**A close up of a logo

AI-generated content may be incorrect.**

**Mini Project Report - 02**

Master of Computer Application – Data Science

Semester – I

**Sub: Frontend Framework And Technologies**

**Topic: Student Details**   
By  
**Name:** DECHAMMA KP  
**Reg no.:** PROV/ASAC/MCA/7/25/040

**Faculty Name:** VEERA RAGHAV K

**Faculty Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Department of Computer Application  
Alliance University  
Chandapura - Anekal Main Road, Anekal  
Bengaluru - 562 106**

**September 2025**

**INDEX**

1. Introduction to CSS
2. Input code
3. Output
4. Conclusion

**INTRODUCTION**

**Introduction to CSS**

CSS (Cascading Style Sheets) is the language that makes websites visually attractive and user-friendly. While HTML provides the structure of a webpage, CSS is responsible for its design—controlling colors, fonts, spacing, layouts, and even animations. With CSS, the same HTML content can be presented in completely different styles,giving web designers full creative control over how a site looks and feels.

In CSS:

• **Selector** → Targets the HTML element you want to style (e.g., h1).

• **Property** → Specifies the aspect to change (e.g., color, font-size).

• **Value** → Defines the exact style setting (e.g., blue, 20px).

## **Why Do We Use CSS?**

To **separate content (HTML)** from **design (CSS)**.

To make websites **look beautiful and user-friendly**.

To **reuse styles** across multiple pages (instead of writing inline styles everywhere).

To create **responsive designs** that look good on phones, tablets, and desktops.

## **Key Features of CSS**

****Cascading****: If multiple styles apply, the latest or more specific one wins.

****Selectors**:** Target elements by tag, class, id, or even conditions.

****Responsive Design**:** Media queries make websites adjust to screen sizes.

****Reusability****: One CSS file can style an entire website.

# **CSS Selectors:**

**Element Selector** → p { } targets all <p> elements.

**Class Selector** → .className { } targets elements with a class.

**ID Selector** → #idName { } targets a specific element.

**Universal Selector** → \* { } applies to all elements.

**Group Selector** → h1, h2, p { } applies to multiple elements.

# **CSS Box Model**

Every HTML element is like a box, consisting of:

**Content** → Text or image.

**Padding** → Space inside the box.

**Border** → Line around the padding.

**Margin** → Space outside the box.

**INPUT CODE:**

<!DOCTYPE html>

<html lang="en">

<head>

    <title>Student details</title>

    <style>

        div{

        text-align: center;

        background-color: aqua;

        width: 500px;

        padding:5px;

        margin-left:350px;

        border-radius: 20%;

        margin-top: 40px;

        box-shadow: 4px 4px 10px black;

        }

        #p1{

            margin-right:100px;

        }

        .p2{

            margin-left:100px;

        }

    </style>

</head>

<body>

    <div>

      <h1><u>Student details</u></h1>

      <p>Sreg:123456789</p>

      <p>Sname:DECHAMMA KP</p>

      <p id="p1">Scourse:MCA</p>

      <p class="p2">Semail:@dech</p>

      <p>Saddress:Kodagu</p>

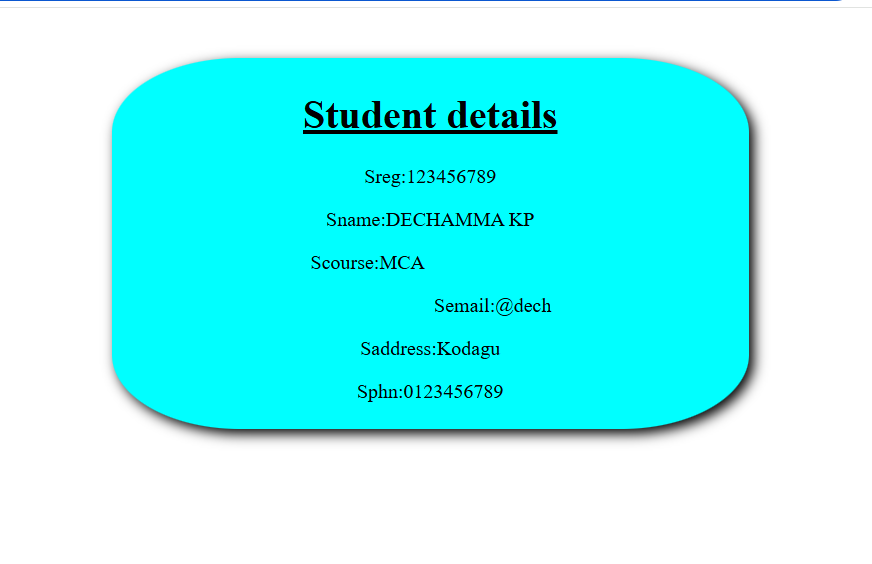
      <p>Sphn:0123456789</p>

    </div>

</body>

</html>

**OUTPUT :**



**CONCLUSION:**

This mini project demonstrates how HTML and CSS can be used to design a simple and structured Student Details web page. It helped me understand the basics of web development, including page layout, styling, and presenting information in a clean and user-friendly way. By applying CSS properties like background color, text alignment, padding, margin, border-radius, and box-shadow, the output looks more attractive and user-friendly.

This project helped me understand the fundamentals of:

• Structuring a webpage using HTML elements (div, p, h1).

• Applying CSS styling to improve design and layout.

• Creating a simple user interface that is easy to read and navigate.