**NULLCLASS INTERNSHIP REPORT**

**(DATA ANALYTICS)**

**Introduction**

In today’s data-driven world, the ability to analyze and visualize data effectively is paramount to understanding trends, making informed decisions, and driving innovation. This project aimed to solve three real-world business problems using Tableau, a leading data visualization tool. The tasks spanned diverse domains, including social media analytics, job market insights, and role-specific hiring trends. By leveraging advanced filtering techniques, time-based dashboard controls, and domain-specific metrics, this project demonstrates the practical application of data visualization in decision-making processes. The report outlines the objectives, methodologies, and outcomes of each task, showcasing the value of actionable insights derived from data.

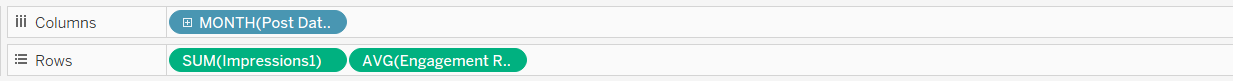
**Abstract**

This project focuses on solving three distinct analytical tasks using Tableau, emphasizing dynamic dashboards and interactive visualizations. The first task analyzes Twitter engagement rates and impressions to identify trends in social media performance within a specified timeframe. The second task delves into job market analytics, identifying top companies hiring for specific roles while applying complex filters, such as geographic exclusions, gender preferences, and educational qualifications. The third task examines job opportunities for mechanical engineers in Asia, factoring in company size, salary thresholds, and candidate experience. Each task incorporates time-based visibility controls to ensure focused data presentation during specified hours. The project highlights the versatility of Tableau in handling multi-dimensional datasets and delivering meaningful insights.

**Task 1: Twitter Engagement Analysis**

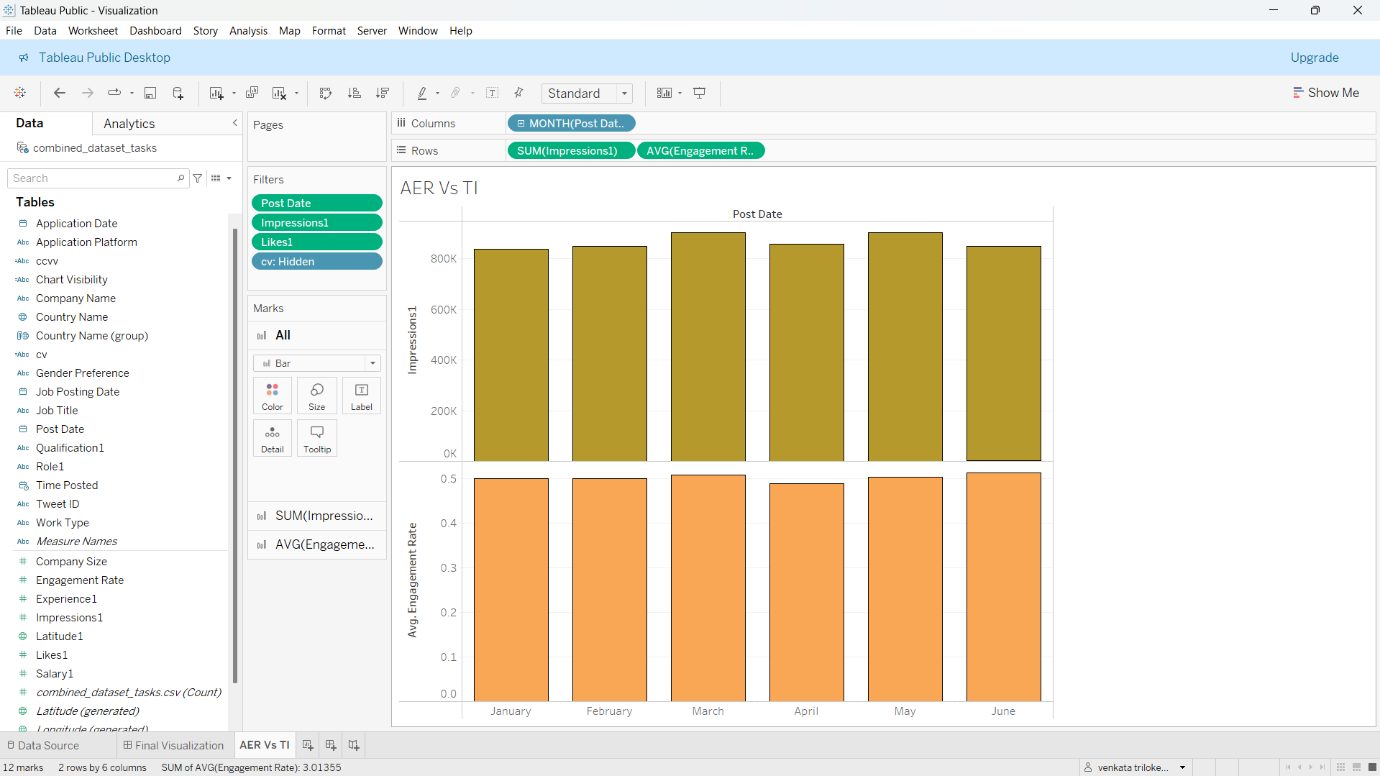
The task was to analyze tweets for the period between **January 1, 2020, and June 30, 2020**, and visualize the average engagement rate and total impressions. The specific conditions included:

1. Tweets must have received more than **100 impressions**.
2. Tweets with **0 likes** should be filtered out.
3. The visualization must only appear on the dashboard between **3 PM IST and 5 PM IST**.



**Steps Performed**:

1. **Data Import**: Imported the Twitter dataset into Tableau.
2. **Filtering Data**:
   * Applied a date range filter to include tweets between January 1, 2020, and June 30, 2020.
   * Added conditions to exclude tweets with impressions fewer than 100 and tweets with 0 likes.
3. **Engagement Rate Calculation**: Created a calculated field for Engagement Rate using the formula: Engagement Rate=(LikesImpressions)×100\text{Engagement Rate} = \left(\frac{\text{Likes}}{\text{Impressions}}\right) \times 100
4. **Visualization**:
   * Designed a bar chart where the x-axis represents tweet posting dates and the y-axis shows average engagement rates and total impressions.
   * Used color coding to differentiate high and low engagement rates.
5. **Time-Based Dashboard Control**:
   * Configured a time filter using Tableau's parameters to ensure the graph is visible only between 3 PM IST and 5 PM IST. Outside this timeframe, the graph is hidden.



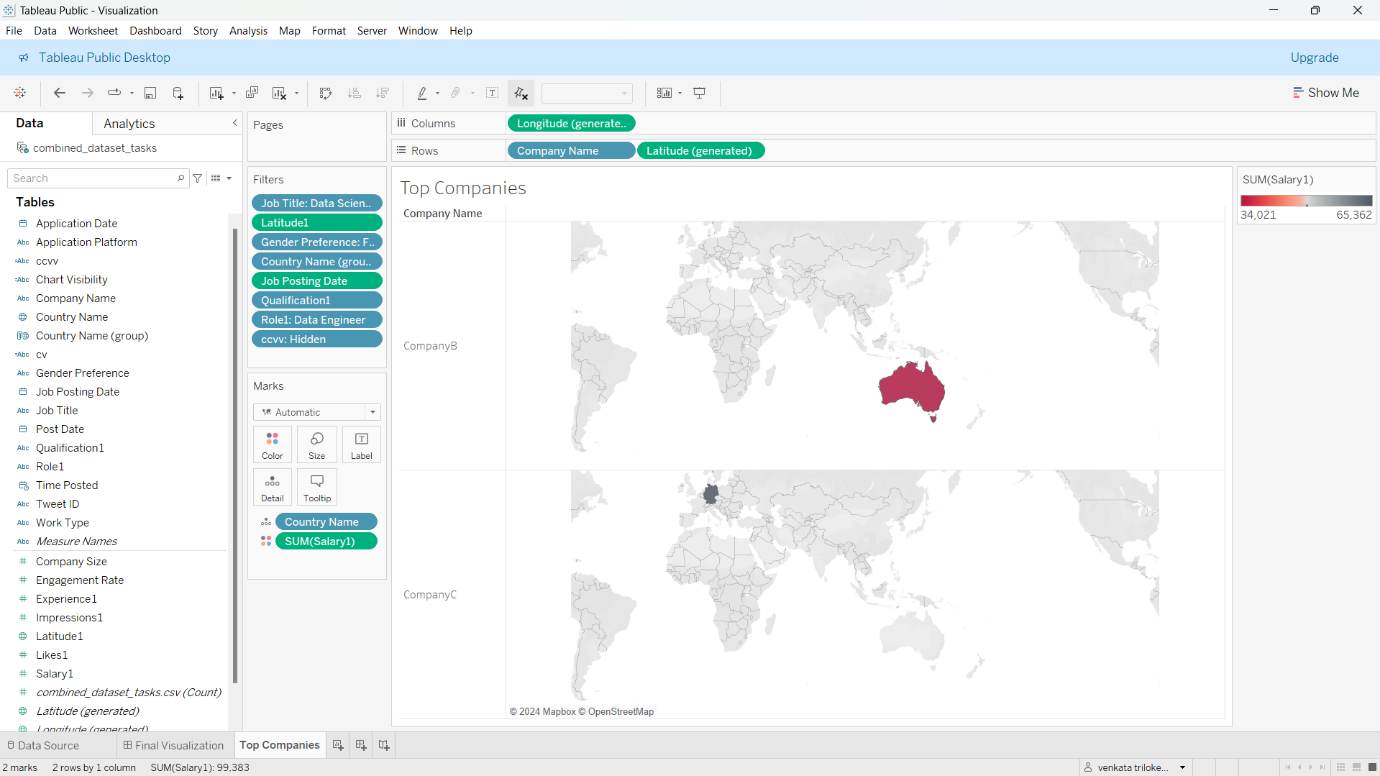
**Task 2: Top 10 Companies with Specific Roles**

To identify the **top 10 companies** with the most postings for the **Data Engineer** role and **Data Scientist** job title under specific conditions:

1. Exclude **Asian countries** and countries starting with the letter **‘C’** or with **latitude below 10**.
2. The job posting must be for **females only** and require a **B.Tech qualification**.
3. The job posting dates must be between **January 1, 2023, and June 1, 2023**.
4. Visualization must only be visible between **3 PM IST and 5 PM IST**.

**Steps Performed**:

1. **Data Import**: Loaded the job postings dataset into Tableau.
2. **Filtering Data**:
   * Excluded Asian countries.
   * Added filters to remove countries starting with 'C' and countries with latitude below 10.
   * Applied gender filter for females and qualification filter for B.Tech.
   * Restricted the date range to January 1, 2023, to June 1, 2023.
3. **Grouping and Aggregation**:
   * Grouped data by company name.
   * Calculated the count of job postings for the given roles and titles.



1. **Visualization**:
   * Designed a horizontal bar chart displaying the top 10 companies by job postings count.
2. **Time-Based Dashboard Control**:
   * Set up a time-based filter to ensure the chart is displayed only between 3 PM IST and 5 PM IST.

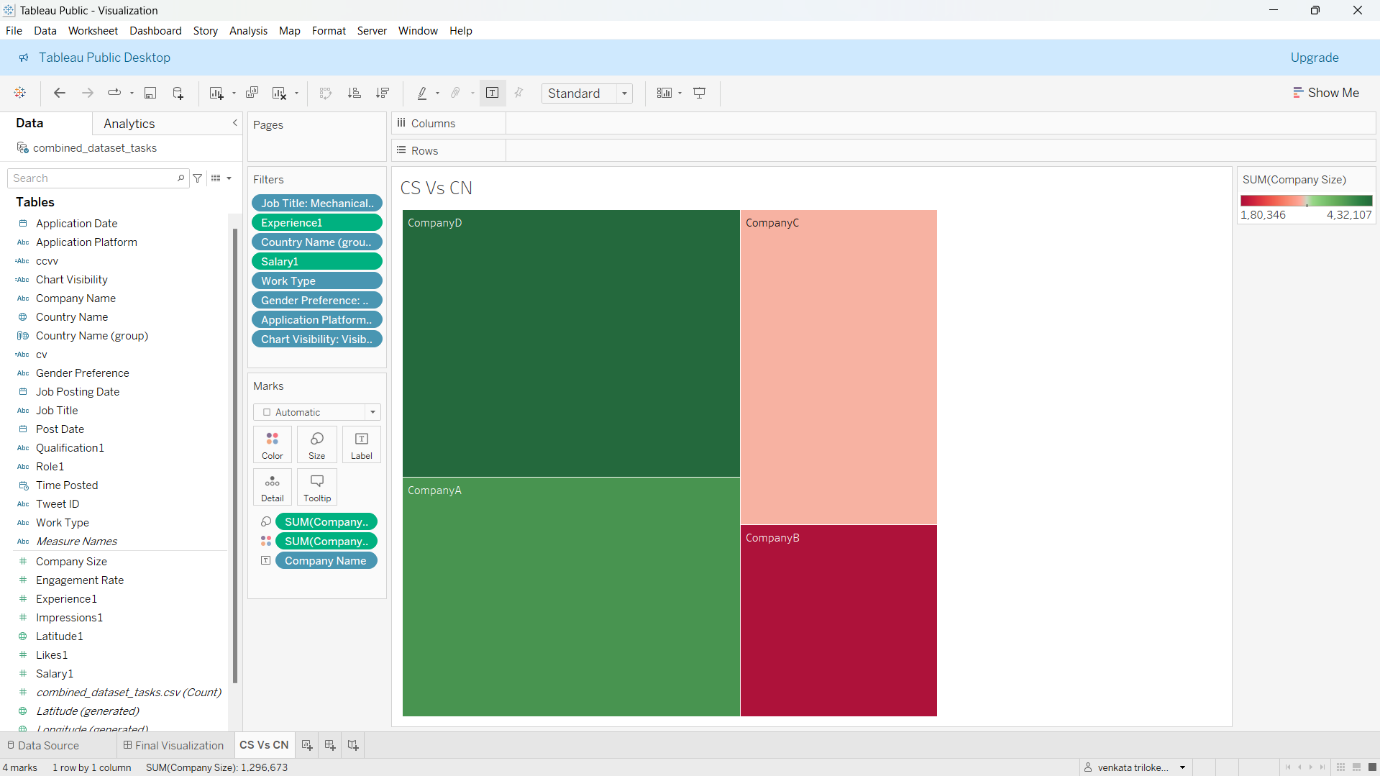
**Task 3: Mechanical Engineering Job Analysis**

To analyze **mechanical engineering job postings** based on the following conditions:

1. The **company size** should be less than **50,000 employees**.
2. The job title should be **Mechanical Engineer** with an **experience requirement of over 5 years**.
3. The postings must come from **Asian countries** offering a **salary above $50,000**.
4. The work type must be either **part-time or full-time** (other work types ignored).
5. Gender preference is **male**, and candidates must have applied via **Idealist**.
6. Visualization must only be visible between **3 PM IST and 5 PM IST**.

**Steps Performed**:

1. **Data Import**: Loaded the dataset into Tableau.
2. **Filtering Data**:
   * Applied filters for company size (< 50,000) and experience (> 5 years).
   * Included only Asian countries and salaries exceeding $50,000.
   * Filtered for work type to include only part-time and full-time jobs.
   * Added conditions to include male candidates who applied via Idealist.
3. **Visualization**:
   * Created a scatter plot with company size on the x-axis and salary on the y-axis.
   * Used different markers to highlight part-time and full-time work types.
4. **Time-Based Dashboard Control**:
   * Configured the dashboard to hide the chart outside the 3 PM IST to 5 PM IST timeframe.

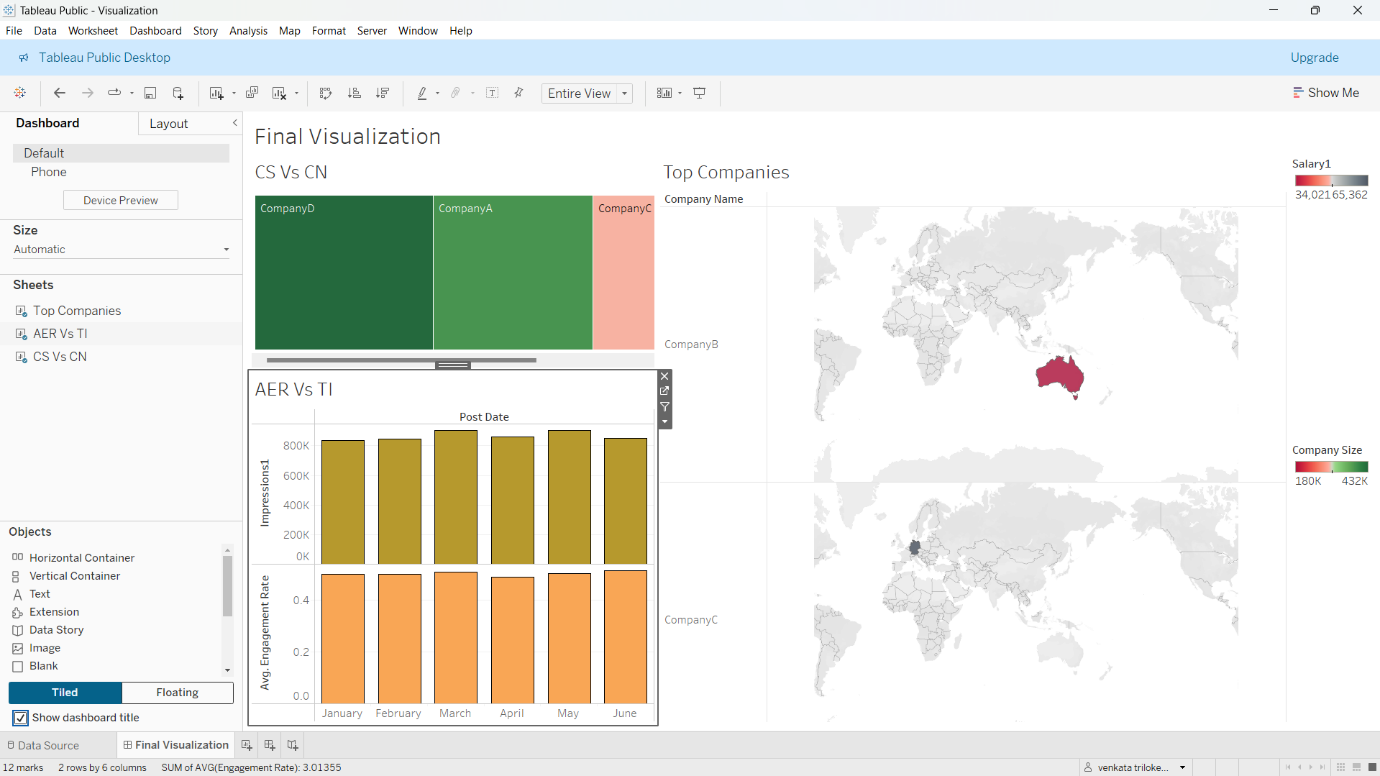


**Task 4: Comprehensive Dashboard Creation**

The fourth task involved integrating the visualizations developed in Tasks 1, 2, and 3 into a single, cohesive dashboard. The goal was to create an interactive and user-friendly interface that consolidates insights from the three tasks while adhering to the specific visibility conditions of each chart. This dashboard is designed to streamline data exploration and decision-making processes.

**Steps Performed**:

1. **Combining Visualizations**:
   * Imported the visualizations from Tasks 1, 2, and 3 into Tableau’s dashboard creation module.
   * Ensured that each chart retained its individual configurations, including filters, calculated fields, and dynamic conditions.
2. **Layout Design**:
   * Used a **grid-based layout** to organize the charts in a logical and visually appealing manner:
     + Chart 1 (Task 1) was placed at the top for quick access to Twitter engagement insights.
     + Chart 2 (Task 2) was positioned on the left side to display job market trends for Data Engineers and Data Scientists.
     + Chart 3 (Task 3) occupied the right side to present insights into mechanical engineering job opportunities.
   * Added titles and annotations for each chart to provide context and ensure clarity.
3. **Time-Based Visibility Control**:
   * Configured each chart to appear only during the **3 PM IST to 5 PM IST** timeframe by using Tableau’s parameters and filters.
   * Outside this time window, placeholder messages were added to inform users that the charts would be unavailable.
4. **Interactivity Enhancements**:
   * Introduced **filter actions** to allow users to interact with the charts dynamically. For example:
     + Clicking on a specific date range in the Task 1 chart filters the engagement data further.
     + Selecting a company in Task 2 updates the job posting details.
   * Linked filters across charts to allow for cohesive data exploration.
5. **Dashboard Testing**:
   * Conducted rigorous testing to ensure proper functioning of the time-based visibility settings.
   * Verified that interactions between charts were seamless and that all filters operated as expected.



**Task 5: Creating webpage**

**Steps:**

1. **Publish the Dashboard**:
   * Save and publish the dashboard to Tableau Public or Tableau Server.
   * After publishing, obtain the embed code. You can find this under the **"Share"** option on Tableau Public or Tableau Server.
2. **Embed the Code in HTML**:
   * Use the embed code provided by Tableau and include it in an HTML file.

Here’s the HTML code:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <div class='tableauPlaceholder' id='viz1735039573898' style='position: relative'><noscript><a href='#'><img alt='Final Visualization ' src='https:&#47;&#47;public.tableau.com&#47;static&#47;images&#47;Vi&#47;Visualization\_17350359259670&#47;FinalVisualization&#47;1\_rss.png' style='border: none' /></a></noscript><object class='tableauViz'  style='display:none;'><param name='host\_url' value='https%3A%2F%2Fpublic.tableau.com%2F' /> <param name='embed\_code\_version' value='3' /> <param name='site\_root' value='' /><param name='name' value='Visualization\_17350359259670&#47;FinalVisualization' /><param name='tabs' value='no' /><param name='toolbar' value='yes' /><param name='static\_image' value='https:&#47;&#47;public.tableau.com&#47;static&#47;images&#47;Vi&#47;Visualization\_17350359259670&#47;FinalVisualization&#47;1.png' /> <param name='animate\_transition' value='yes' /><param name='display\_static\_image' value='yes' /><param name='display\_spinner' value='yes' /><param name='display\_overlay' value='yes' /><param name='display\_count' value='yes' /><param name='language' value='en-US' /><param name='filter' value='publish=yes' /></object></div>                <script type='text/javascript'>                    var divElement = document.getElementById('viz1735039573898');                    var vizElement = divElement.getElementsByTagName('object')[0];                    if ( divElement.offsetWidth > 800 ) { vizElement.style.width='100%';vizElement.style.height=(divElement.offsetWidth\*0.75)+'px';} else if ( divElement.offsetWidth > 500 ) { vizElement.style.width='100%';vizElement.style.height=(divElement.offsetWidth\*0.75)+'px';} else { vizElement.style.width='100%';vizElement.style.height='1127px';}                     var scriptElement = document.createElement('script');                    scriptElement.src = 'https://public.tableau.com/javascripts/api/viz\_v1.js';                    vizElement.parentNode.insertBefore(scriptElement, vizElement);                </script>

</body>

</html>

**Instructions for the Code:**

1. Replace the src attribute value (https://public.tableau.com/views/YourDashboardName/SheetName?embed=true) with the embed URL of your Tableau dashboard.
2. Save the file with a .html extension (e.g., dashboard.html).
3. Open the HTML file in a browser to view the embedded Tableau dashboard.

**Task 6: To publish the HTML file on Netlify using a GitHub repository, follow these steps**

**Step 1: Prepare Your Repository**

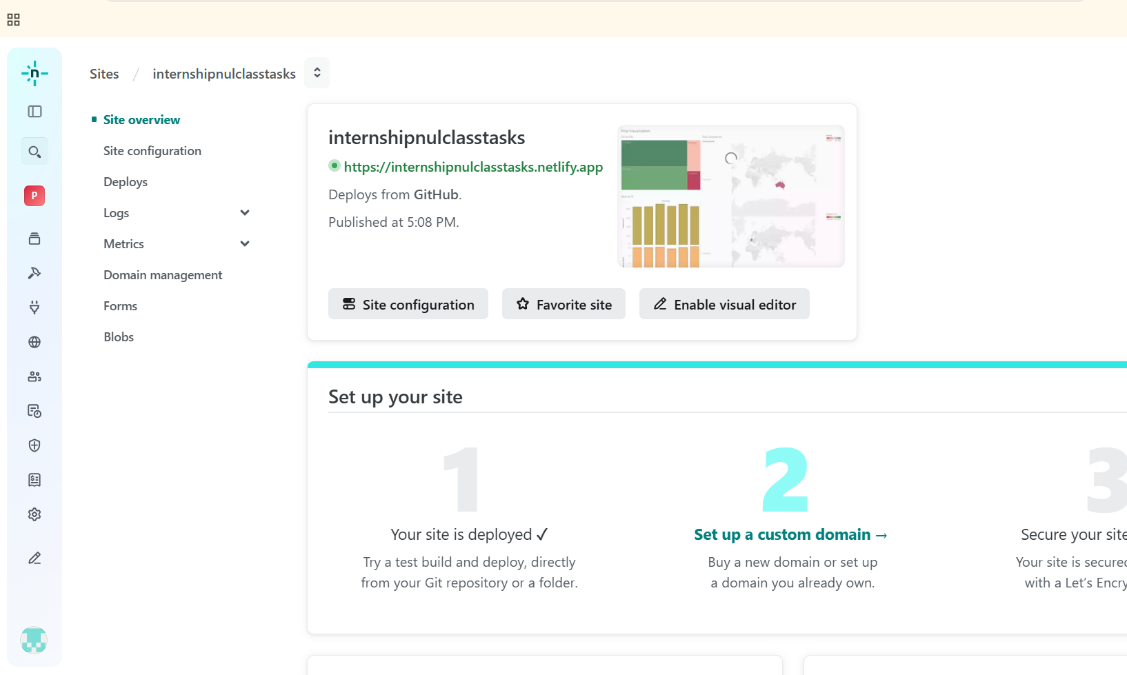
1. **Create a GitHub Repository**:
   * Go to [GitHub](https://github.com/) and create a new repository (e.g., Tableau-Dashboard).
   * Clone the repository to your local machine using Git or directly in your IDE (e.g., VS Code).
2. **Add the HTML File**:
   * Place the HTML file (e.g., dashboard.html) into the repository folder.
   * Add a README.md file to document the project (optional but recommended).
3. **Push the Files to GitHub**: Run the following commands in your terminal:
4. git add .
5. git commit -m "Initial commit with HTML file for Tableau Dashboard"
6. git branch -M main
7. git push -u origin main

**Step 2: Set Up Netlify Deployment**

1. **Log In to Netlify**:
   * Go to [Netlify](https://www.netlify.com/) and log in with your account. If you don’t have an account, create one.
2. **Create a New Site**:
   * Click on **"Add New Site"** and select **"Import from GitHub"**.
   * Connect your GitHub account to Netlify if not already done.
3. **Select Your Repository**:
   * Search for and select the repository containing your HTML file.
4. **Configure Build Settings**:
   * In the **Basic Build Settings**, set:
     + **Build command**: Leave this blank (no build required for static HTML).
     + **Publish directory**: Leave this blank or set to the root folder.
5. **Deploy the Site**:
   * Click **"Deploy Site"** and wait for Netlify to process the deployment.
   * Once the deployment is complete, Netlify will provide a unique URL for your site (link to my webpage: [**https://internshipnulclasstasks.netlify.app**](https://internshipnulclasstasks.netlify.app)).

**Step 3: Test the Deployed Dashboard**

1. Visit the provided Netlify URL.
2. Verify that the Tableau dashboard is loading correctly.
3. If needed, update the embed URL or make other modifications and push them to GitHub. Netlify will automatically redeploy changes.



**Conclusion:**

This project successfully demonstrates the power of data visualization in solving real-world challenges across multiple domains. By leveraging Tableau, we effectively transformed raw datasets into actionable insights, enabling users to make informed decisions. The tasks showcase the importance of dynamic filters, calculated fields, and time-based dashboards in creating focused and interactive visualizations. Through these analyses, we gained deeper insights into social media performance, hiring trends, and job market dynamics. This project underscores the significance of combining technical expertise with visualization tools to extract value from data, paving the way for more informed and strategic decision-making in business environments.