# Report: Gaze Heatmap and Statistics

## Objective

In this project, we are analyzing gaze data collected from eye-tracking software. Our aim is to visualize where the user was looking on a webpage and also extract useful statistics like which words or elements they focused on the most.

---

## What We Are Doing in the Code

### 1. Loading the Gaze Data

- In the preload() function, we are loading a .csv file (d2.csv) which contains gaze tracking data such as:
  - FPOGX and FPOGY → X and Y positions of gaze (from 0 to 1).
  - FPOGV → Gaze validity.
  - FPOGD → Duration of fixation.
- In setup(), we are creating a full-screen canvas and initializing UI elements like buttons and sliders to control the visualization.

---

### 2. Drawing the Heatmap

- In this code, we are mapping the normalized X and Y gaze values to actual pixel positions on the screen.
- We are storing these gaze points in an array heatmapData with details like:
  - Gaze position (x, y)
  - Duration of gaze
  - Size of the point
- In the draw() function, we are drawing semi-transparent red circles on the screen to create a heatmap effect where the user looked.

---

### 3. Detecting Gaze Words

- Here, we are trying to find which **words** the user was looking at.
- For this, we are using caretPositionFromPoint() to get the word under the gaze point.
- We are storing each detected word in a dictionary called detectedWords along with:
  - How many times it was looked at (fixation count)
  - How long it was looked at (total duration)
- When we click the “Gaze Words” button, we display this data in a resizable, draggable popup where font size and color are based on how much the user looked at each word.

---

**4. Tracking Element Types**

- We are also tracking **what type of element** the user looked at — whether it was a heading, paragraph, image, link, etc.
- To do this, we are using document.elementFromPoint() to detect the HTML element at the gaze point.
- We then use getElementType() to classify the element into:
  - Heading
  - Paragraph
  - Image
  - List
  - Link
  - Button
  - Other

---

**5. Showing Gaze Statistics**

- We created a “Show Statistics” button.
- When we click it, it shows a **popup with summarized data**:
  - Total fixations

    - Time spent on each element type
    - Percentage share of attention for each type
- This helps us understand what kind of content attracted the most attention.

---

**6. User Interface Controls**

- We made the buttons (Gaze Words, Show Statistics) fixed on the screen.
- The popup windows are **draggable** and **resizable**, which makes it easier for the user to move or resize the panels.
- We used CSS and JS functions to add this interactivity.