

Name: Anlea Maria Jose, SE AIML

Roll No.: 02

Date: 27-01-2026

Web Programming Lab - Exp 3

26-01-2026 class work

Instructionsindex.htmlstyles.css

Step 2

Add a `header` element within the `body` element and assign a class of `header` to it.

Inside the `header`, create an `h1` with `css flexbox photo gallery` as the text.

index.html

```
1 <!DOCTYPE html>
2 <html>
3   <head lang="en">
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width,
6       initial-scale=1.0">
7     <title>Photo Gallery</title>
8     <link rel="stylesheet" href="./styles.css">
9   </head>
10  <body>
11    <header class="header">
12      <h1>css flexbox photo gallery</h1>
13    </body>
14  </html>
```

✓ Congratulations. Your code passes.

css flexbox photo gallery

Instructionsindex.htmlstyles.css

Step 5

In order to better visualize how your elements are sized, adding a border can be helpful.

Give your `.gallery` element a `width` of `50%` and a `border` set to `5px solid red`.

Then give your `img` elements a `width` of `100%`, `padding` set to `5px`, and a `border` set to `5px solid blue`.

index.html

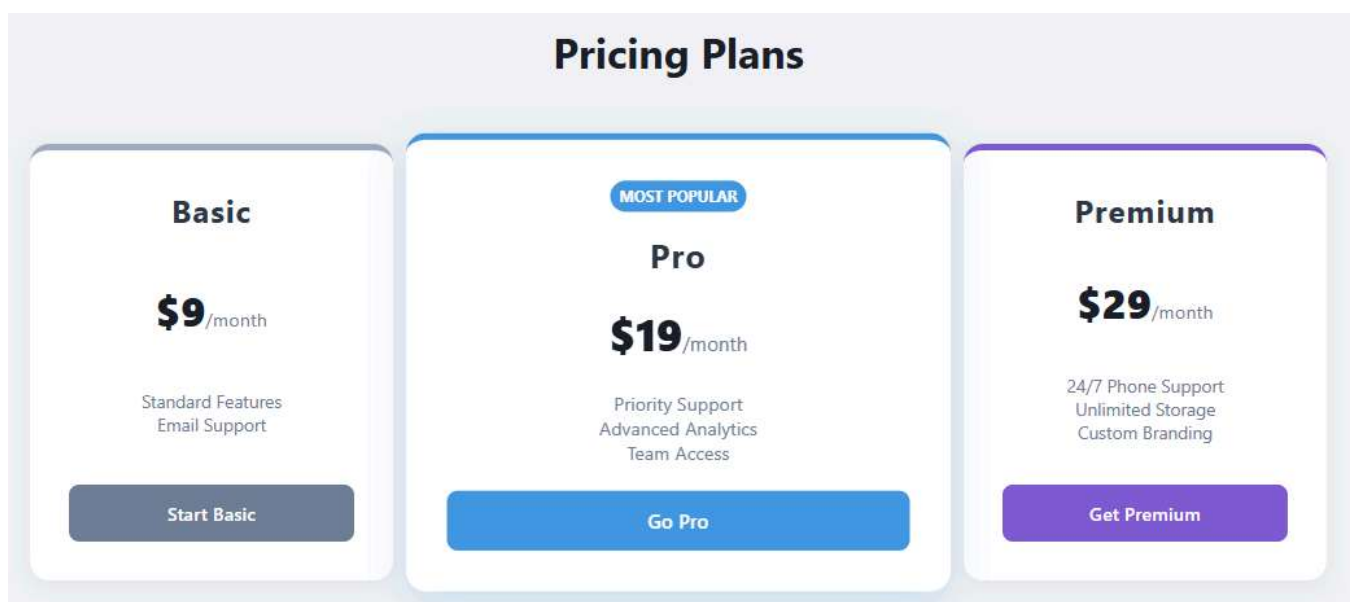
```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="utf-8">
5     <meta name="viewport"
6       content="width=device-width, initial-scale=1.0">
7     <title>Photo Gallery</title>
8     <link
9       rel="stylesheet" href="./styles.css">
10   </head>
11   <body>
12     <header
13       class="header">
14       <h1>css flexbox
15         photo gallery</h1>
16     </header>
17     <div class="gallery">
18       
3 <head>
4 <meta charset="UTF-8">
5 <title>Pricing Plans</title>
6 <style>
7 body{font-family: 'Segoe UI',Tahoma, Geneva, Verdana, sans-serif;padding:40px;background-color: #f0f2f5;color: #333;}
8 h1{text-align:center;margin-bottom:50px;font-weight:700;color: #1a202c;}
9 .pricing-container{display:flex;flex-wrap:wrap;gap:20px;justify-content:center;align-items:stretch;max-width:1000px;margin:auto;}
10 .pricing-card{display:flex;flex-direction:column;justify-content:space-between;flex:0 0 220px;padding:30px;background-color: #white;
11 text-align:center;min-height:250px;border-radius:15px;box-shadow:0 4px 15px #000;
12 transition:transform 0.3s ease,box-shadow 0.3s ease;border:none;}
13 .pricing-card:hover{transform:translateY(-5px);box-shadow:0 12px 20px #000;}
14 .basic-plan{order:0;border-top:5px solid #a0aec0;}
15 .pro-plan{order:1;flex-grow:2;border-top:5px solid #4299e1;transform:scale(1.05);z-index:1;box-shadow:0 10px 25px #666;}
16 .pro-plan:hover{transform:scale(1.07) translateY(-5px);}
17 .premium-plan{order:2;border-top:5px solid #805ad5;}
18 h2{margin:0;font-size:1.5rem;color: #2d3748;letter-spacing:1px;}
19 p{font-size:2rem;font-weight:800;margin:20px 0;color: #1a202c;}
20 p span{font-size:0.9rem;font-weight:400;color: #718096;}
21 .btn{margin-top:20px;padding:12px;background: #4299e1;color: #white;text-decoration:none;border-radius:8px;font-weight:600;font-size:0.9rem;}
22 .basic-plan .btn{background: #718096;}
23 .premium-plan .btn{background: #805ad5;}
```



```

24 </style>
25 </head>
26 <body>
27   <h1>Pricing Plans</h1>
28   <div class="pricing-container">
29     <div class="pricing-card basic-plan">
30       <h2>Basic</h2>
31       <p>$9<span>/month</span></p>
32       <div style="font-size:0.85rem;color:#718096;">Standard Features<br>Email Support</div>
33       <a href="#" class="btn">Start Basic</a>
34     </div>
35     <div class="pricing-card pro-plan">
36       <div style="background:#4299e1;color:white;font-size:0.7rem;padding:4px 0;border-radius:20px;width:100px;
37       margin:0 auto 15px;text-transform:uppercase;font-weight:bold;">Most Popular</div>
38       <h2>Pro</h2>
39       <p>$19<span>/month</span></p>
40       <div style="font-size:0.85rem;color:#718096;">Priority Support<br>Advanced Analytics<br>Team Access</div>
41       <a href="#" class="btn">Go Pro</a>
42     </div>
43     <div class="pricing-card premium-plan">
44       <h2>Premium</h2>
45       <p>$29<span>/month</span></p>
46       <div style="font-size:0.85rem;color:#718096;">24/7 Phone Support<br>Unlimited Storage<br>Custom Branding</div>
47       <a href="#" class="btn">Get Premium</a>
48     </div>
49   </div>
50 </body>
51 </html>

```



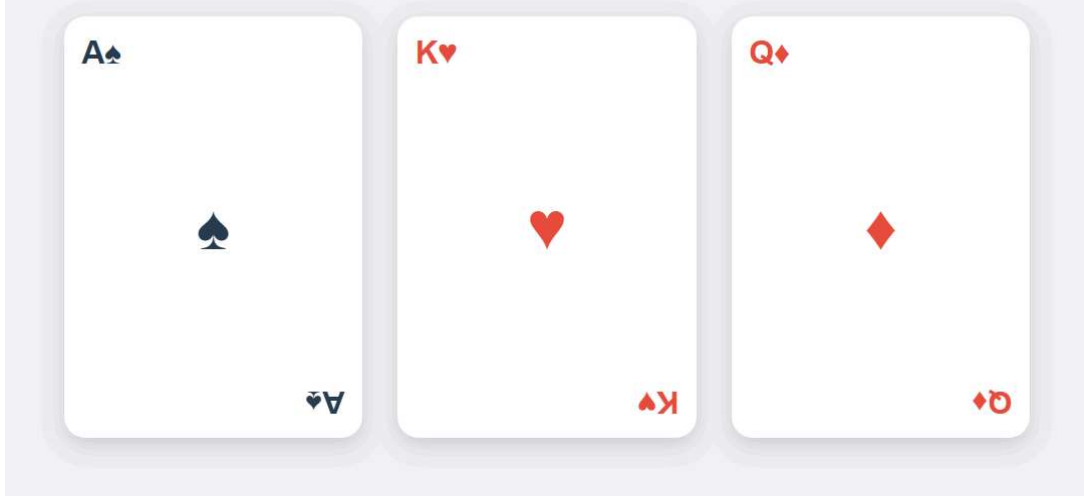
2. Build a Page of Playing Cards

```

C: > Users > HP > Downloads > part_2_PlayingCards.html > ...
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>Playing Cards</title>
7 <style>
8 body{margin:0;font-family:sans-serif;background:#f0f0f5;display:flex;flex-direction:column;align-items:center;min-height:100vh}
9 h1{margin:20px 0;color:#0a0a23;text-transform:uppercase;letter-spacing:2px}
10 #playing-cards{display:flex;flex-wrap:wrap;justify-content:center;gap:20px} .card{display:flex;flex-direction:column;
11 justify-content:space-between;width:150px;height:220px;padding:10px;border-radius:12px;background:#fff;
12 box-shadow:0 4px 12px rgba(0,0,0,0.15);font-weight:bold;color:#333;position:relative}.left,.right{font-size:18px}
13 .right{align-self:flex-end;transform:rotate(180deg)}.middle{display:flex;flex-direction:column;align-items:center;
14 justify-content:center;font-size:36px}.red{color:#e74c3c}.black{color:#2c3e50}
15 </style>
16 </head>
17 <body>
18 <h1>Playing Cards</h1>
19 <main id="playing-cards">
20 <div class="card"><div class="left black">A♠</div><div class="middle black">♠</div><div class="right black">A♠</div></div>
21 <div class="card"><div class="left red">K♥</div><div class="middle red">♥</div><div class="right red">K♥</div></div>
22 <div class="card"><div class="left red">Q♦</div><div class="middle red">♦</div><div class="right red">Q♦</div></div>
23 </main>
24 </body>
25 </html>

```

PLAYING CARDS



3. Build a Technical Documentation Page

```
C:\> Users\HP\Downloads> part_3_TechnicalDoc.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <meta charset="UTF-8">
5  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  <title>Technical Documentation</title>
7  <style>
8  body{margin:0;font-family:'Segoe UI',sans-serif;display:flex;background:#f0f2f5;color:#222}
9  #navbar{position:fixed;left:0;top:0;width:220px;height:100vh;padding:20px;background:#1b1f3b;color:#fff;overflow:auto}
10 #navbar header{font-size:22px;margin-bottom:20px;text-transform:uppercase;letter-spacing:1px}#navbar a{display:block;color:#fdb347;
11 text-decoration:none;margin:6px 0;padding:4px;border-radius:4px;transition:0.2s}#navbar a:hover{background:#fdb347;color:#1b1f3b}
12 #main-doc{margin-left:240px;padding:30px;flex:1}section{margin-bottom:40px;background:#fff;padding:20px;border-radius:12px;
13 box-shadow:0 4px 15px rgba(0,0,0,0.1)}section header{font-size:24px;margin-bottom:12px;color:#0a0a23}p{margin:10px 0;
14 line-height:1.6}code{background:#f4f4f4;padding:2px 6px;border-radius:5px;font-family:monospace;color:#c7254e}
15 ul{margin:10px 0 10px 20px}li{margin:5px 0}@media(max-width:768px){#navbar{position:relative;width:100%;height:auto}
16 #main-doc{margin-left:0;padding:20px}}
17 </style>
18 </head>
19 <body>
20 <nav id="navbar">
21 <header>JS Docs</header>
22 <a class="nav-link" href="#Introduction">Introduction</a>
23 <a class="nav-link" href="#Variables_and_Constants">Variables and Constants</a>
24 <a class="nav-link" href="#Functions">Functions</a>
25 <a class="nav-link" href="#Loops">Loops</a>
26 <a class="nav-link" href="#Objects">Objects</a>
27 </nav>
28 <main id="main-doc">
29 <section class="main-section" id="Introduction">
30 <header>Introduction</header>
```

```
31 <p>JavaScript is a versatile programming language used for web development.</p>
32 <p>It allows dynamic interaction on webpages and supports multiple paradigms.</p>
33 <p>It runs in browsers and on servers using Node.js.</p>
34 <code>console.log("Hello World");</code>
35 <p>JS supports variables, functions, objects, loops, and more.</p>
36 <ul><li>Easy to learn</li><li>Dynamic typing</li><li>Event-driven</li></ul>
37 </section>
38 <section class="main-section" id="Variables_and_Constants">
39 <header>Variables and Constants</header>
40 <p>Variables store data values using <code>var</code>, <code>let</code>, or <code>const</code>.</p>
41 <p>Use <code>const</code> for values that should not change.</p>
42 <code>let name = "Alice";</code>
43 <code>const pi = 3.14;</code>
44 <ul><li>var: function scoped</li><li>let: block scoped</li><li>const: block scoped, read-only</li></ul>
45 </section>
46 <section class="main-section" id="Functions">
47 <header>Functions</header>
48 <p>Functions encapsulate reusable code blocks.</p>
49 <p>They can accept parameters and return values.</p>
50 <code>function sum(a,b){return a+b;}</code>
51 <code>const multiply=(x,y)=>x*y;</code>
52 <ul><li>Function declaration</li><li>Function expression</li><li>Arrow functions</li></ul>
53 </section>
54 <section class="main-section" id="Loops">
55 <header>Loops</header>
```

```

56 <p>Loops are used to execute code multiple times.</p>
57 <p>JavaScript supports <code>for</code>, <code>while</code>, and <code>do...while</code> loops.</p>
58 <code>for(let i=0;i<5;i++){console.log(i);}</code>
59 <code>while(condition){...}</code>
60 <ul><li>for loop</li><li>while loop</li><li>do while loop</li></ul>
61 </section>
62 <section class="main-section" id="Objects">
63 <header>Objects</header>
64 <p>Objects store key-value pairs of data.</p>
65 <p>They are fundamental for structuring complex data.</p>
66 <code>let person={name:"Bob",age:25};</code>
67 <p>Access properties with dot or bracket notation.</p>
68 <code>console.log(person.name);</code>
69 <ul><li>Dynamic properties</li><li>Nested objects</li><li>Methods</li></ul>
70 </section>
71 </main>
72 </body>
73 </html>

```

JS DOCS

Introduction

Variables and Constants

Functions

Loops

Objects

Introduction

JavaScript is a versatile programming language used for web development.

It allows dynamic interaction on webpages and supports multiple paradigms.

It runs in browsers and on servers using Node.js.

```
console.log("Hello World");
```

JS supports variables, functions, objects, loops, and more.

- Easy to learn
- Dynamic typing
- Event-driven

Variables and Constants

Variables store data values using `var`, `let`, or `const`.

Use `const` for values that should not change.

```
let name = "Alice"; const pi = 3.14;
```

- var: function scoped
- let: block scoped
- const: block scoped, read-only

Functions

Functions encapsulate reusable code blocks.

They can accept parameters and return values.

```
function sum(a,b){return a+b}; const multiply=(x,y)=>x*y;
```

- Function declaration
- Function expression
- Arrow functions

Loops

Loops are used to execute code multiple times.

JavaScript supports `for`, `while`, and `do...while` loops.

```
for(let i=0;i<5;i++){console.log(i);} while(condition){...}
```

- for loop
- while loop
- do while loop

Objects

Objects store key-value pairs of data.

They are fundamental for structuring complex data.

```
let person={name:"Bob",age:25};
```

Access properties with dot or bracket notation.

```
console.log(person.name);
```

- Dynamic properties
- Nested objects
- Methods

4. A Newspaper Layout

Code: https://github.com/anlea0/WEB-PROGLAB-2/blob/main/Exp3/part-4_NewspaperLayout.html

THE DAILY GRID

Established 2026 · All the Layout That Fits

Breaking: CSS Grid Revolutionizes Web Design

In an unprecedented development in the world of web development, CSS Grid has transformed how designers think about layout. This two-dimensional system replaces brittle hacks with clarity and intent, unlocking layouts once reserved for print. Experts agree: Grid is no longer the future — it is the standard.



Print inspiration meets modern CSS

Local Community Embraces Digital Transformation

Community workshops are drawing record crowds as citizens explore modern layout tools. By blending traditional design sensibilities with cutting-edge CSS, participants are rediscovering the craft of storytelling — one grid at a time.

Weekly Workshop

Thursday nights: modern CSS layouts, zero frameworks, pure control.

Developer Spotlight

Sarah Chen recreates historic newspapers using nothing but CSS Grid.

Tech Tips

Use grid-template-columns with intent. Think like an editor, not a browser.

5. Build a Product Landing Page

Code: https://github.com/anlea0/WEB-PROGLAB-2/blob/main/Exp3/part-5_ProductLandingPage.html

Pure Sound. Minimal Design.

NovaSound delivers beautifully balanced audio wrapped in a refined, modern experience.

[Get Early Access](#)

Crystal Clear Audio

Precision-tuned drivers for immersive, distortion-free sound.

Wireless Freedom

Low-latency Bluetooth with all-day battery performance.

Timeless Build

Minimal design crafted from premium materials.

See NovaSound in Action

[Watch video on YouTube](#)

Error 153
Video player configuration error



Choose Your Experience

Essential

\$99

Everyday premium listening

Pro

\$149

Enhanced depth & clarity

Studio

\$199

Built for creators

[Notify Me](#)