05	98	volleyball	1
382,571,888	Aaron Younger	AUS	male	1991-09-25	1.93	100	aquatics	1
87,689,776	Auri Lorena Bokesa	ESP	female	1988-12-14	1.8	62	athletics	1
997,877,719	Ababel Yeshaneh	ETH	female	1991-07-22	1.65	54	athletics	1

```
viewof table = Inputs.table(olympians)
```

### Explore Olympic athletes with charts

1 of 7

This guide builds on [First steps with Observable](#).

Let's explore some data! You're probably familiar with the tabular view of data, like the one you see on the left. Each entry (or in our case, athlete) is represented by a row, and each column contains a different piece of data about them.

To see what data in Observable looks like, create a new JavaScript cell by clicking a **+** button on the left and selecting **JavaScript**. Then type the variable name `olympians` into your new cell. This is one of the [sample datasets from our standard library](#).

```
olympians
```

Run the cell by clicking **Run cell** or pressing **shift-return**. You will see the word `Array` with a triangle next to it. An array is a group of objects that usually share the same structure. Click the triangle to expand the array into its objects, then click the triangle next to an object to see its properties.

Practicing  
Observable