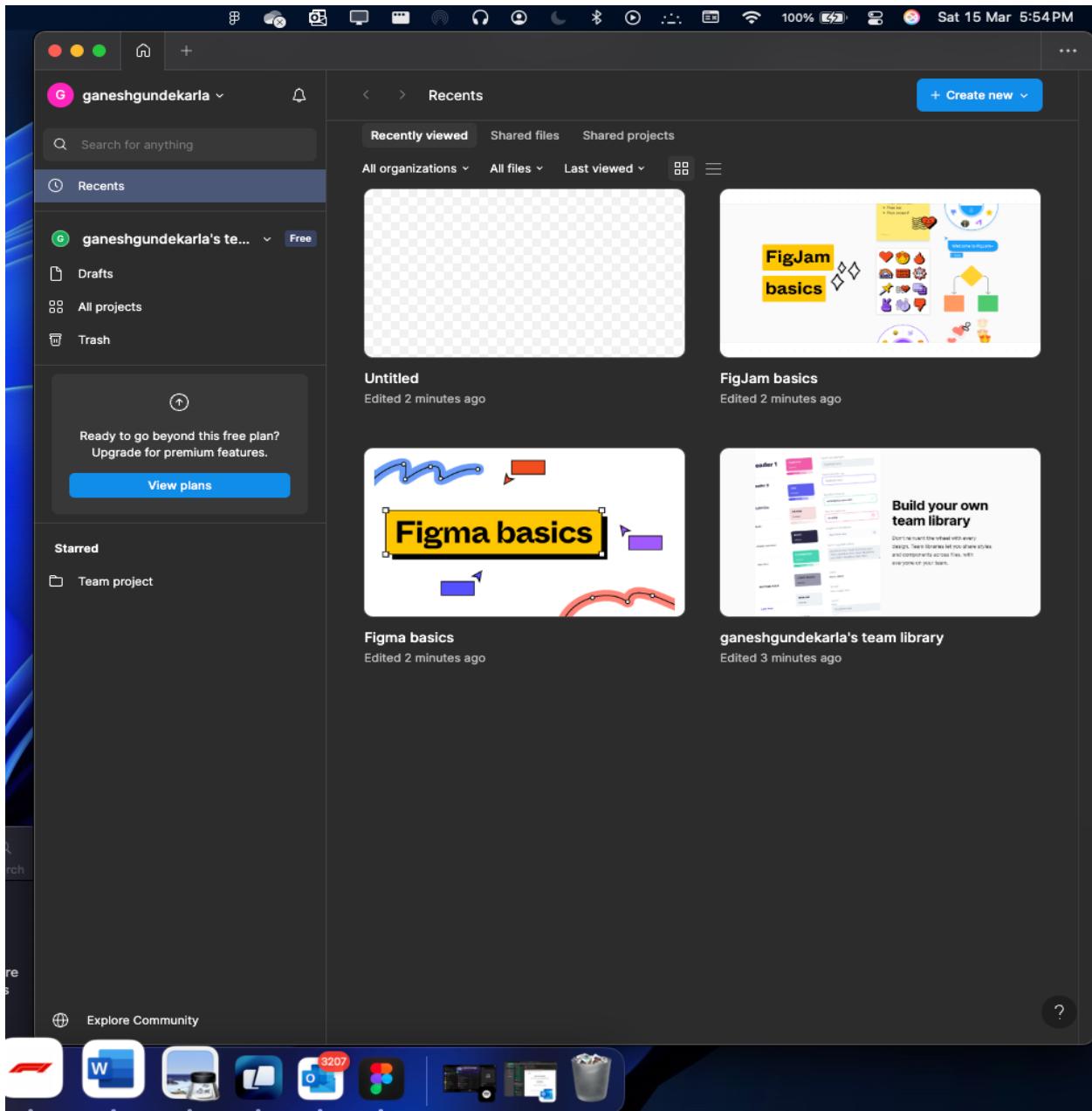


Data Visualization Activity 5

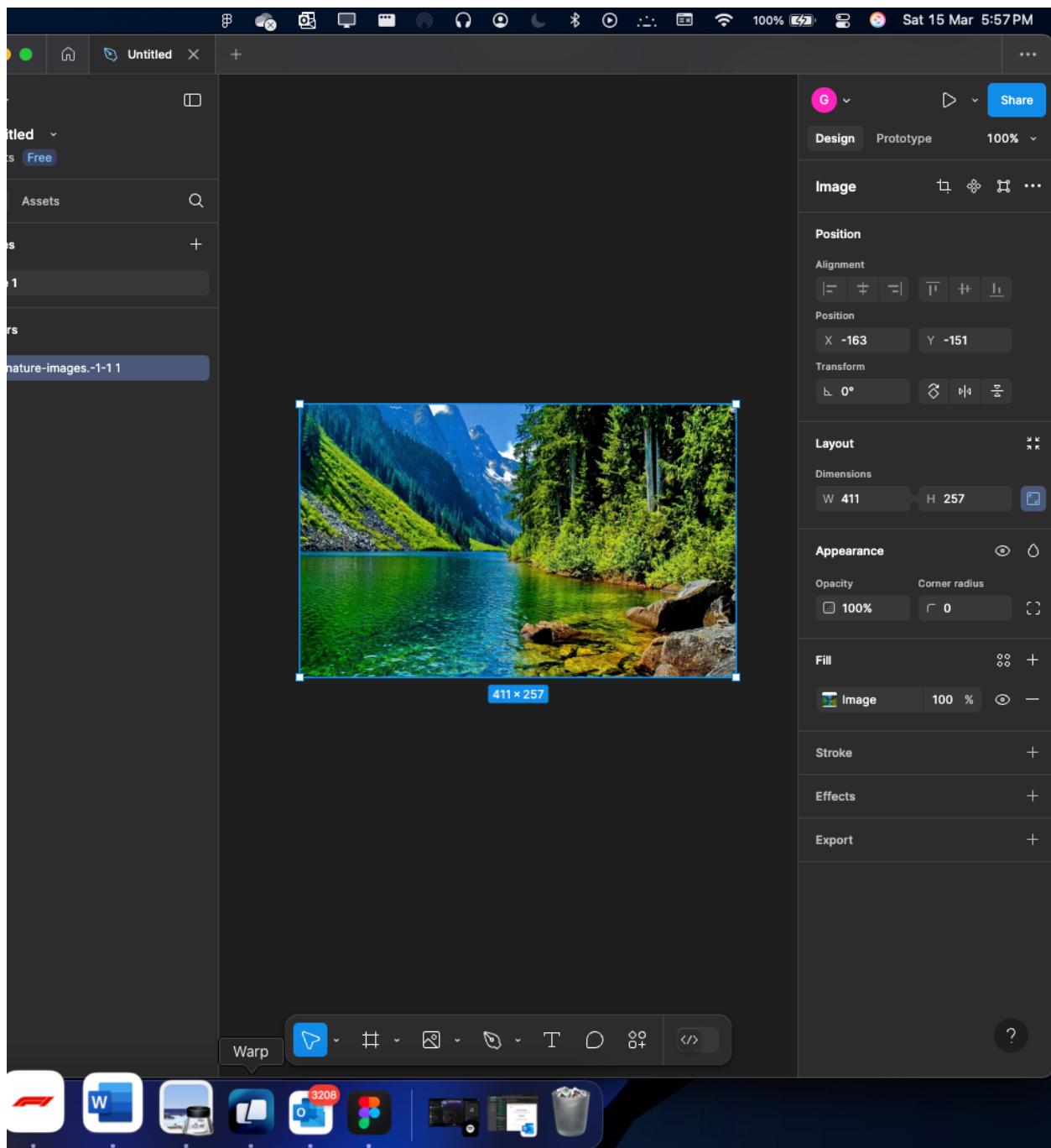
Ganesh Gundekarla(11700551)

Tutorial 1 : intro to the Support Vector Graphics and CSS

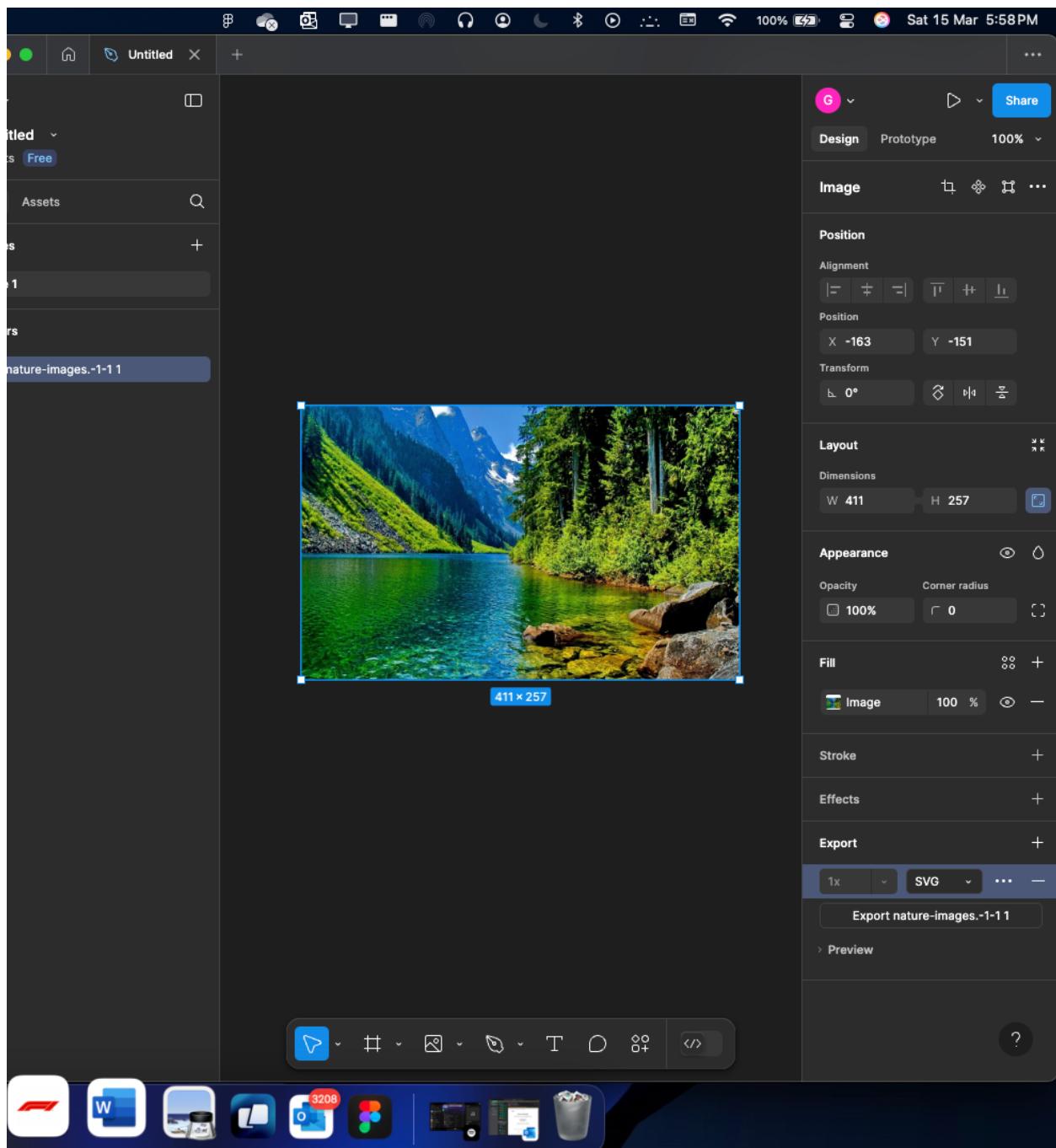
Loading up the figma



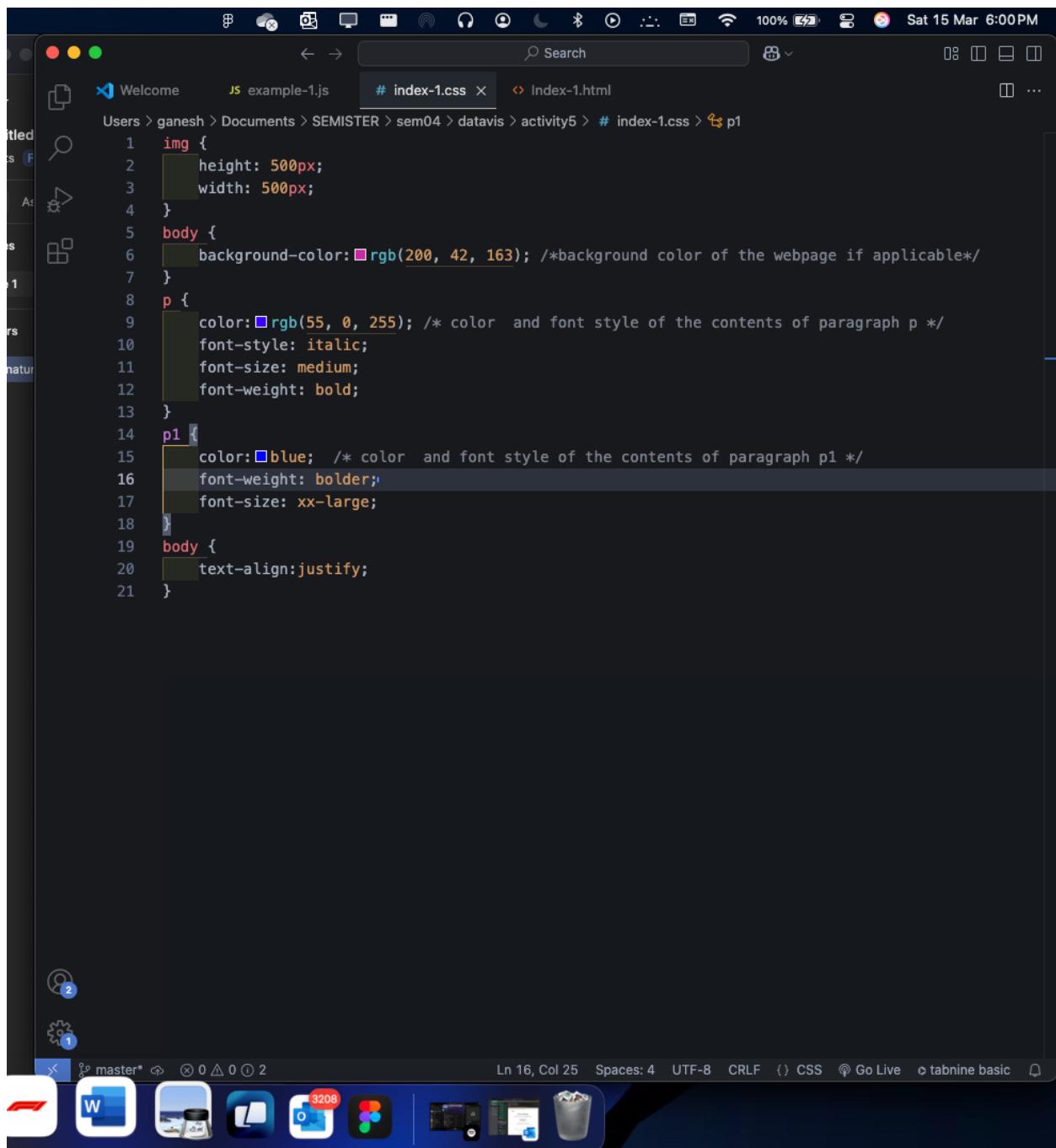
Step 2 : placing the downloaded image in the workspace.



Step 3 : clicking on the export and selecting the type of export as “ SVG ” .



Step 4 : creating our own web page.



```
1  img {  
2      height: 500px;  
3      width: 500px;  
4  }  
5  body {  
6      background-color:#rgb(200, 42, 163); /*background color of the webpage if applicable*/  
7  }  
8  p {  
9      color:#rgb(55, 0, 255); /* color and font style of the contents of paragraph p */  
10     font-style: italic;  
11     font-size: medium;  
12     font-weight: bold;  
13  }  
14  p1 {  
15     color:#blue; /* color and font style of the contents of paragraph p1 */  
16     font-weight: bolder;  
17     font-size: xx-large;  
18  }  
19  body {  
20      text-align:justify;  
21  }
```

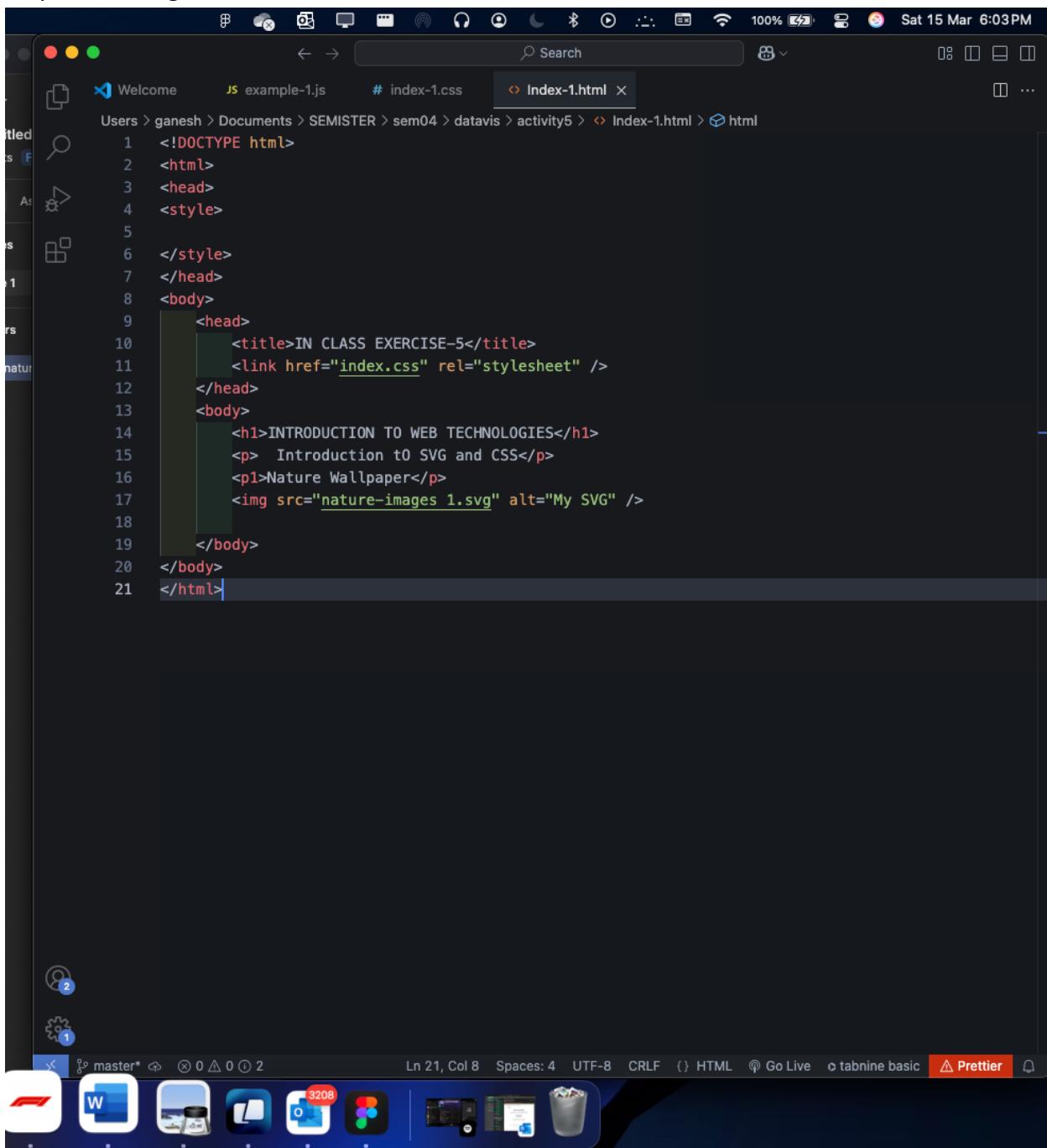
Step 5 : manipulating the colors of the webpage .

The screenshot shows the Adobe Dreamweaver interface with a dark theme. The top menu bar includes icons for file operations, search, and system status (battery at 100%, network, volume). The title bar shows the current file path: "Users > ganesh > Documents > SEMISTER > sem04 > datavis > activity5 > # index-1.css > p1". The main workspace displays the following CSS code:

```
1  img {  
2      height: 500px;  
3      width: 500px;  
4  }  
5  body {  
6      background-color: #rgb(200, 42, 163); /*background color of the webpage if applicable*/  
7  }  
8  p {  
9      color: #rgb(55, 0, 255); /* color and font style of the contents of paragraph p */  
10     font-style: italic;  
11     font-size: medium;  
12     font-weight: bold;  
13  }  
14  p1 {  
15     color: #rgb(155, 50, 121); /* color and font style of the contents of paragraph p1 */  
16     font-weight: bolder;  
17     font-size: xx-large;  
18  }  
19  body {  
20     text-align:justify;  
21  }
```

The bottom status bar indicates "Ln 15, Col 28 (24 selected)" and "Spaces: 4". It also shows icons for Go Live, tabnine basic, and a refresh button. A dock at the bottom contains icons for various Adobe applications like Photoshop, Illustrator, and InDesign.

Step 6 : working with the index.html file.



The screenshot shows a dark-themed code editor window. The title bar includes standard OS X icons and the date/time: "Sat 15 Mar 6:03PM". The main area displays the content of the "Index-1.html" file. The file structure is as follows:

```
<!DOCTYPE html>
<html>
<head>
<style>
</style>
</head>
<body>
<head>
<title>IN CLASS EXERCISE-5</title>
<link href="index.css" rel="stylesheet" />
</head>
<body>
<h1>INTRODUCTION TO WEB TECHNOLOGIES</h1>
<p> Introduction to SVG and CSS</p>
<p>Nature Wallpaper</p>

</body>
</body>
</html>
```

The code editor has a sidebar on the left with icons for file operations like Open, Save, Find, and Copy/Paste. The bottom status bar shows "Ln 21, Col 8" and other file-related information. A dock at the bottom contains icons for various applications, including Figma, which has a red notification badge with the number "3208".

Step 7 : embedding the svg image which we edited in the figma.

```
<!DOCTYPE html>
<html>
<head>
    <title>IN CLASS EXERCISE-5</title>
    <link href="index.css" rel="stylesheet" />
</head>
<body>
    <h1>INTRODUCTION TO WEB TECHNOLOGIES</h1>
    <p> Introduction to SVG and CSS</p>
    <p>Nature Wallpaper</p>
    
</body>
</html>
```

Step 8 : running the html and css and seeing the result in the safari.

The screenshot shows a Mac desktop environment. In the center is a code editor window titled "Index-1.html" with the file path "# index-1.css". The code editor displays the following HTML and CSS:

```
<!DOCTYPE html>
<html>
<head>
<title>IN CLASS EXERCISE-5</title>
<link href="index-1.css" rel="stylesheet" />
</head>
<body>
<h1>INTRODUCTION TO WEB TECHNOLOGIES</h1>
<p> Introduction to SVG and CSS</p>
<p>Nature Wallpaper</p>

</body>
</html>
```

To the left of the code editor is a browser window titled "Personal" showing a pink background with the text "INTRODUCTION TO WEB TECHNOLOGIES" and "Nature Wallpaper". Below the browser is a preview of a nature scene with green hills and a blue lake. The desktop dock at the bottom contains icons for various applications like Finder, Mail, and Safari.

Task 2 :

Intro to JavaScript.

Step 1 : generating a alert by inserting extra lines of code in html page.

```
<!DOCTYPE html>
<html>
<head>
<style>
</style>
</head>
<body>
<head>
<title>IN CLASS EXERCISE-5</title>
<link href="index-1.css" rel="stylesheet" />
</head>
<body>
<h1>INTRODUCTION TO WEB TECHNOLOGIES</h1>
<p> Introduction to SVG and CSS</p>
<p>Nature Wallpaper</p>

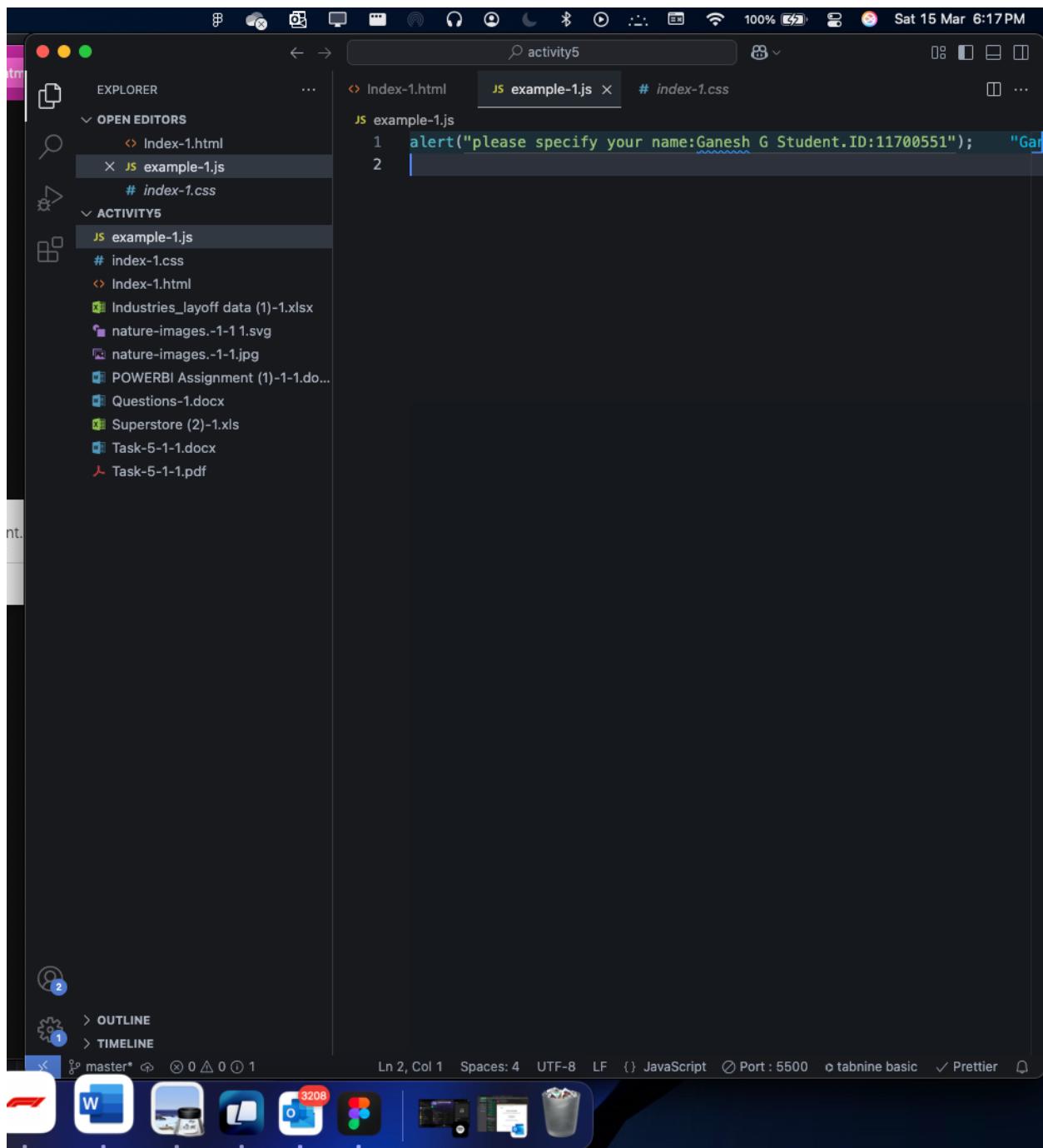
<script src="example-1.js"></script>
</body>
</body>
</html>
```

Step 2 : creating a script file with a js extension which includes the name and the student id ;

And also updating the relative path.

A screenshot of a dark-themed code editor interface. The top bar shows system icons and the date/time: "Sat 15 Mar 6:12PM". The title bar says "activity5". The left sidebar has sections for "EXPLORER", "OPEN EDITORS", and "ACTIVITY". In "OPEN EDITORS", there are three tabs: "Index-1.html" (3 unsaved), "JS example-1.js" (active), and "# index-1.css". In "ACTIVITY", there is a list of files including "JS example-1.js", "# index-1.css", "Index-1.html" (3), "Industries_layout data (1)-1.xlsx", "nature-images.-1-1.svg", "nature-images.-1-1.jpg", "POWERBI Assignment (1)-1-1.doc...", "Questions-1.docx", "Superstore (2)-1.xls", "Task-5-1-1.docx", and "Task-5-1-1.pdf". A modal window titled "Accounts - Sign in to Sync Settings - Sign in requested" is open at the bottom left. The bottom status bar shows "Ln 3, Col 1", "Spaces: 4", "UTF-8", "JavaScript", "Port : 5500", "tabnine basic", "Prettier", and a file count of "3208". The bottom dock contains various application icons.

```
JS example-1.js
1 alert("please specify your name:Ganesh G Student.ID:11700551");
2 <script src="example.js"></script>;
3
```



A screenshot of a Mac desktop environment. The top menu bar includes Apple, Code, File, Edit, Selection, View, Go, Run, Terminal, Window, and Help. A tab in the browser window shows the URL `127.0.0.1:5500/index-1.html`. The code editor on the right displays the file `index-1.html` with the following content:

```
<!DOCTYPE html>
<html> <body> <p>activity5</p>
</body></html>

<title>IN CLASS EXERCISE-5</title>
<link href="index-1.css" rel="stylesheet" />

<h1>INTRODUCTION TO WEB TECHNOLOGIES</h1>
<p> Introduction to SVG and CSS</p>

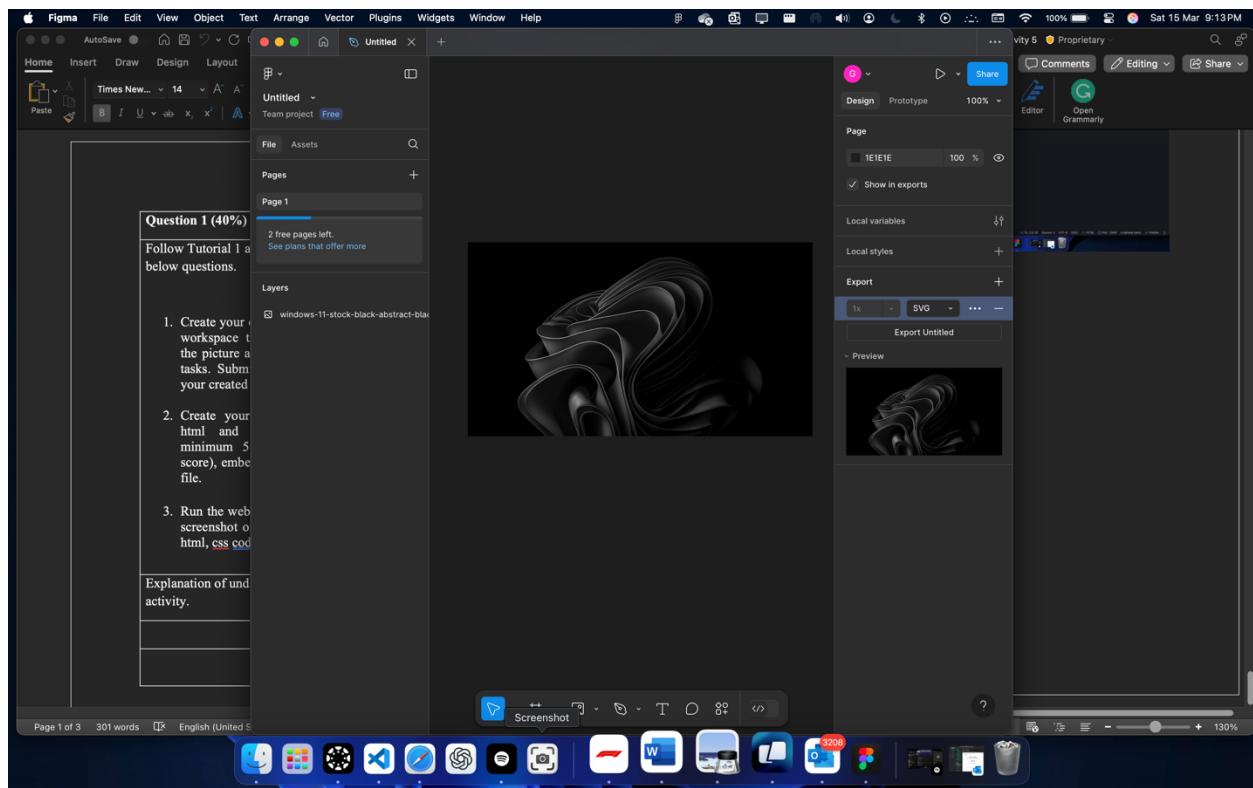
<script src="example-1.js"></script>
```

The left sidebar shows an Explorer view with files like `Index-1.html`, `example-1.js`, and `index-1.css`. Below the code editor is a toolbar with icons for Outline, Timeline, and other development tools. The bottom of the screen shows the Mac OS X dock with various application icons.

Questions for task 1 and task 2 :

Task 1 :

Creating own figma image.



2. own webpage.

Safari browser window showing a local file at `127.0.0.1:5500/index.html`. The page title is "Welcome to My Webpage". The content area is empty, displaying only the Windows 11 stock black abstract background image. The right side of the screen shows the browser's developer tools, specifically the "Elements" tab, displaying the HTML and CSS code for the page.

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>My Webpage with SVG</title>
    <link rel="stylesheet" href="styles.css" />
  </head>
  <body>
    <header>
      <h1>Welcome to My Webpage</h1>
    </header>
    <div class="container">
      <p>This is a simple webpage with nothing because I choose to show nothing.</p>
      <p>This is a static Windows wallpaper.</p>
      
    </div>
  </body>
</html>

```

Here , I understood how html and css work together in creating. A entire web page , I also found how js is used to include dynamic nature to the html pages , html provides us with a body where as css is used for styling and js is used to have the dynamic nature of the web page as html is a static in nature.

Question 2 :

please specify your course name , activity name and full name : Data visualization , activity 5 , Ganesh Gundekarla

[Close](#)

```
index.html > ...
1  <!DOCTYPE HTML File (index.html) ...
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width,
6          initial-scale=1.0" />
7      <title>My Webpage with SVG</title>
8      <link rel="stylesheet" href="styles.css" />
9    </head>
10   <body>
11     <header>
12       <h1>Welcome to My Webpage</h1>
13     </header>
14     <div class="container">
15       <p>
16         This is a simple webpage with nothing because I choose to
17         show nothing.
18       </p>
19       <p>This is a static Windows wallpaper.</p>
20       
24       <script src="ex.js"></script>
25     </div>
26   </body>
</html>
```

Understanding of the code : here , we added a alert button to the html page , well the page is still there but the integration of the js file will create a mandatory alert to the web page before accessing it. Here, this is basically a dynamic feature that binds with html and used to provide websites with interactivity.