1. HTML and CSS for equation

HTML CODE:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Wave Equation</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="equation">

<div class="fraction">

<span class="numerator">

<span>&#8706;<sup class="superscript">2</sup>u</span>

</span>

<span class="denominator">

<span>&#8706;t<sup class="superscript">2</sup></span>

</span>

</div>

<span> = </span>

<span>c<sup class="superscript">2</sup></span>

<div class="fraction">

<span class="numerator">

<span>&#8706;<sup class="superscript">2</sup>u</span>

</span>

<span class="denominator">

<span>&#8706;x<sup class="superscript">2</sup></span>

</span>

</div>

</div>

</body>

</html>

CSS CODE:

body {

font-family: 'Times New Roman', Times, serif;

font-size: 24px;

display: flex;

align-items: center;

justify-content: center;

height: 100vh;

margin: 0;

}

.equation {

display: flex;

align-items: center;

justify-content: center;

}

.fraction {

display: inline-block;

vertical-align: middle;

text-align: center;

}

.fraction > span {

display: block;

padding: 0 0.1em;

}

.fraction .numerator {

border-bottom: 1px solid #000;

}

.fraction .denominator {

margin-top: -0.5em;

}

.superscript {

font-size: 0.75em;

vertical-align: super;

}

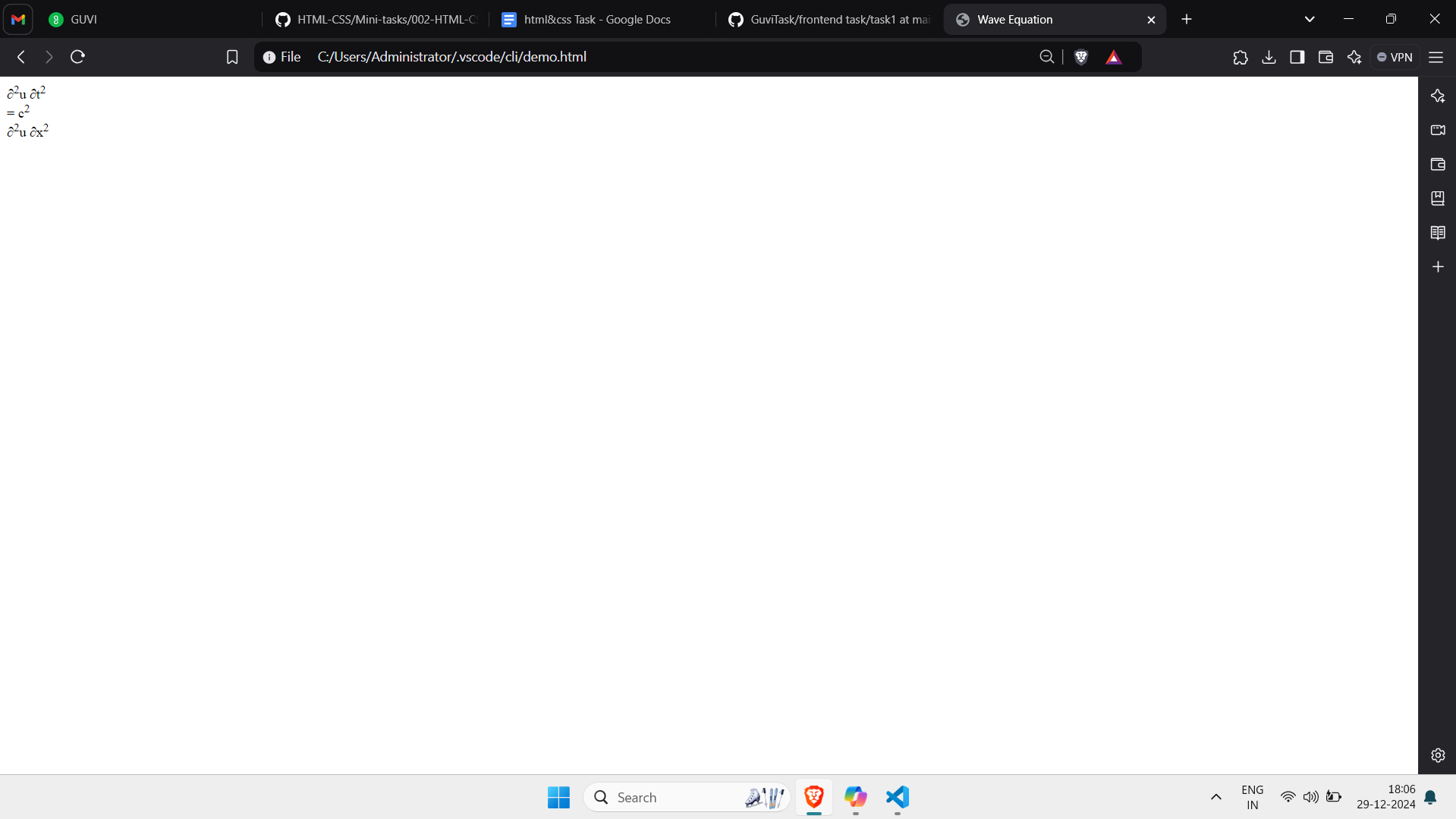
.subscript {

font-size: 0.75em;

vertical-align: sub;

}

OUTPUT:



2. Write a css rule of rounded rectangle, dashed border, grayscale for all image tags in your webpage.

img {

border: 2px dashed #000;

border-radius: 15px;

filter: grayscale(100%);

display: block;

max-width: 100%;

height: auto;

}

3. Write a media rule to change only the font size of web page when it's resized.

/\* Default font size \*/

body {

font-size: 16px;

}

/\* When the viewport width is 600px or less \*/

@media (max-width: 600px) {

body {

font-size: 14px;

}

}

/\* When the viewport width is 400px or less \*/

@media (max-width: 400px) {

body {

font-size: 12px;

}

}

4. Use HTML & CSS to display different types of cursors.

CODE.HTML

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Cursors Example</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<h1>Different Types of Cursors</h1>

<div class="cursor-default">Default Cursor</div>

<div class="cursor-pointer">Pointer Cursor</div>

<div class="cursor-crosshair">Crosshair Cursor</div>

<div class="cursor-move">Move Cursor</div>

<div class="cursor-text">Text Cursor</div>

<div class="cursor-wait">Wait Cursor</div>

<div class="cursor-help">Help Cursor</div>

</body>

</html>

CODE.CSS:

body {

font-family: Arial, sans-serif;

padding: 20px;

}

h1 {

margin-bottom: 20px;

}

div {

margin: 10px 0;

padding: 10px;

border: 1px solid #000;

}

/\* Default cursor \*/

.cursor-default {

cursor: default;

}

/\* Pointer cursor \*/

.cursor-pointer {

cursor: pointer;

}

/\* Crosshair cursor \*/

.cursor-crosshair {

cursor: crosshair;

}

/\* Move cursor \*/

.cursor-move {

cursor: move;

}

/\* Text cursor \*/

.cursor-text {

cursor: text;

}

/\* Wait cursor \*/

.cursor-wait {

cursor: wait;

}

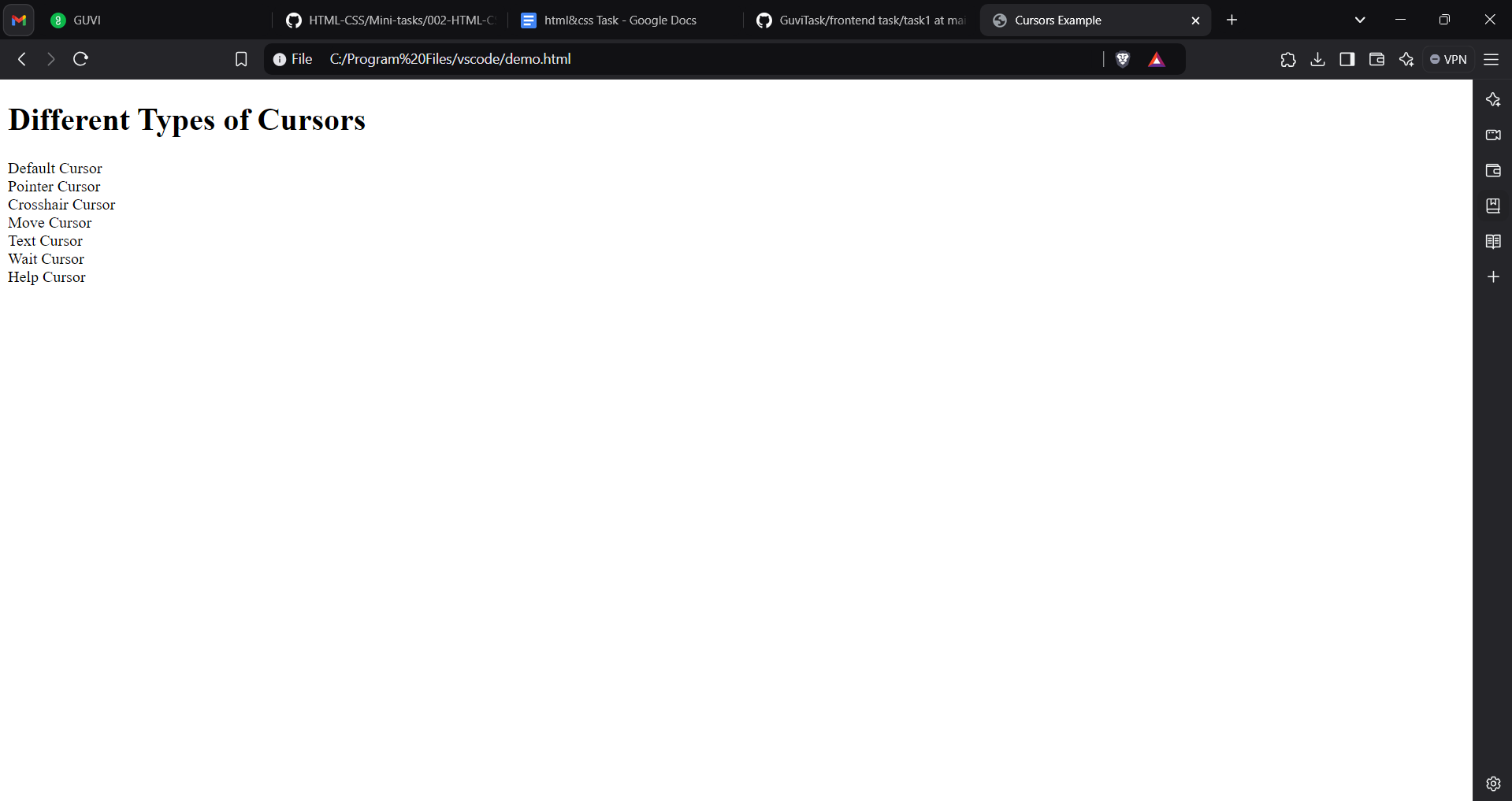
/\* Help cursor \*/

.cursor-help {

cursor: help;

}

OUTPUT:



5. Write a CSS class that meets the Box Model requirement for all the div tags in your page.

* + Use certain properties padding, margin, border.

CODE

/\* Box model for all div tags \*/

.box-model {

padding: 20px;

margin: 10px;

border: 2px solid #000;

box-sizing: border-box;

}

6. Write a CSS class to change size of font, color font family when mouse moves over of span tag in your webpage.

CODE.HTML:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Hover Effect Example</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<p>

Hover over this text to see the effect: <span>Hover over me!</span>

</p>

</body>

</html>

CODE.CSS:

/\* Default span styling \*/

span {

font-size: 16px;

color: black;

font-family: Arial, sans-serif;

}

/\* Hover effect for span \*/

span:hover {

font-size: 20px;

color: red;

font-family: 'Times New Roman', Times, serif;

}

7. Write a CSS 2D rule with translate() property to move an element.

HTML CODE:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Test Translate</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="moving-element"></div>

</body>

</html>

CSS CODE:

.moving-element {

width: 100px;

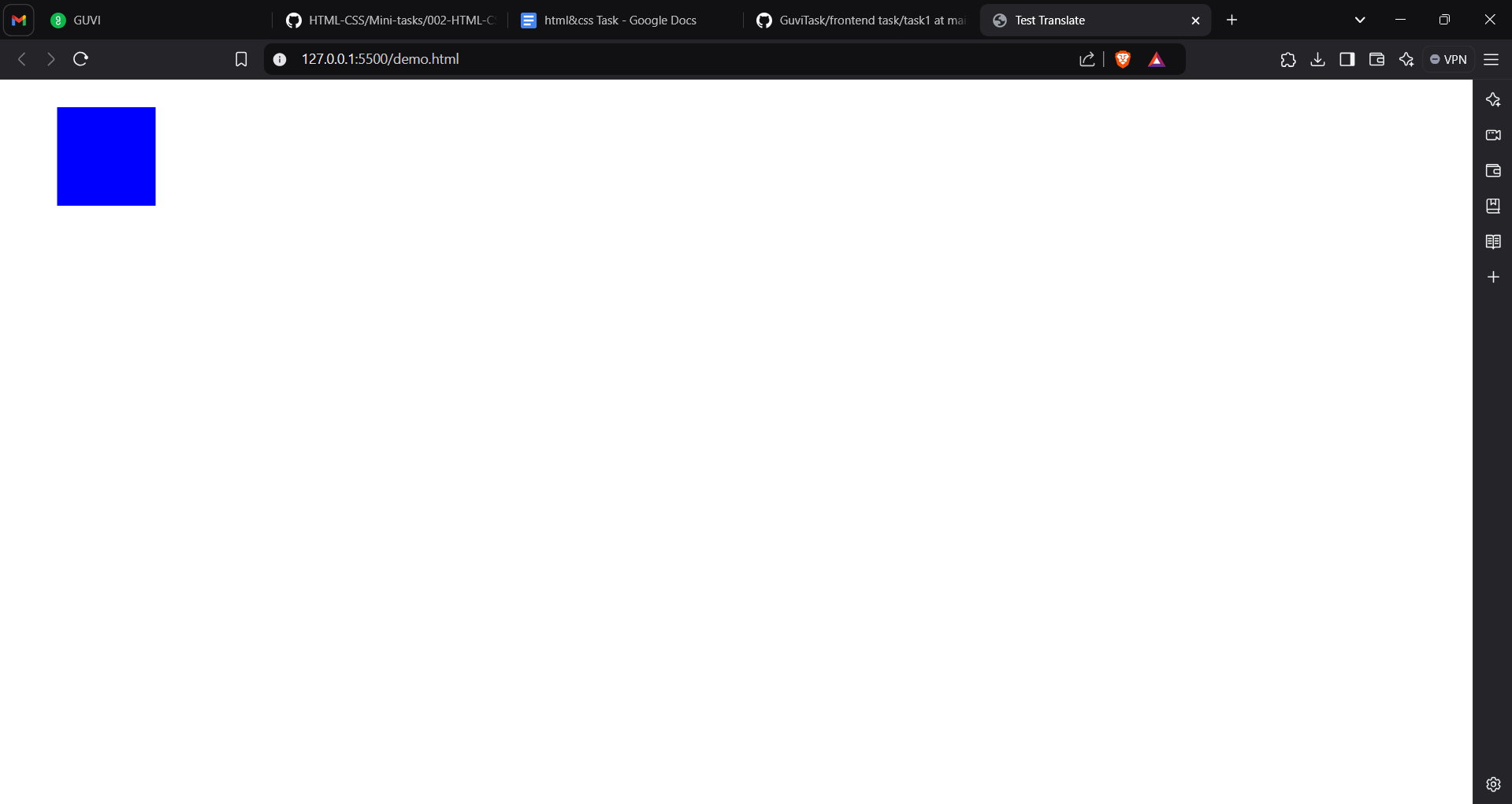
height: 100px;

background-color: blue;

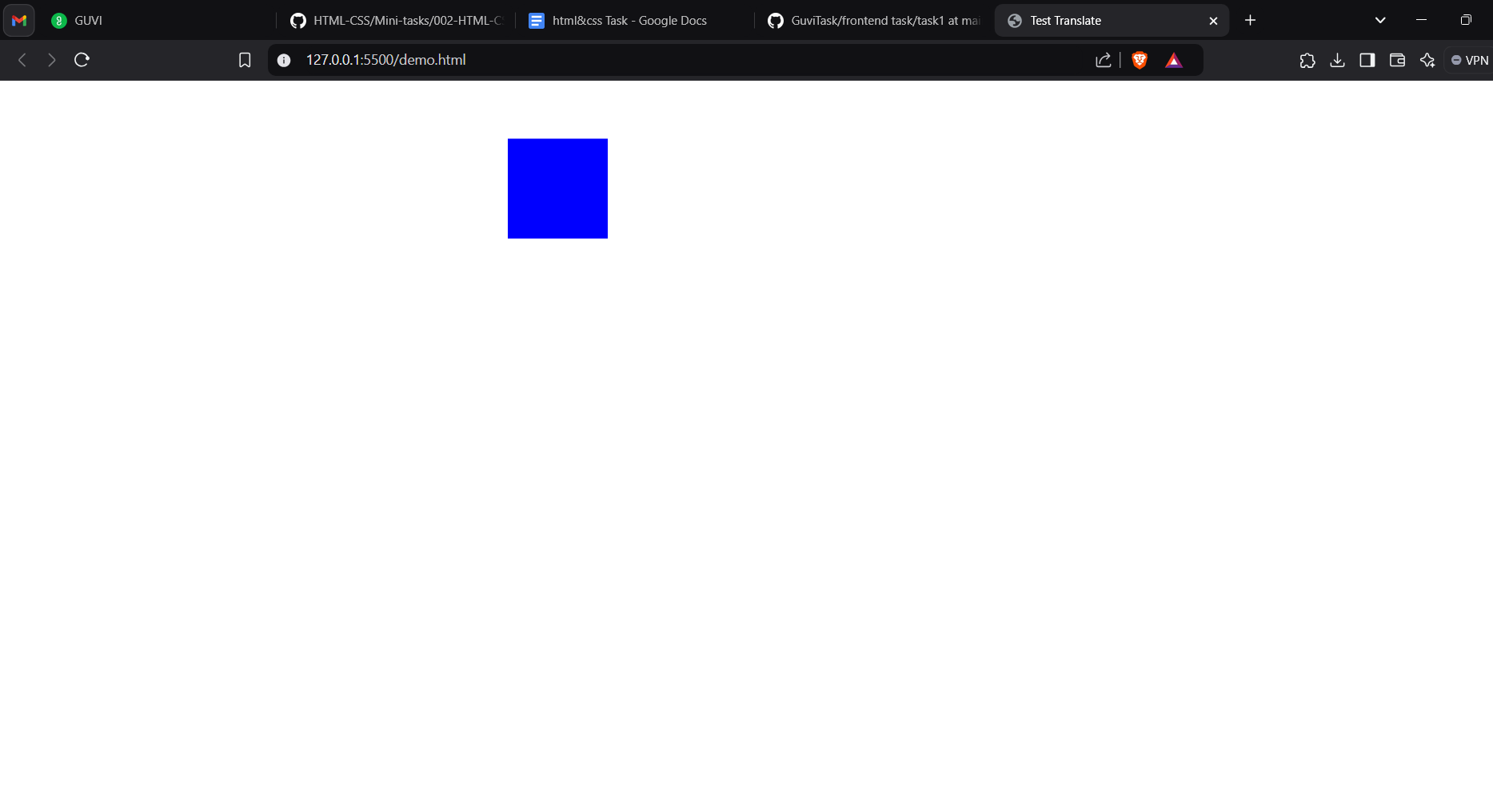
transform: translate(50px, 20px);

}

OUTPUT:



If i change the value in translate tag to 500px,50px, then output will be



8. Define a CSS class to show tooltip when mouse moved over the elements.

HTML CODE:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Tooltip Example</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="tooltip-container">

Hover over me

<div class="tooltip-text">This is a tooltip</div>

</div>

</body>

</html>

CSS CODE:

.tooltip-container {

position: relative;

display: inline-block;

cursor: pointer;

}

.tooltip-container .tooltip-text {

visibility: hidden;

width: 150px; /\* Adjust width as needed \*/

background-color: #333;

color: #fff;

text-align: center;

border-radius: 4px;

padding: 5px;

position: absolute;

bottom: 125%; /\* Position above the element \*/

left: 50%;

transform: translateX(-50%);

z-index: 1;

opacity: 0;

transition: opacity 0.3s;

}

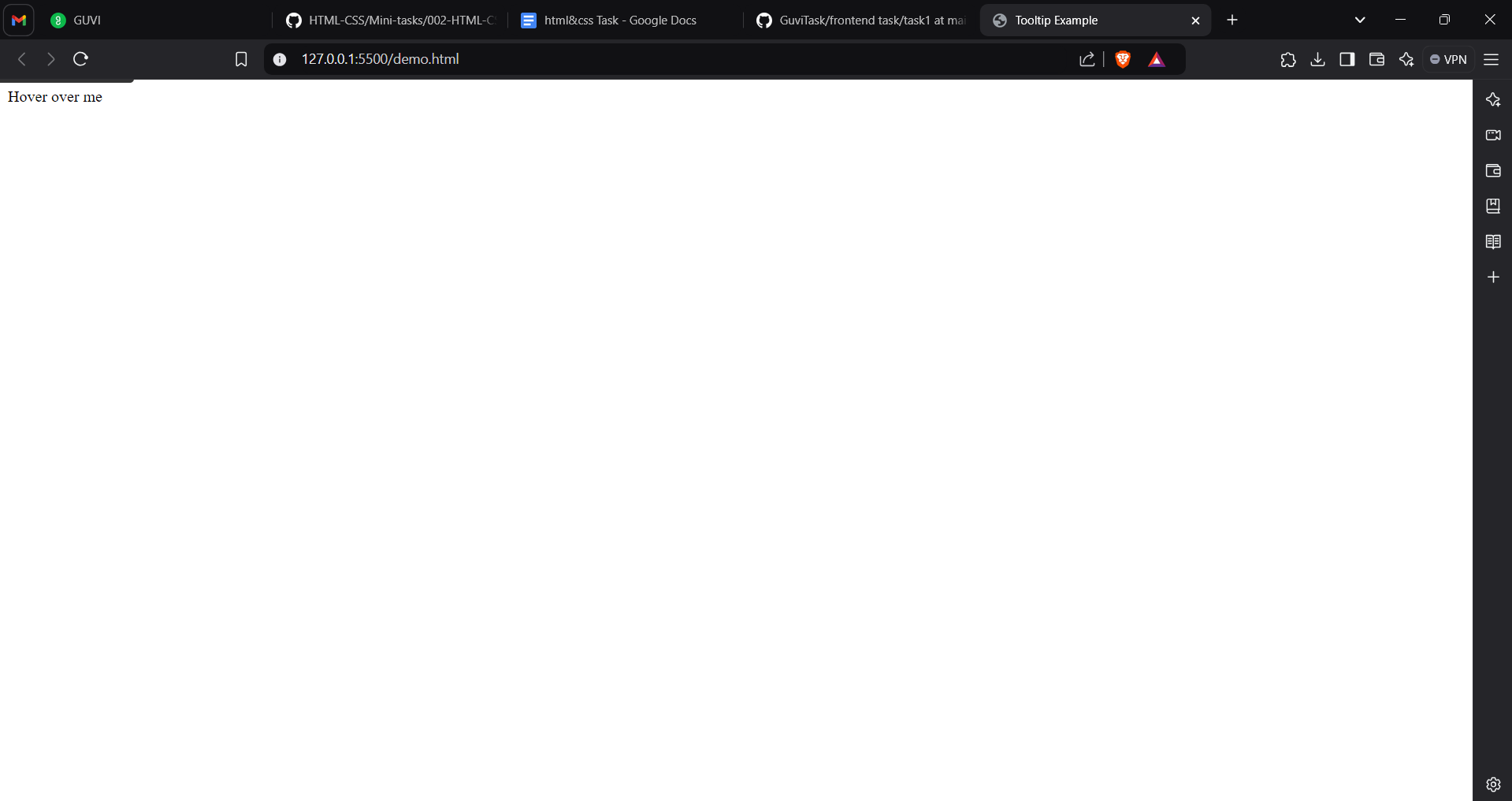
.tooltip-container:hover .tooltip-text {

visibility: visible;

opacity: 1;

}

OUTPIUT:



9. To apply clip property to an image:

HTML CODE:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Clip Property Example</title>

</head>

<body>

<img src="image/image.png" alt="GUVI Logo" class="clipped-image">

</body>

</html>

CSS CODE:

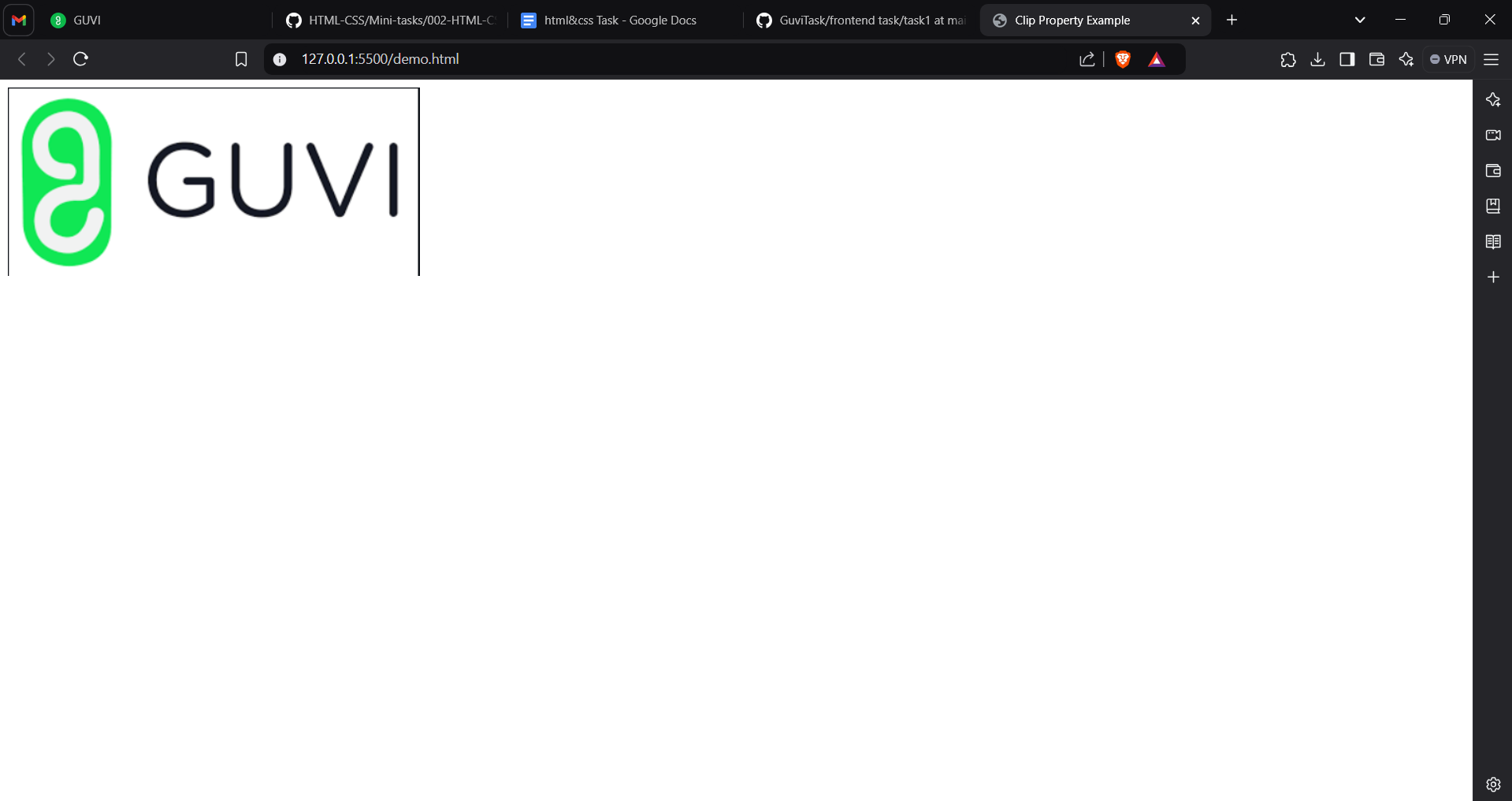
.clipped-image {

position: absolute; /\* Required for clipping \*/

clip: rect(0px, 150px, 100px, 0px); /\* Top, Right, Bottom, Left \*/

}

OUTPUT:



10. Html css code to get image:

HTML CODE:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Drop Cap Example</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="drop-cap">

Drop caps can be used to add emphasis to blocks of text but know they can affect readability.

</div>

</body>

</html>

CSS CODE:

body {

font-family: 'Georgia', serif;

background: linear-gradient(to bottom, #e0eafc, #cfdef3); /\* Simulating light texture background \*/

margin: 0;

padding: 20px;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

}

.drop-cap {

font-size: 24px; /\* Adjusted to match the image \*/

line-height: 1.8;

color: #333;

max-width: 350px;

text-align: justify;

}

.drop-cap::first-letter {

font-size: 120px; /\* Adjusted to match the large 'D' \*/

font-weight: bold;

color: #000;

float: left;

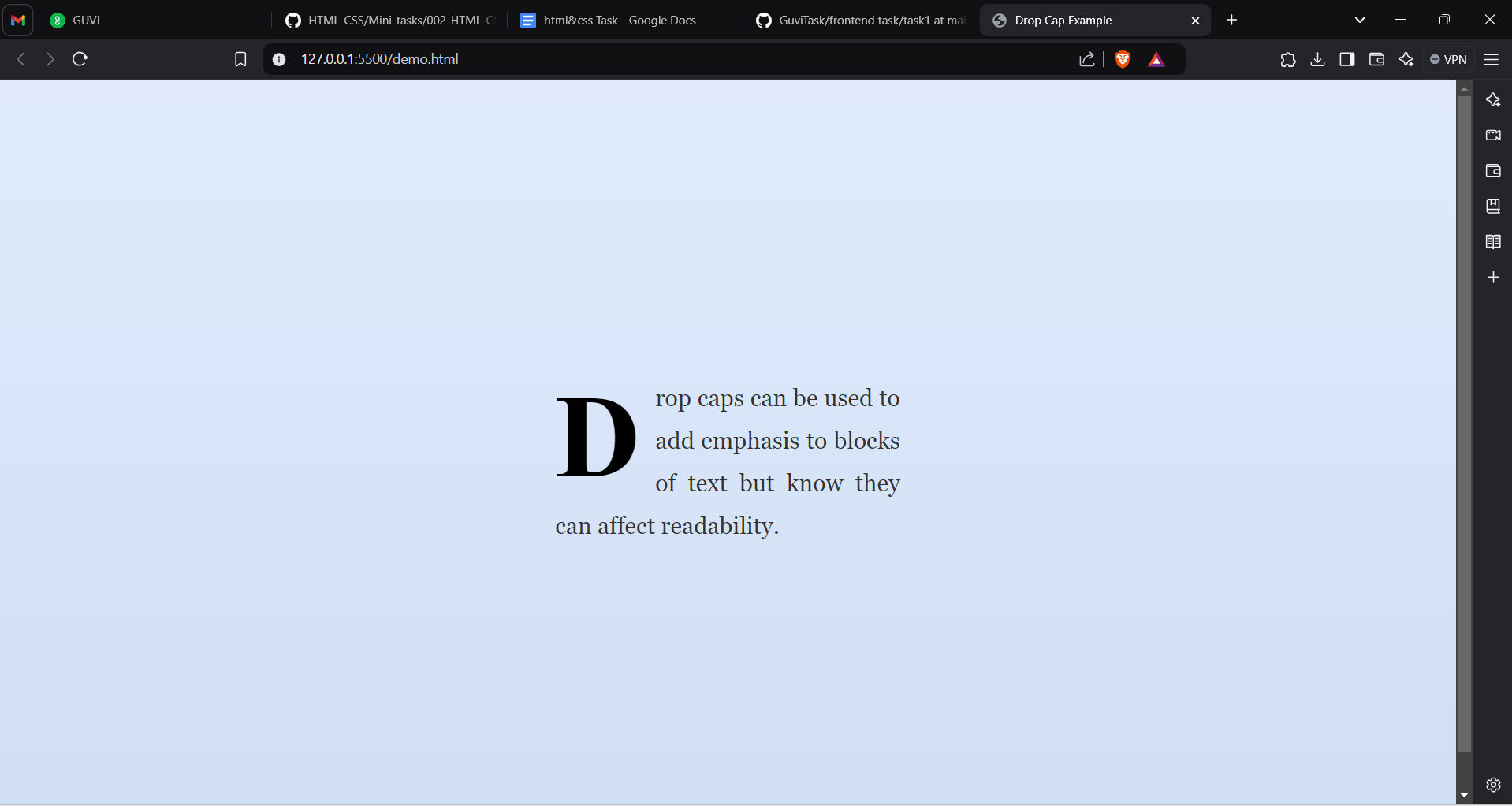
line-height: 1;

margin-right: 15px;

font-family: 'Times New Roman', serif; /\* Classic serif font for the drop cap \*/

}

OUTPUT:



11. Reshaping the given div

HTML CODE:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Diagonal Rounded Rectangle</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<div class="container">

<div class="diagonal-rounded"></div>

<div class="label">Diagonal rounded rectangle</div>

</div>

</body>

</html>

CSS CODE:

body {

font-family: Arial, sans-serif;

background-color: #f8f8f8;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

margin: 0;

}

.container {

text-align: center;

}

.diagonal-rounded {

width: 200px;

height: 150px;

background-color: #b3e5fc;

border-radius: 50px 0 0 50px; /\* Top-left and bottom-right corners are rounded \*/

margin: 0 auto;

box-shadow: 0px 4px 8px rgba(0, 0, 0, 0.2); /\* Optional shadow for depth \*/

}

.label {

margin-top: 20px;

background-color: #cccccc;

padding: 10px;

color: #333;

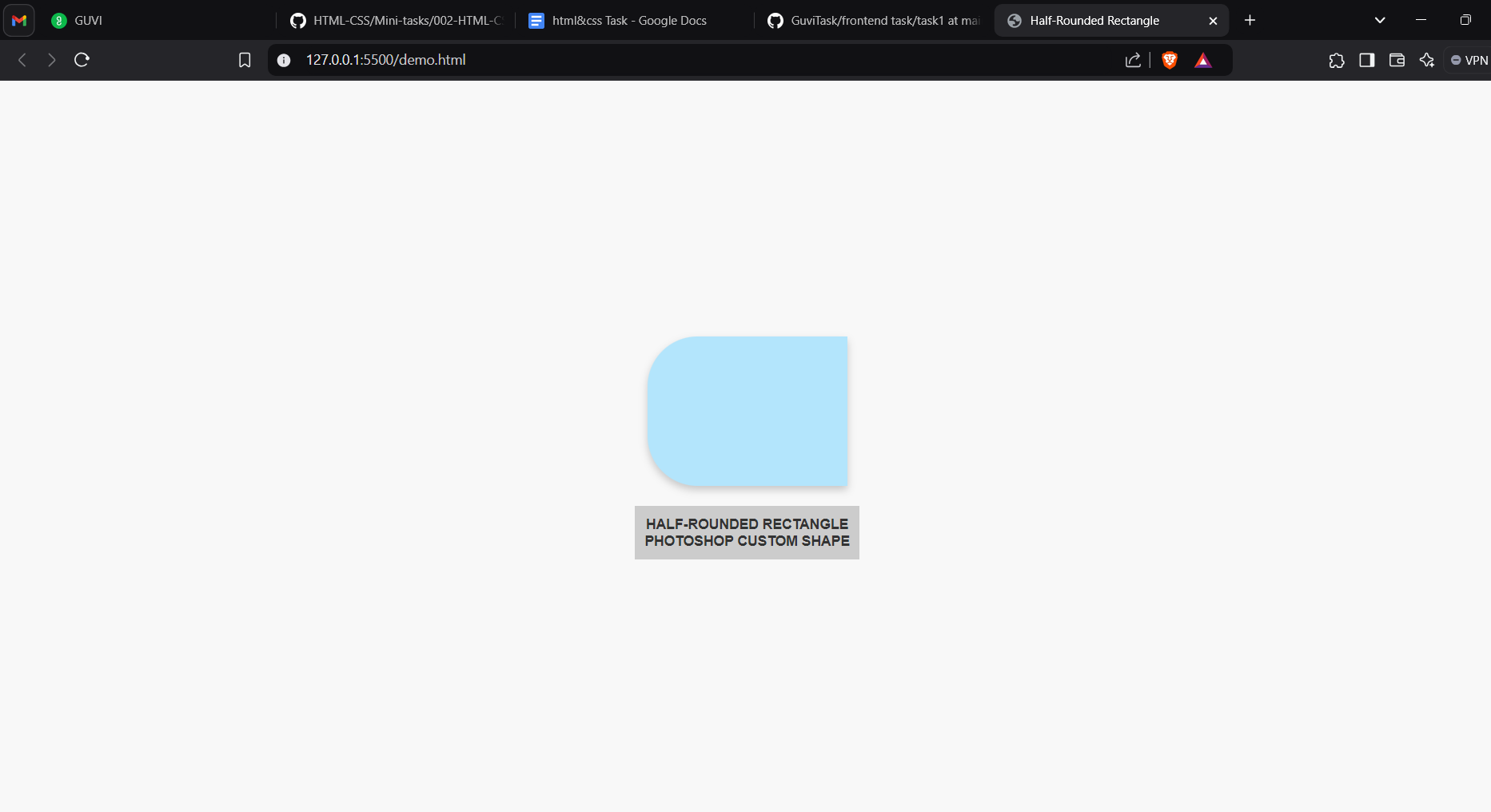
font-size: 14px;

font-weight: bold;

text-transform: uppercase;

}

OUTPUT:



12. Use CSS to set a background image should be centered and should not repeat.

body {

background-image: url('your-image-url.jpg');

background-position: center;

background-repeat: no-repeat;

}

13. Write CSS rules to set the direction of the flexible items inside the <div></div> element in reverse order.

ROW REVERSE:

div {

display: flex;

flex-direction: row-reverse;

}

COLUMN REVERSE:

div {

display: flex;

flex-direction: column-reverse;

}

14. Give the <div> element an image border using the image "border.png". Slice the image at 30px and stretch it.

div {

border: 30px solid; /\* Defines the border width \*/

border-image: url('border.png') 30 stretch; /\* Uses the image and slices it at 30px \*/

}

15. Give the <div> element a rounded corner (25px radius) on the bottom left side.

div {

border-bottom-left-radius: 25px;

}

16. Remove the transparency/opacity of the <img> element when the user hovers over it with the mouse pointer.

img {

opacity: 0.5; /\* Initial opacity (can be adjusted) \*/

transition: opacity 0.3s ease; /\* Smooth transition for opacity change \*/

}

img:hover {

opacity: 1; /\* Full opacity on hover \*/

}

17. Position the <div> element all the way to the right using absolute positioning.

body {

position: relative;

}

div {

position: absolute;

right: 0;

}

18. Add a scrollbar to the <div> element.

<div style="width: 300px; height: 200px; overflow: auto;">

<!-- Your content here, e.g., text or images -->

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus lacinia augue at sapien cursus, at efficitur nunc posuere. Nulla facilisi.

<br><br>

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus lacinia augue at sapien cursus, at efficitur nunc posuere. Nulla facilisi.

<br><br>

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus lacinia augue at sapien cursus, at efficitur nunc posuere. Nulla facilisi.

</div>

19. Set a linear gradient background for the <div> element, going from the top to bottom, transitioning from "white" to "red" to "blue" to "green".

<div style="width: 300px; height: 200px; background: linear-gradient(to bottom, white, red, blue, green; overflow: auto;">

<!-- Your content here -->

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus lacinia augue at sapien cursus, at efficitur nunc posuere. Nulla facilisi.

</div>

20. With the transform property, rotate the <div> element 45 degrees.

<div style="width: 300px; height: 200px; background: linear-gradient(to bottom, white, red, blue, green); overflow: auto; transform: rotate(45deg);">

<!-- Your content here -->

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus lacinia augue at sapien cursus, at efficitur nunc posuere. Nulla facilisi.

</div>