

SgLDC Virtual Meet 1 2024

Creating EdTech Apps with ChatGPT

TAN SENG KWANG
ST (PHYSICS)

✉ tan_seng_kwang@moe.edu.sg

🌐 physicslens.com

📍 Temasek Junior College

CONTENTS

1. Live Demonstration

2. The Future of Coding

3. Examples in Physics

4. Other Examples

5. Workflow

6. Recommendations

7. Limitations

html

Copy code

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <style>
    canvas {
      border: 1px solid #000;
    }
  </style>
  <title>Water Molecule Simulation</title>
</head>
<body>
  <h2>Water Molecule Simulation</h2>
  <label for="energySlider">Average Energy:</label>
  <input type="range" id="energySlider" min="1" max="10" value="5">
  <span id="energyValue">5</span>

  <canvas id="waterCanvas" width="800" height="400"></canvas>

  <script>
    const canvas = document.getElementById("waterCanvas");
    const ctx = canvas.getContext("2d");
    const energySlider = document.getElementById("energySlider");
    const energyValue = document.getElementById("energyValue");

    let molecules = [];

    function randomColor(energy) {
      const red = Math.floor(255 - energy * 20);
      const green = Math.floor(255 - energy * 10);
      const blue = Math.floor(255 - energy * 5);
      return `rgb(${red}, ${green}, ${blue})`;
    }
  </script>
</body>
</html>
```



You

Create a rock, paper, scissors game using html5 and javascript. The user will choose one of the 3 signs and the computer will randomly generate one. The winner will get a point. Keep score and show it as the game progresses.



ChatGPT

Certainly! Below is a simple example of a rock, paper, scissors game using HTML5 and JavaScript. You can copy and paste this code into an HTML file and open it in a web browser to play the game:

```
html Copy code
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Rock, Paper, Scissors Game</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      text-align: center;
      margin: 20px;
    }
    #result {
      font-size: 18px;
      margin-top: 10px;
    }
  </style>
</head>
<body>

<h1>Rock, Paper, Scissors Game</h1>

<button onclick="playGame('rock')">Rock</button>
<button onclick="playGame('paper')">Paper</button>
<button onclick="playGame('scissors')">Scissors</button>

<p id="result">Choose your move!</p>
<p>Player Score: <span id="playerScore">0</span> | Computer Score: <span id="computerScore">0</span>

<script>
  let playerScore = 0;
  let computerScore = 0;

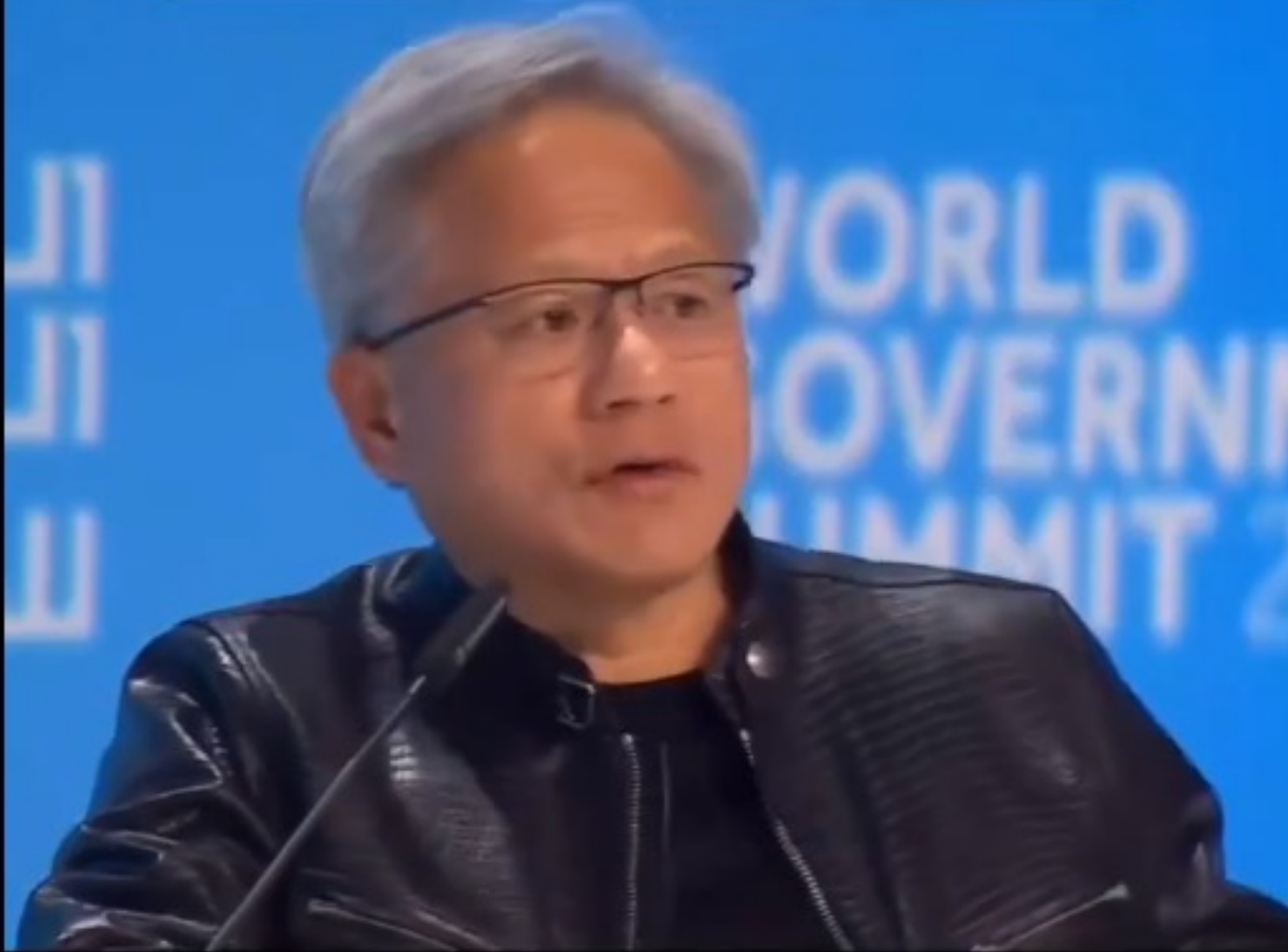
  function playGame(playerMove)
    const moves = ['rock', 'paper', 'scissors'];
```

A quick live demo

Rock, Paper, Scissors game

Try it with Google's **Gemini** or Microsoft's **CoPilot** too

**Jensen Huang (Nvidia CEO)
on what students should
study in school now**



DON'T LEARN TO CODE?

Over the course of the last 10 years, 15 years, almost everybody who sits on a stage like this would tell you that it is vital that your children learn computer science. [That] everybody should learn how to program. And in fact, it's almost exactly the opposite.

*It is our job to create computing technology such that nobody has to program and that the programming language is human. **Everybody in the world is now a programmer.***

This is the miracle of artificial intelligence.

MINISTER'S RESPONSE TO NVIDIA CEO'S COMMENT



"我们不一定要去学 coding。不过最重要的是我们怎么在逻辑思维, 数学那方面给我们的学生打好一个基础。"

It's not a must to learn coding. However, what's more important is how we give our students a foundation in logical thinking and mathematical skills.

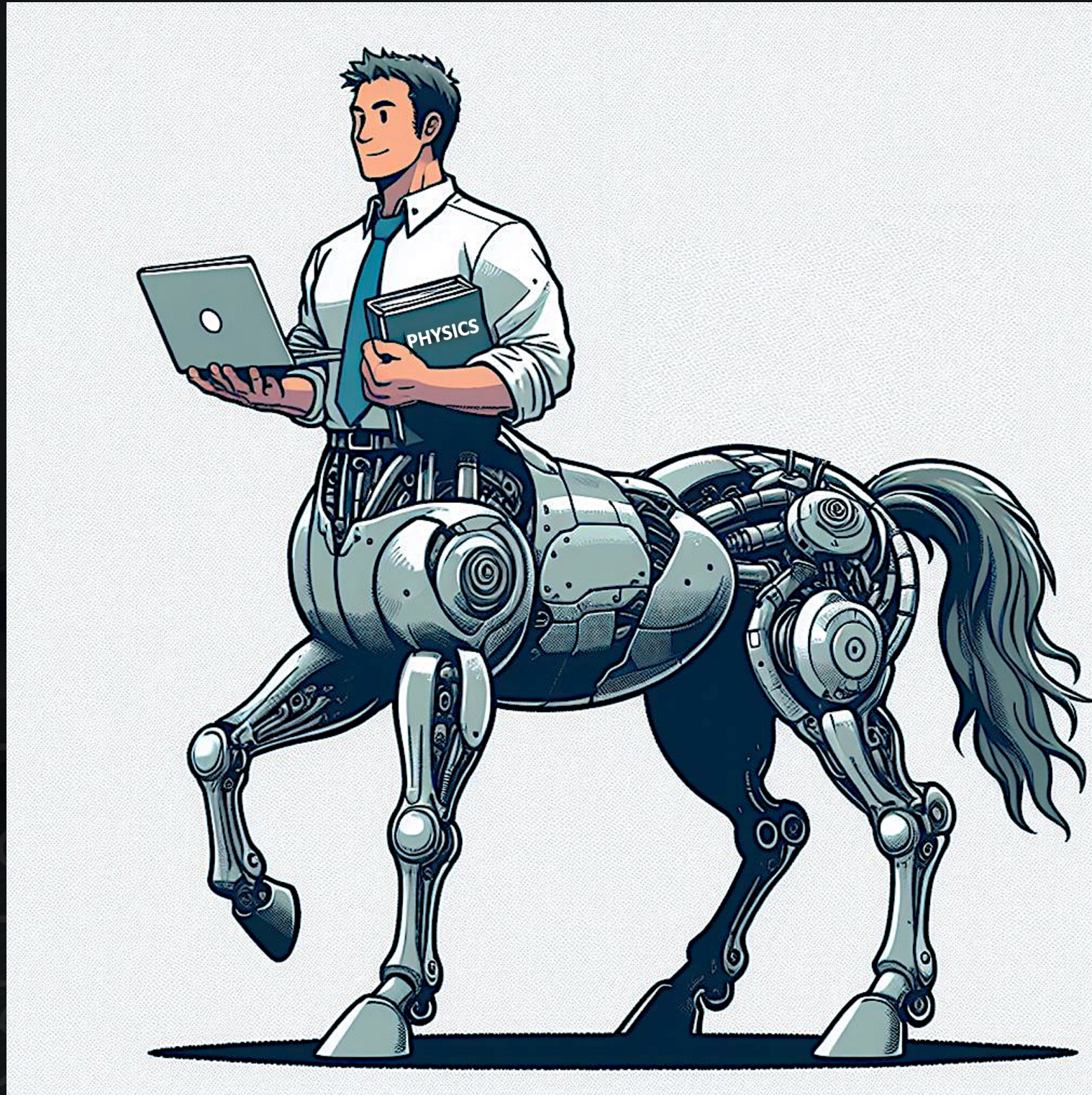
Min Chan, 4 Mar 2023

S'pore to triple AI talent pool to 15,000 as part of national strategy update: DPM Wong



“Knowledge-based work like research, coding and writing was considered safe from disruption in the past. But with AI, that is no longer the case”

DPM Lawrence Wong, 4 Dec 2023



CENTAUR > MAN || MACHINE

While experts will always be in demand, more intelligent machines are continually lowering the bar to creating with new technology.

Kasparov, 2017

Lower barrier of entry for app development; allow for focus on higher levels of problem solving

Bull and Kharrufa, 2023

POSSIBLE USE CASES FOR CODING WITH AI

01

Integrate T&L interactives seamlessly into SLS for free!

03

Simplify work through Robotic Process Automation, e.g. python, Microsoft VBA

02

Kickstart ALP/science research projects for students, e.g. micro:bit, Arduino

04

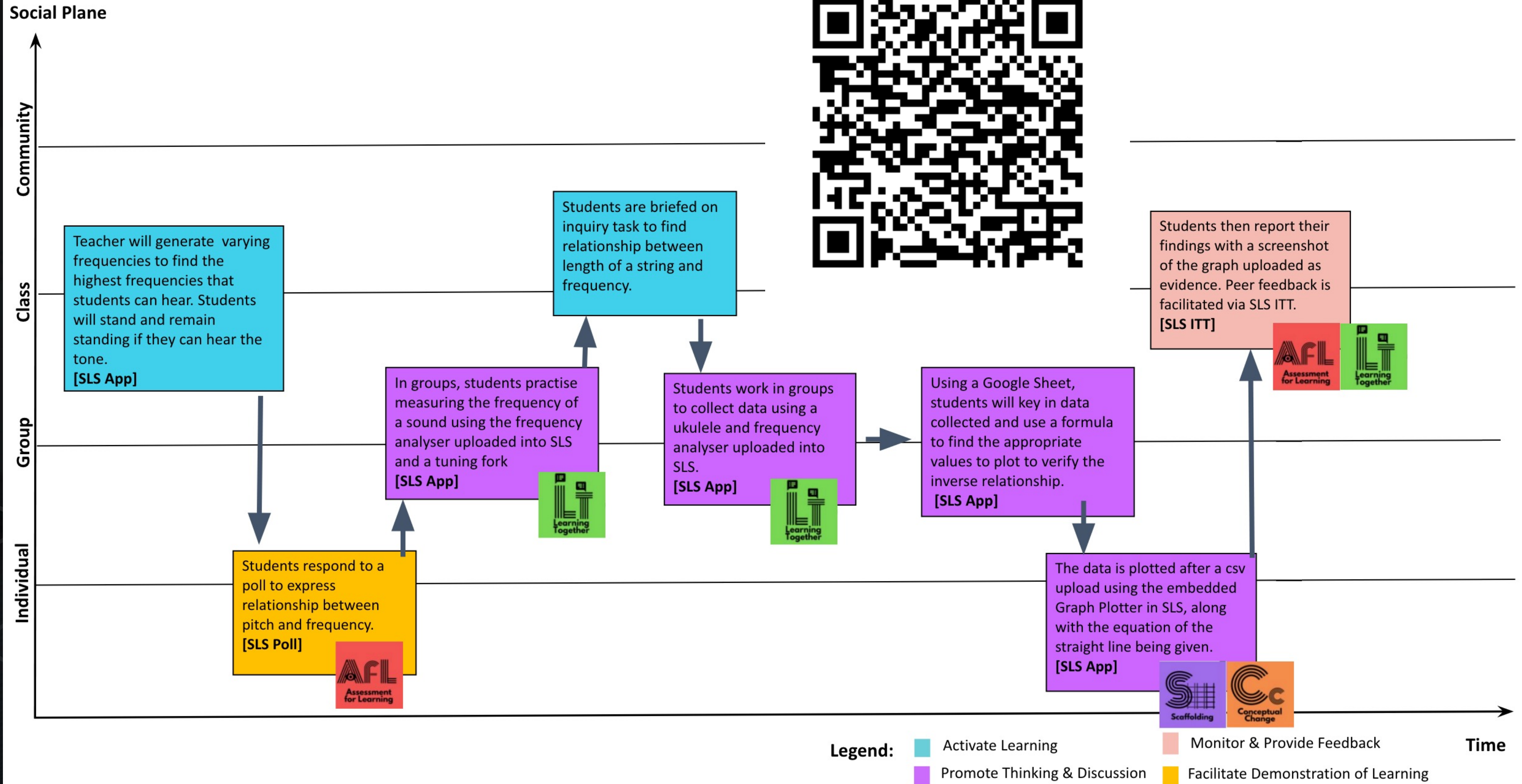
Analyse data (e.g. RStudio, python or pair.gov.sg for "Restricted/Sensitive Normal" data)



Overview of an Inquiry Learning Experience

Sec 3 / General Wave Properties

Duration: 60 minutes





<https://vle.learning.moe.edu.sg/community-gallery/lesson/view/f9c45429-0261-4999-aaeb-f8ded90fe8eb/page/73124376>

An example in the SLS
Community Gallery

INTEGRATING INTERACTIVES WITH SLS

Pitch and frequency

Optional

In this series of activities, you will plan and conduct experiments to investigate the factors affecting the pitch produced, quantified using the physical quantity of **frequency**, when a string is plucked.

Before we begin the experiment, to demonstrate the relationship between **pitch** and **frequency**, you can use this sound generator.

Sound Generator

Use the slider or input box to control the frequency of the generated sound.

Frequency: Hz



Play

Stop

 sound-generator.zip



The default frequency given is 440 Hz. Click the play button to hear the tone.

Change the frequency to any value between 200 to 1000 Hz and listen to how the sound changes.

2 Measuring frequency

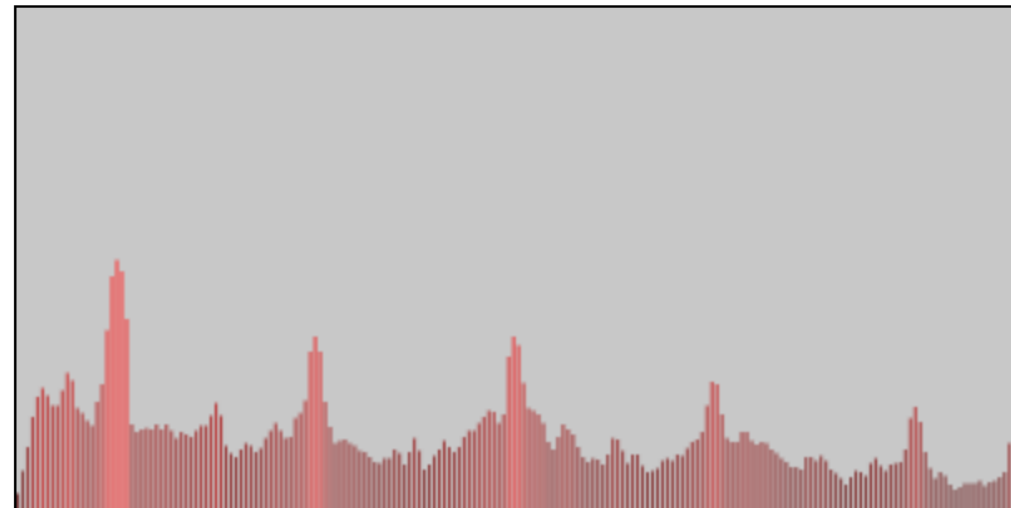
Optional

How can you measure the frequency of sound? In this Harness activity, you will learn to measure the frequency of the sound produced by a tuning fork.

There are a number of ways to measure the frequency of sound with the use of technology. The basic setup usually involves a microphone connected to a scope (e.g. oscilloscope) that records, analyses and displays the frequencies.

The following is an app written for use with your PLD. Click on the button at the bottom right corner of the app to expand it and to activate the microphone.

Sound Frequency Analyser



Dominant Frequency: 430.66 Hz

 sound-frequency-analyser.zip



Check if the dominant frequency is the same as that indicated in the tuning fork.

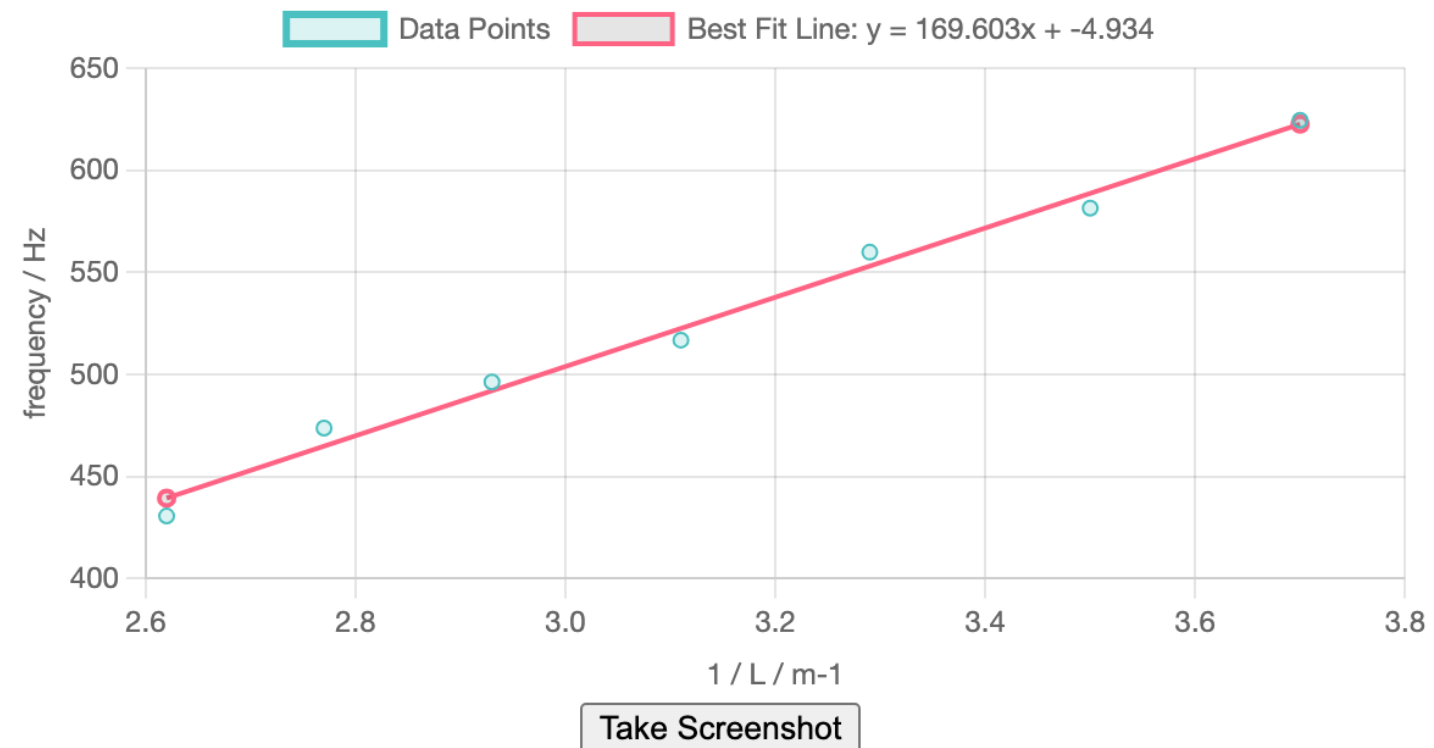
DATA COLLECTION WITHIN SLS



DATA ANALYSIS WITHIN SLS

Graph Plotter

Choose file Frequency a... of String.csv



graph-plotter.zip

Upload the csv file into the graph plotter above.

[Read Less](#)

	A	B	C	D
1	L / m	1 / L / m-1	frequency / Hz	
2	0.382	2.62	431	
3	0.361	2.77	474	
4	0.341	2.93	496	
5	0.322	3.11	517	
6	0.304	3.29	560	
7	0.286	3.50	581	
8	0.270	3.70	624	
9				

ADVANTAGES

NO NEED TO INSTALL AND LEARN
MULTIPLE NEW APPS

REDUCES COGNITIVE LOAD DUE TO
SIMPLIFIED U.I.

SEAMLESS EXPERIENCE WITH SLS AND PLD





OTHER EXAMPLES

<https://physicstjc.github.io/sls>



Flashcards

Previous

Orange

Next

Definition?

An orange fruit

Upload a CSV file ([link to sample](#)) with the terms and definitions:

Choose file sampleq.csv

Student Flashcards

This is the second flashcard app in this collection. The main difference is that students get to upload their own terms and definitions in the form of a csv file.

Keyboard entries are taken as alternative input method.

Test it out

Download for SLS

Prompts

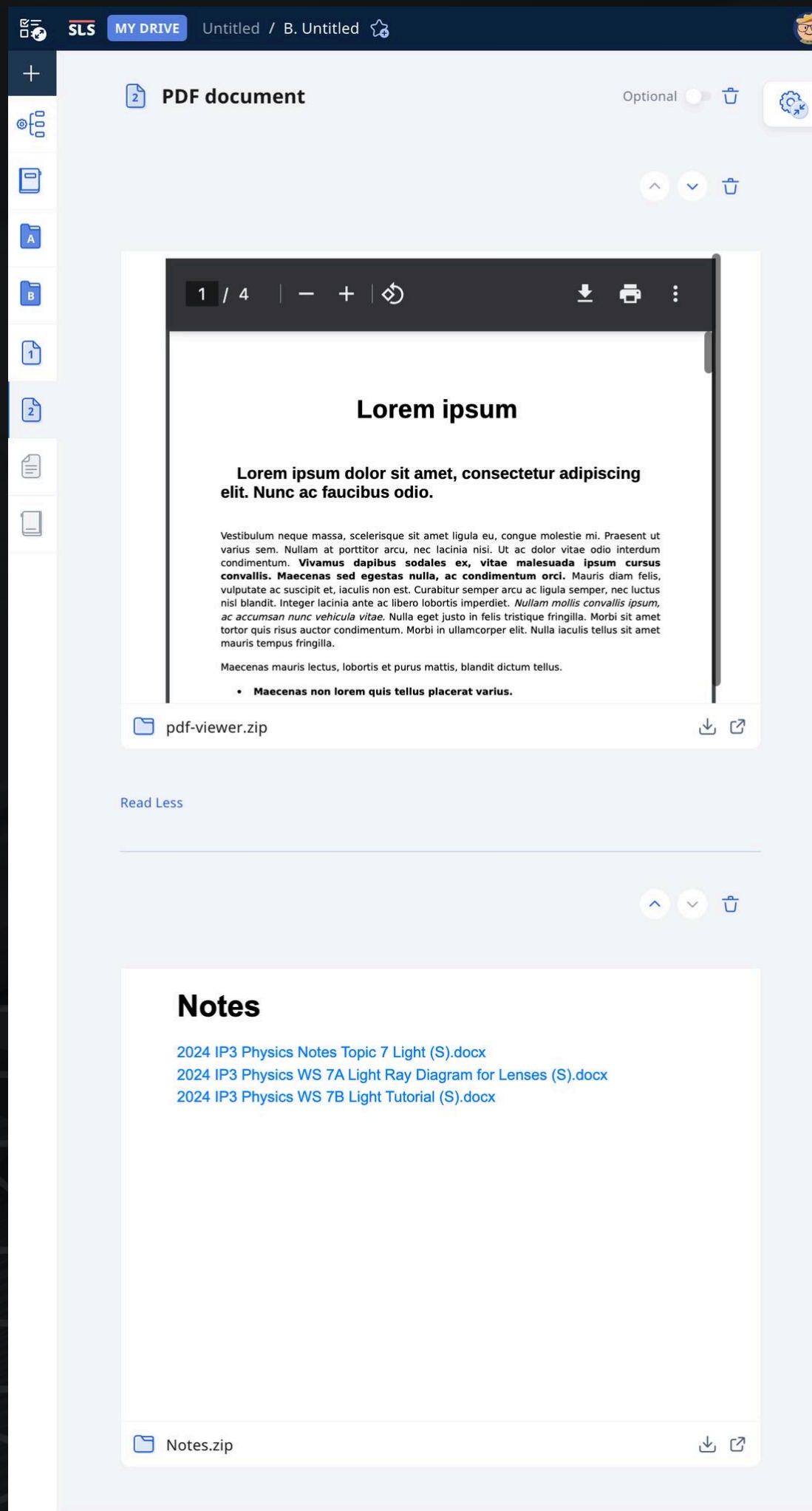
Prompts

1. Create a revision webpage using html, javascript and css that references a csv file in the same folder with two columns: "term" and "definition".
2. The csv file is to be uploaded by the user.
3. Each term in the "term" field will be displayed in turn using a left and right button. Another button labelled "Definition" will be used to show or hide the corresponding "definition" field at the bottom. Put all the script and style codes in the html file.

PRODUCTIVITY TOOLS

- Student can upload a csv file of terms and definitions to create digital flashcards

<https://physicstjc.github.io/sls/student-flashcards>

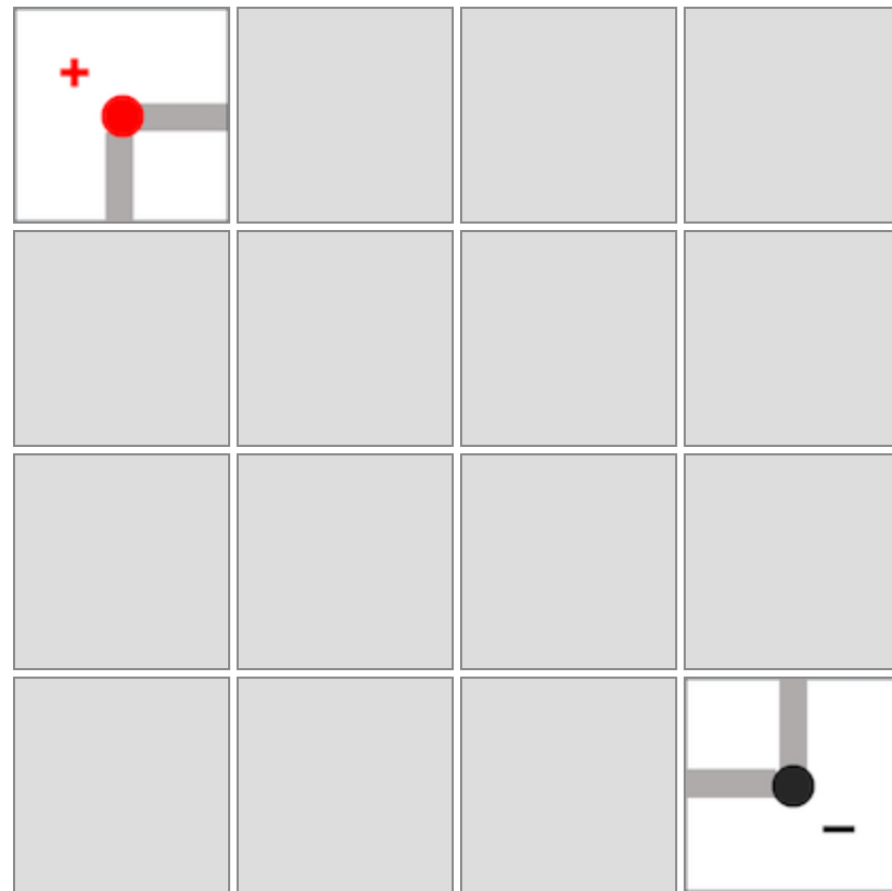


MODIFYING SLS EXPERIENCE

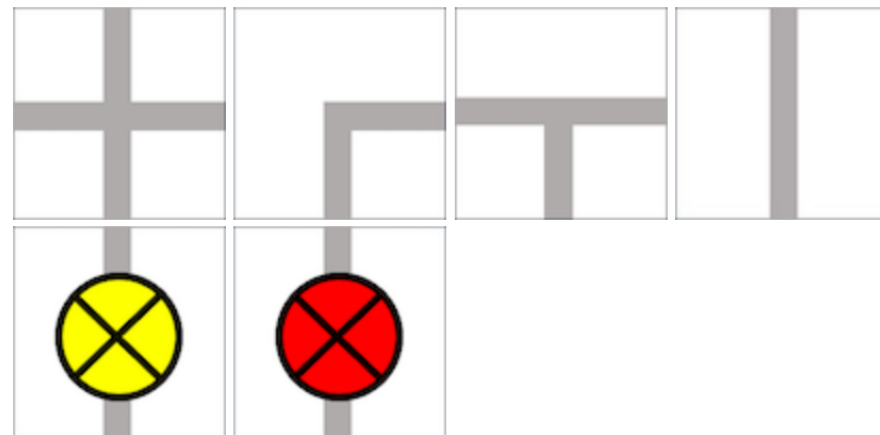
- Allowing PDF files to load in frame
- List downloadable files in a zip file

<https://physicstjc.github.io/sls/pdf-viewer>
<https://physicstjc.github.io/sls/zip-directory>

DELIGHT - electrifying fun



Undo



Rules

01. The aim is to light up as many bulbs of your colour as possible.
02. Players will take turns to place one piece on the 4-by-4 game board by clicking to select the electrical component and clicking on the square on the board to place it.
03. Upon placing the piece, the player can also turn that piece in any orientation (by clicking on it) within the same turn.
04. Players can choose to use up to two turns at any point in the game to rotate any piece that had been placed by any player.
05. In other words, each player has 9 turns: 7 placement turns and 2 rotation turns.

GAME-BASED LEARNING

- Turn-based game to place wires to light up your bulb
- Sabotage your opponent by bypassing or blocking current

<https://physicstjc.github.io/sls/delight/>

ASSESSMENT TOOL

tRAT Quiz

Question 6



Score: 16

[Previous](#) [Next](#)

- Replaces physical scratch card
- Facilitates discussion among team members before consensus

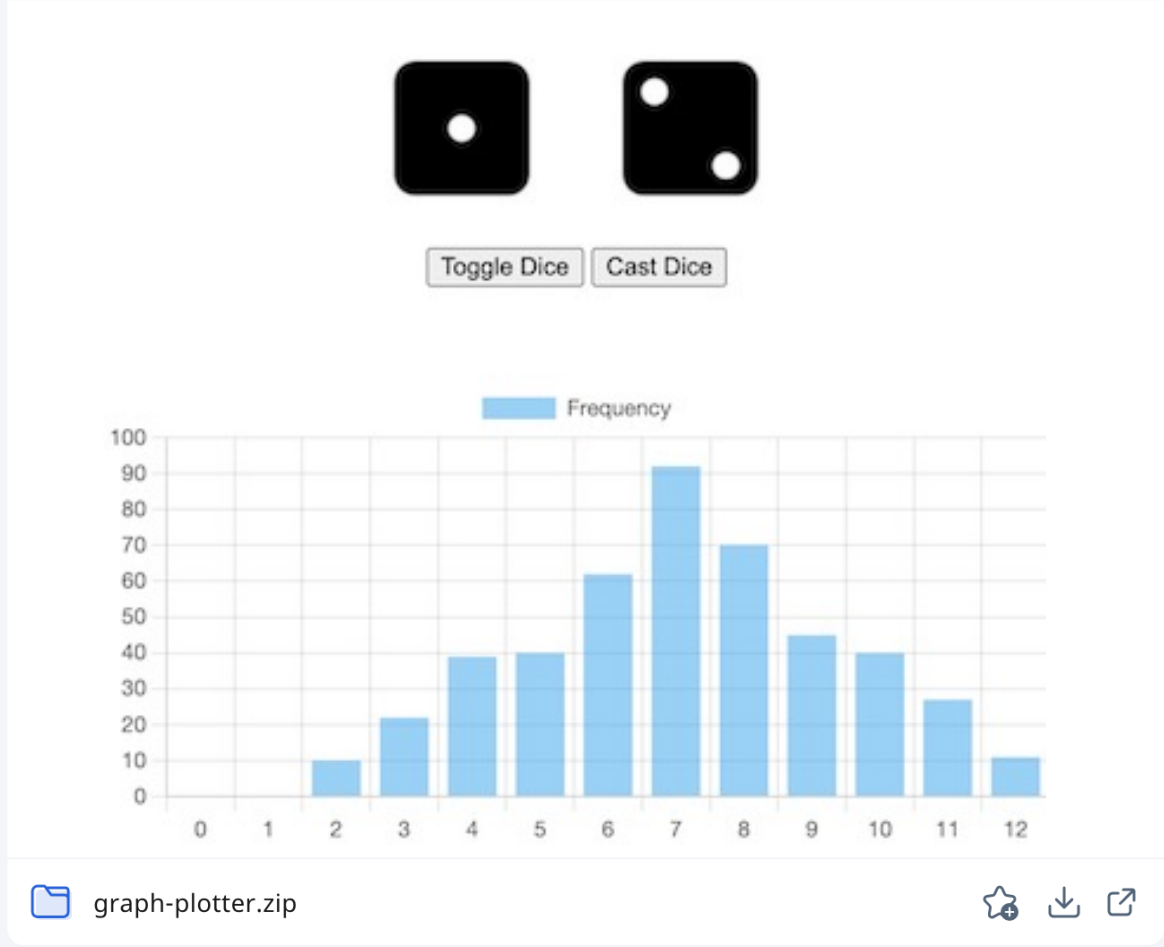
<https://physicstjc.github.io/sls/trat>

SLS MY DRIVE Untitled / A. Untitled

You're in the Lesson View mode.

Optional

Untitled



graph-plotter.zip

Read Less

Sum	Frequency
2	10
3	20
4	30
5	40
6	50
7	60
8	70
9	80
10	90
11	100
12	100

VISUALISATION FOR MATH

- Learning statistics representation
- Inquiry activity to verify expected probability distribution
- Accounting for differences due to randomness

<https://physicstjc.github.io/sls/dice>

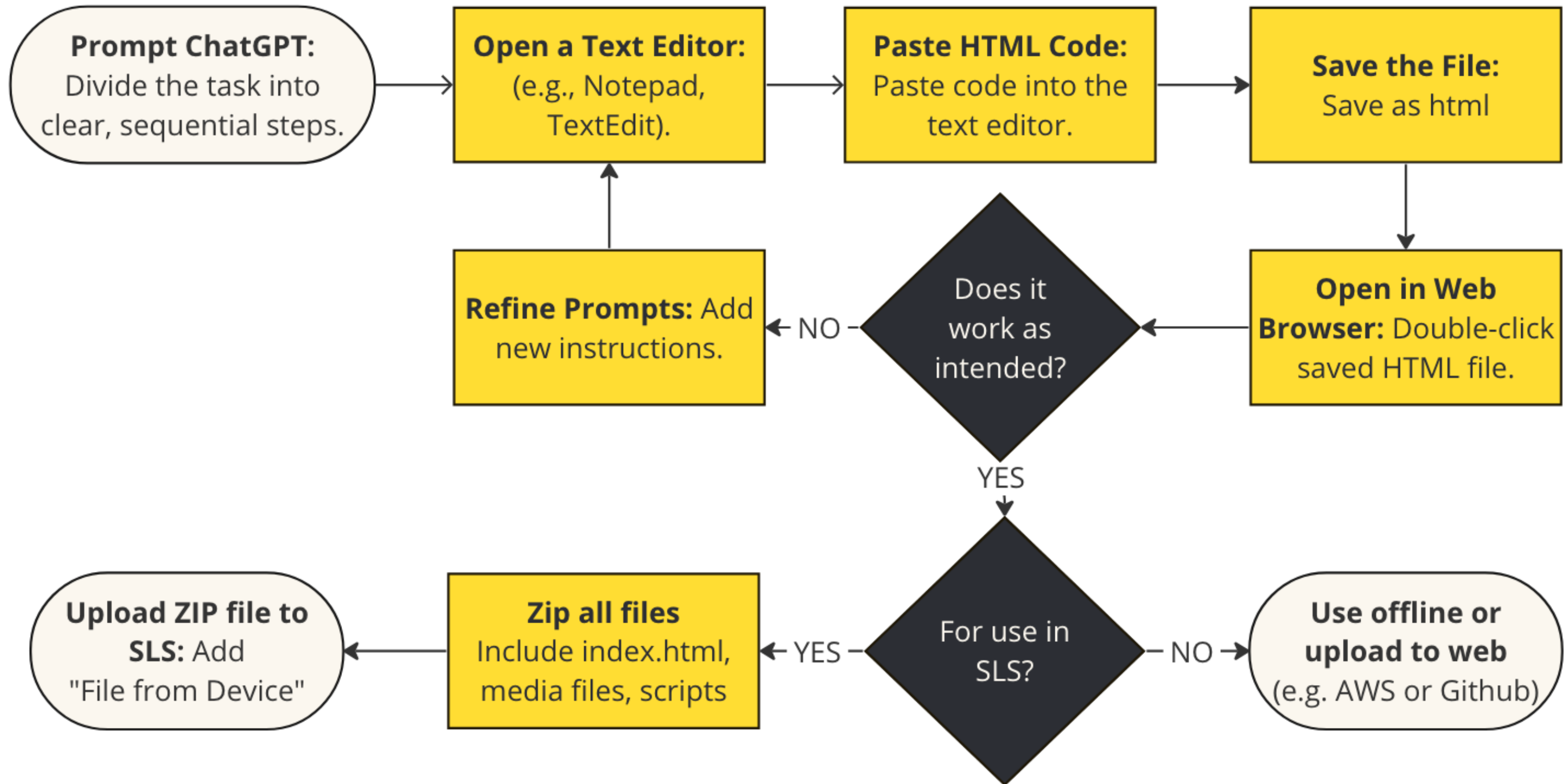
VIDEO TUTORIAL

The video player shows a YouTube interface with the following elements:

- Header:** YouTube logo, search bar, microphone icon, and 'Sign in' button.
- Browser Tab:** 'ChatGPT' and 'Prime Number Guessing Game'.
- URL:** chat.openai.com/c/6066a7a9-4835-4c45-8785-5a3b2792e7c9
- ChatGPT 3.5:** A chat window with the following text:
 - intended. We'll generate a random prime number between 1 and 1000, and then compare the user's guess with this randomly generated prime number.
 - Here's the updated code:
- Code Editor:** A dark-themed editor showing HTML and CSS code for a 'Prime Number Guessing Game'.

```
html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Prime Number Guessing Game</title>
  <style>
    body {
      font-family: Arial, sans-serif;
    }
  </style>
  <input type="number" />
```
- Video Player:** Progress bar at 3:34 / 6:21, play/pause, volume, and other controls.
- Caption:** ChatGPT can make mistakes. Consider checking important information.
- Taskbar:** Windows taskbar with various application icons and system tray.
- Video Title:** Generating standalone web apps for SLS using ChatGPT
- Channel:** Physics Lens, 554 subscribers
- Interactions:** Subscribe button, 2 likes, dislike, share, save, and more options.

WORKFLOW



PROMPT ENGINEERING

Common Prompt Engineering Basics

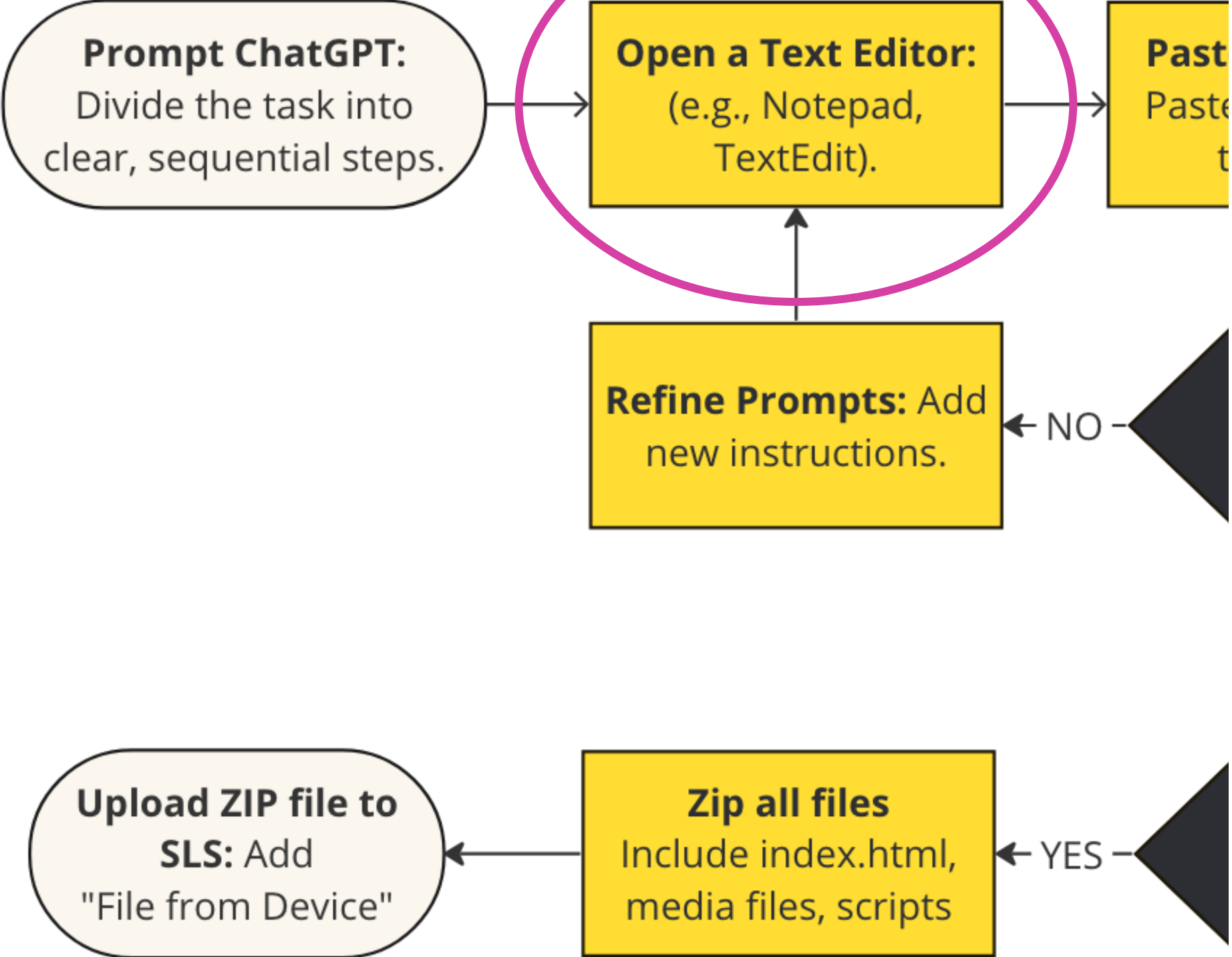
- Instruction, e.g. "Develop a website to demonstrate Brownian motion"
- Context, e.g. "There will be 500 grey particles in a box moving in straight lines at random speeds and directions. A red particle will start from the centre of the box. All the 501 particles are colliding with each other and the walls of the box."
- Input, e.g. "There should be a slider to increase the speed of the particles"
- Output, e.g. "Use Javascript and HTML5 and putting all the codes in one file"

My Recommendations

- Be specific on what the app should accomplish.
- Use common interactive elements like buttons, input boxes and sliders.
- Place all the codes in one file for starters

WORKFLOW

Web-based Code Editor
produced with ChatGPT!



<https://physicstjc.github.io/sls/converter>

WORKFLOW

Include path of any media file (e.g. images, audio, video)



You

Instead of using text, use images to represent rock, paper and scissors. The file names are "rock.png", "paper.png" and "scissors.png" respectively.

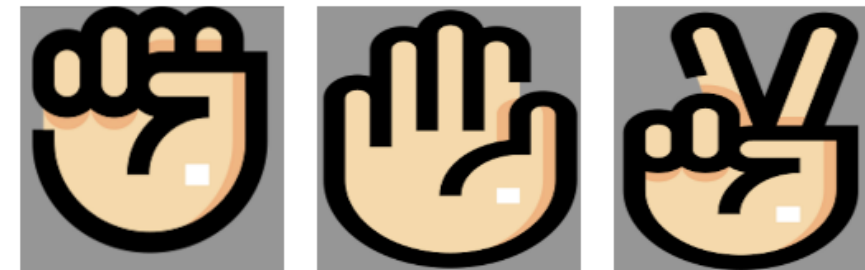


ChatGPT

Certainly! You can modify the code to use images for rock, paper, and scissors. Make sure to have the image files (rock.png, paper.png, and scissors.png) in the same directory as your HTML file. Here's an updated version of the code:

```
html Copy code  
  
<div id="playerChoice">  
    
    
    
</div>
```

Rock, Paper, Scissors Game



Choose your move!

Player Score: 0 | Computer Score: 0

WORKFLOW

Copy external scripts, if any

```
<script src="https://cdn.plot.ly/plotly-latest.min.js"></script>  
<style>
```

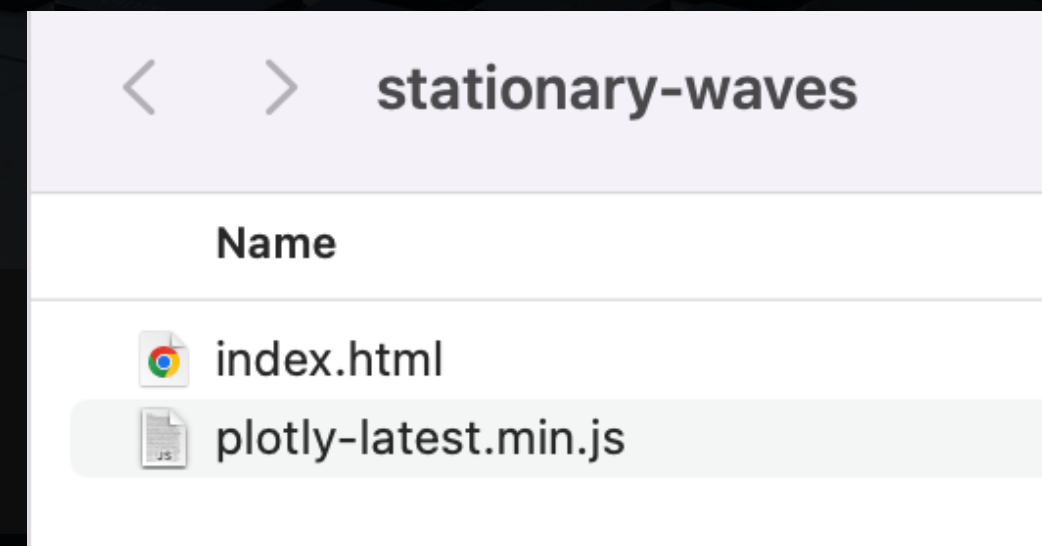
Paste the link on browser.

Copy the entire page and paste in an editor.

Save it as .js extension in the same directory as the html file.


Change the file path in the html.


```
<script src="plotly-latest.min.js"></script>  
<style>
```




DEBUGGING

If you are not sure, just describe the unexpected behaviour.


 **You**
The speed slider is now affecting the number of particles, which should not be the case




 **ChatGPT**
I apologize for the oversight. I've corrected the code so that the speed slider only affects the speed of particles, not their number. Here's the updated code:

```
html Copy code  
  
<!DOCTYPE html>  
<html lang="en">
```

If it still does not work after some iterations, ask it to try a new approach.

 **You**
Does not seem to work. Is there another way

 **ChatGPT**
Certainly! We can use a combination of FileReader, JSZip, and Blob APIs to achieve the desired functionality. This approach will handle the zipped folder processing entirely on the client-side without relying on experimental APIs.

Here's how you can implement it:

```
html Copy code  
  
<!DOCTYPE html>  
<html lang="en">  
<head>
```

DEBUGGING

If you understand the codes,
tell it straight

OR just change it yourself!



You

```
function createMolecule() {  
  const x = Math.random() * canvas.width;  
  const y = Math.random() * (canvas.height / 2);
```

should be

```
function createMolecule() {  
  const x = Math.random() * canvas.width;  
  const y = canvas.height - Math.random() * (canvas.height / 2);
```



ChatGPT

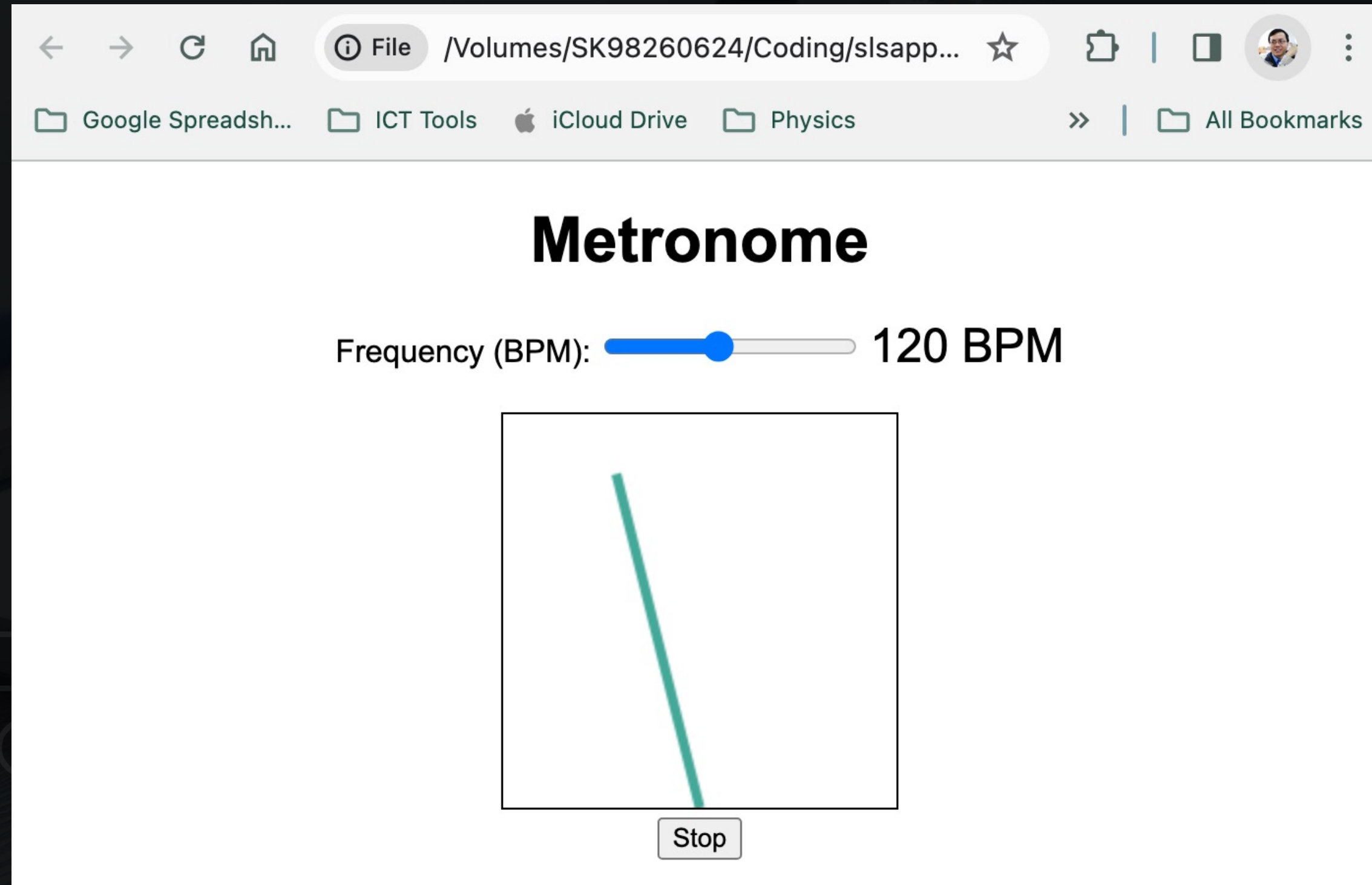
Certainly! I apologize for the oversight. Here's the corrected `createMolecule` function:

```
javascript Copy code  
  
function createMolecule() {  
  const x = Math.random() * canvas.width;  
  const y = canvas.height - Math.random() * (canvas.height / 2);  
  const energy = Math.random() * parseFloat(energySlider.value);  
  
  return {  
    x,  
    y,  
    energy,  
    color: randomColor(energy),  
    velocity: {  
      x: Math.random() * 2 - 1,  
      y: Math.random() * 2 - 1,  
    },  
  };  
}
```

Now, the water molecules will be created in the bottom half of the canvas as intended. Thank you for catching that mistake!



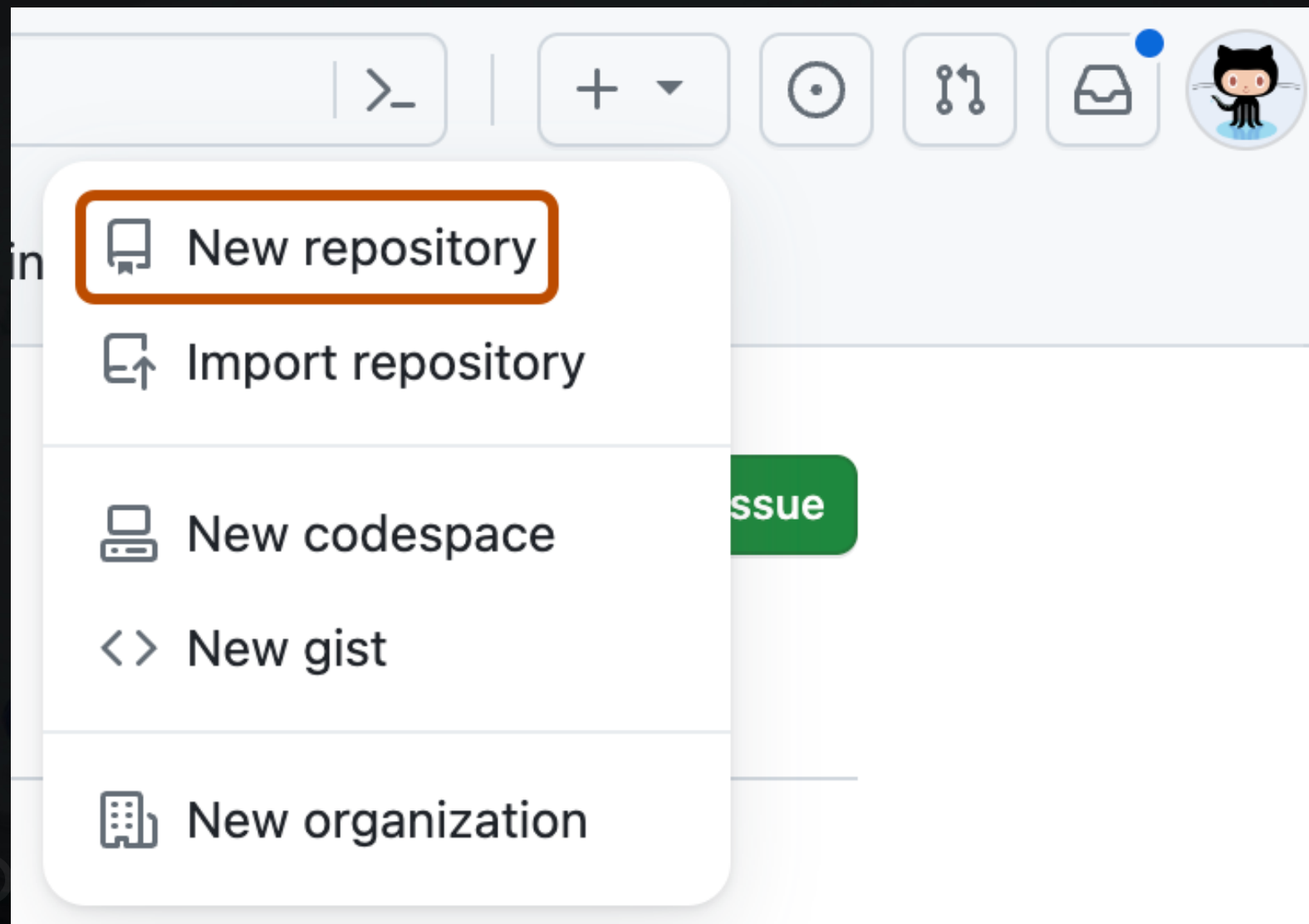
DEPLOYMENT OPTION 1: USE IT ON YOUR BROWSER



The screenshot shows a web browser window with the following elements:

- Address Bar:** Shows the file path `/Volumes/SK98260624/Coding/slsapp...`.
- Bookmarks:** Includes "Google Spreadsh...", "ICT Tools", "iCloud Drive", and "Physics".
- Page Title:** "Metronome".
- Control:** A slider for "Frequency (BPM)" is set to "120 BPM".
- Visual:** A green line representing a metronome pendulum is shown in a square frame.
- Button:** A "Stop" button is located below the pendulum frame.

DEPLOYMENT OPTION 2: UPLOAD INTO WEB



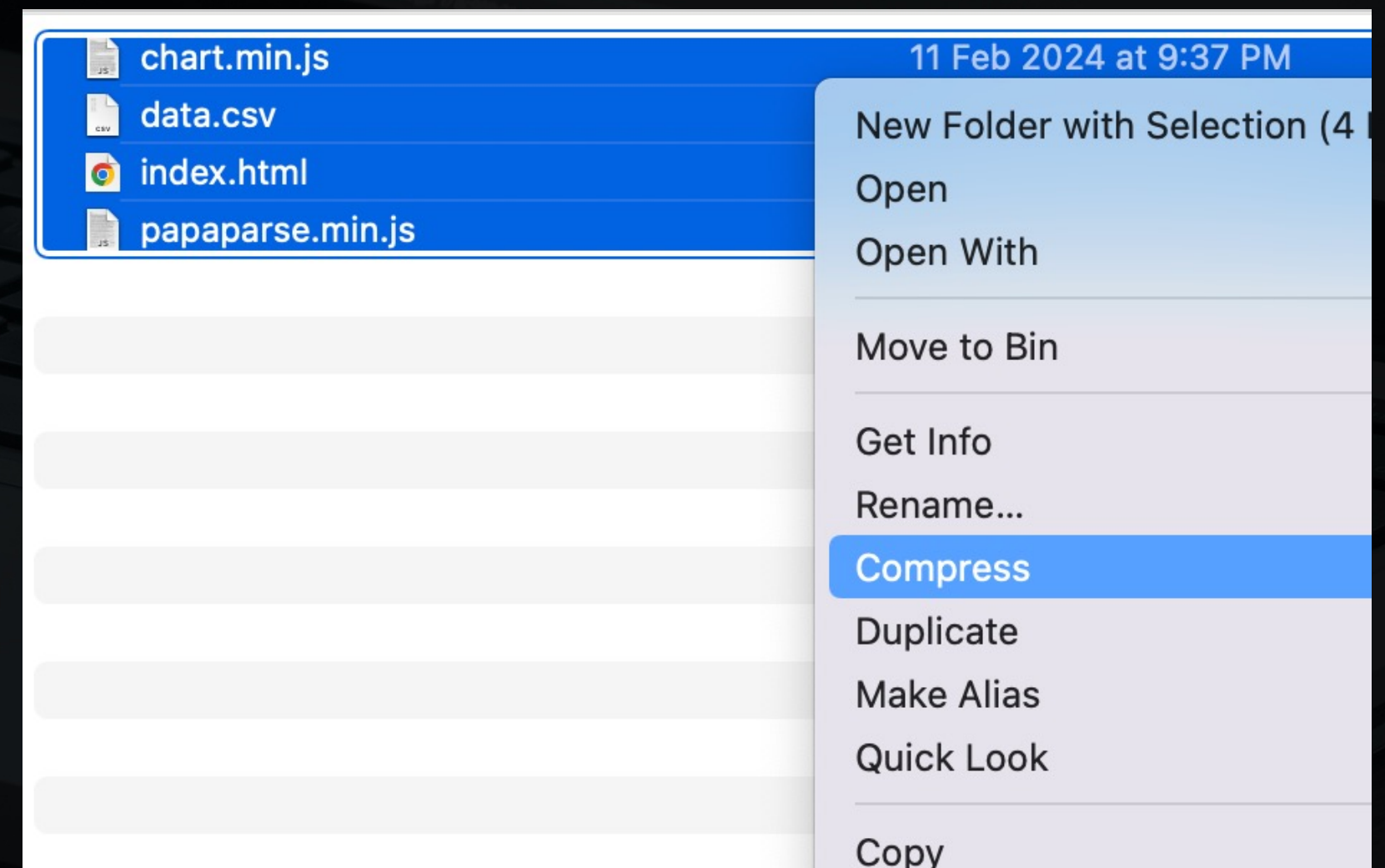
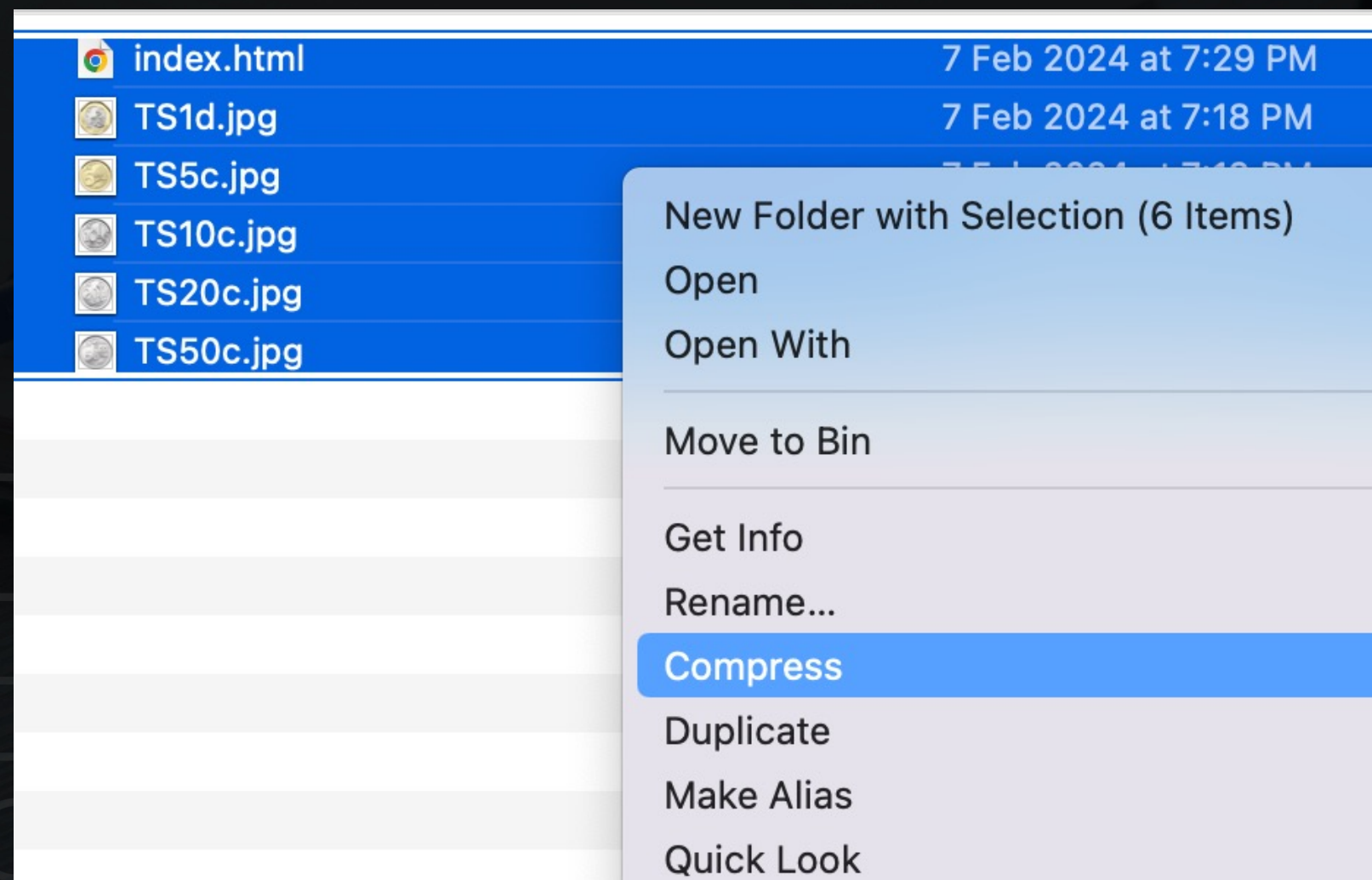
<https://docs.github.com/en/pages/getting-started-with-github-pages/creating-a-github-pages-site>

An advertisement for AWS Free Cloud Object Storage with Amazon S3. The background is dark blue with a faint grid pattern. The AWS logo is in the top left. The main heading is 'Free Cloud Object Storage with Amazon S3'. Below the heading is a paragraph of text: 'Looking for highly-scalable cloud object storage? Amazon S3 can help you with backup, archive, big data analytics, disaster recovery, and more.' Below this are three circular icons representing the free tier offer: '5GB FREE STANDARD STORAGE', '20K GET REQUESTS INCLUDED', and '2K PUT REQUESTS INCLUDED'. At the bottom right is an orange button that says 'Get Started For Free'.

<https://aws.amazon.com/free/storage/s3>

DEPLOYMENT OPTION 3: ZIP IT! FOR SLS

1. Name the main file index.html
2. Zip all the files in one folder



WORKFLOW

Prompt ChatGPT:
Divide the task into clear, sequential steps.

Op

Ref

In

Upload ZIP file to SLS: Add "File from Device"

The screenshot shows the SLS interface with a dark blue header containing 'SLS MY DRIVE' and 'Untitled / A. Untitled'. A central panel displays 'There is no content added'. Below this is a menu with categories: Text/Media, Question, Interactive, and Display. The 'File From Device' option is highlighted in the 'Text/Media' category. Other options include Split-Page Media, Text, Google File, YouTube Video, App, and Website. A '+ ADD NEW' button is also visible.

File:
s.html

in Web
Double-click
TML file.

File or
to web
(or Github)

LIMITATIONS OF STANDALONE APPS IN SLS

1. No connection to online databases. However, data can be stored in browser.
2. No use of external scripts outside of the zip folder.
3. No transfer of information between SLS and app.

GLOSSARY OF TERMS

- **Prompt:** The input provided to ChatGPT to generate a response
- **HTML5:** Markup language to create the structure and content of web pages
- **CSS:** Language that describes presentation and layout of HTML documents
- **Javascript:** Language that enables interactive behaviour in web pages

- **API:** Application Programming Interface, a set of protocols allowing different software applications to communicate with each other
- **Database:** Structured data stored electronically
- **Canvas:** Surface allowing dynamic rendering of graphics and animations

- **Interactive Elements**
 - **Buttons:** when clicked or pressed, trigger a function
 - **Radio buttons/Checkboxes:** used to select or deselect options independently
 - **Sliders:** used to adjust values within a specified range
 - **Input Boxes:** used to enter text

REFERENCES

1. Jensen Huang (2024, February 23). *Don't learn to code*. Vulcan Post. <https://vulcanpost.com/853029/dont-learn-to-code-jensen-huang-on-career/>
2. Mediacorp (2024, March 4). *Hello Singapore* [Video] <https://www.mewatch.sg/watch/Mar-2024-Hello-Singapore-443072>
3. Kasparov, G. (2017). *Deep thinking: where machine intelligence ends and human creativity begins*. Hachette UK.
4. Bull, C., & Kharrufa, A. (2023). *Generative AI Assistants in Software Development Education*.



THANK YOU

QUESTIONS & ANSWERS



THANK YOU

If you're interested in more quality content on e-Pedagogy...

Sign up on SgLDC to access even more materials by teachers, for teachers: <https://go.gov.sg/sgldc>

Sharing is caring! Grab a friend and tell them all about what you've learned today!

Feedback



Please provide some **feedback** for the session that you have just attended. You may access the form via the QR code above or this link:

<https://go.gov.sg/24vm1feedback>

