

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Grid Layout</title>
  <link rel="stylesheet" href="style.css">
  <link
href="https://fonts.googleapis.com/css2?family=Bungee&family=Rubik:wght@400;500;7
00&display=swap" rel="stylesheet">
</head>
<body style="background-color: #f4f4f4;">

  <div class="grid-container">
    <header class="main-header">
      <div class="logo"></div>
      <h1>Monkey Business</h1>

      <nav class="main-nav">
        <ul>
          <li class="links"><a href="index.html">Home</a></li>
          <li class="links"><a href="about.html">About</a></li>
          <li class="links"><a href="contact.html">Contact</a></li>
        </ul>
      </nav>
    </header>

    <div class="side-nav">
      <h2>Contents</h2>
      <ul>
        <li><a href="#">Home</a></li>
        <li><a href="#">Ape</a></li>
        <li><a href="#">Chimpanzee</a></li>
      </ul>
    </div>

    <div class="main">
      <h2>Overview</h2>
      <div class="text-area">
        <p>Welcome to the Monkey Business website! Here, we explore
various topics related to monkeys and their habitats.</p>
        <p>Bakit mahilig sa saging ang mga unggoy?</p>
```

```
        </div>
    </div>

    <div class="sidebar">
        <h2>Sidebar</h2>
        <ul>
            <li><a href="#">What do monkeys eat?</a></li>
            <li><a href="#">Where do we find them?</a></li>
            <li><a href="#">Biggest Monkey</a></li>
        </ul>
    </div>
    <footer class="footer">
        &copy; 2025 Your Website. All rights reserved.
    </footer>
</div>
</body>
</html>
```

CSS

```
body {
    font-family: 'Rubik', sans-serif;
    background-color: #F0F2F5;
    color: #2C3E50;
    padding: 20px;
}

/* Main Grid Layout */

.grid-container {
    display: grid;
    grid-template-areas:
        "header header header"
        "nav main sidebar"
        "footer footer footer";
    grid-template-columns: 200px 1fr 250px;
    grid-template-rows: auto 1fr auto;
    gap: 20px;
    max-width: 1200px;
    margin: auto;
    padding: 20px;
}

/* Header Section */
.main-header {
```

```
    grid-area: header;
    display: flex;
    justify-content: space-between;
    align-items: center;
    flex-wrap: wrap;
    background-color: #E5C531;
    color: #471e02;
    padding: 30px;
    border-radius: 1.5rem;
    border-bottom: 5px solid #1ABC9C;
    box-shadow: 0 6px 16px rgba(0, 0, 0, 0.3);
    margin: 0;
}

.logo {
    width: 60px;
    height: 60px;
    background-color: #f4f4f4;
    border-radius: 50%;
    display: flex;
    align-items: center;
    justify-content: center;
    flex-shrink: 0;
    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.15);
    margin-left: 5%;
}

.main-header h1 {
    font-family: 'Bungee', cursive;
    font-size: 2.5rem;
    letter-spacing: 1px;
    text-transform: uppercase;
}

.main-nav ul {
    list-style: none;
    display: flex;
    gap: 20px;
    padding: 20px 0;
    margin: 0;
}

.links a {
    text-decoration: none;
    color: #2C3E50;
```

```
    font-weight: 500;
    transition: color 0.3s ease;
}

.links a:hover {
    color: #A3CB38;
}

/*sidebar and side navigationd*/
.side-nav {
    grid-area: nav;
    background-color: #E5C531;
    padding: 20px;
    border-radius: 10px;
    box-shadow: 0 4px 12px rgba(0, 0, 0, 0.15);
}

.side-nav h2, .sidebar h2 {
    color: #471e02;
    margin-bottom: 20px;
    font-size: 1.5rem;
    text-align: center;
    padding-bottom: 10px;
    border-bottom: 2px solid #471e02;
}

.side-nav ul, .sidebar ul {
    list-style: none;
    margin: 0;
    padding: 0;
    display: flex;
    flex-direction: column;
    gap: 15px;
}

.side-nav li, .sidebar li {
    background-color: #471e02;
    border-radius: 8px;
    transition: transform 0.2s ease, box-shadow 0.2s ease;
}

.side-nav li:hover, .sidebar li:hover {
    transform: translateY(-2px);
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
}
```

```
.side-nav a, .sidebar a {
  display: block;
  padding: 12px 18px;
  font-weight: 600;
  color: #ffffff;
  text-decoration: none;
  transition: background 0.3s ease;
  border-radius: 8px;
  position: relative;
}

.side-nav a:hover, .sidebar a:hover {
  background-color: #a84116;
}

.main {
  grid-area: main;
  background-color: #E5C531;
  padding: 20px;
  border-radius: 10px;
  box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);
}

.main h2 {
  color: #491d02;
  margin-bottom: 20px;
  font-size: 1.5rem;
  padding-bottom: 10px;
  border-bottom: 2px solid #471e02;
}

.text-area {
  background-color: #f7f7f7;
  color: #444444;
  padding: 20px;
  border-radius: 10px;
  box-shadow: 0 2px 6px rgba(0, 0, 0, 0.1);
}

.sidebar {
  grid-area: sidebar;
  background-color: #E5C531;
  padding: 20px;
  border-radius: 10px;
  box-shadow: 0 4px 12px rgba(0, 0, 0, 0.15);
}
```

```
}

.footer {
  grid-area: footer;
  background-color: #3B7A57;
  color: #fff;
  text-align: center;
  padding: 15px;
  border-radius: 10px;
  box-shadow: 0 2px 8px rgba(0, 0, 0, 0.1);
}

/* Responsive for 992PX */
@media (max-width: 992px) and (min-width: 769px) {
  .grid-container {
    grid-template-areas:
      "header header header"
      "nav main main"
      "sidebar sidebar sidebar"
      "footer footer footer";
    grid-template-columns: 200px 1fr 0;
  }
}

/* Responsive for 768px */
@media (max-width: 768px) {
  .grid-container {
    grid-template-areas:
      "header"
      "nav"
      "main"
      "sidebar"
      "footer";
    grid-template-columns: 1fr;
  }

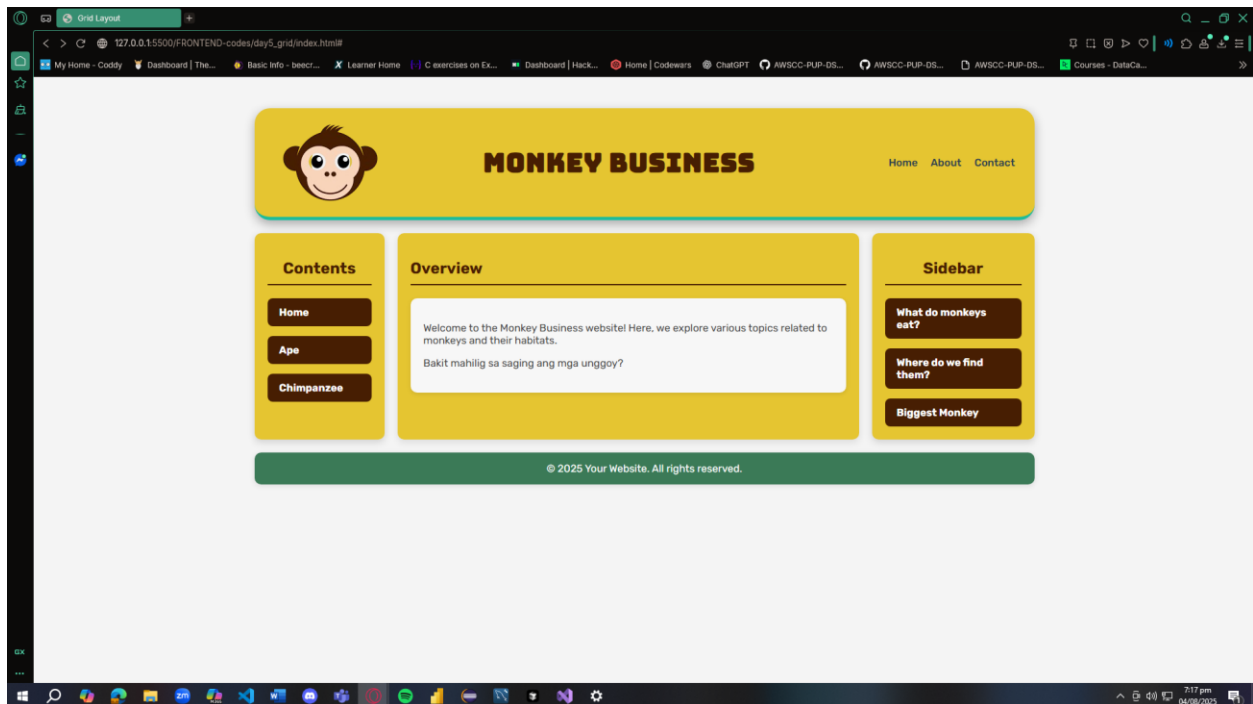
  .main-header {
    flex-direction: column;
    align-items: flex-start;
  }

  .main-nav ul {
    flex-direction: column;
    gap: 10px;
  }
}
```

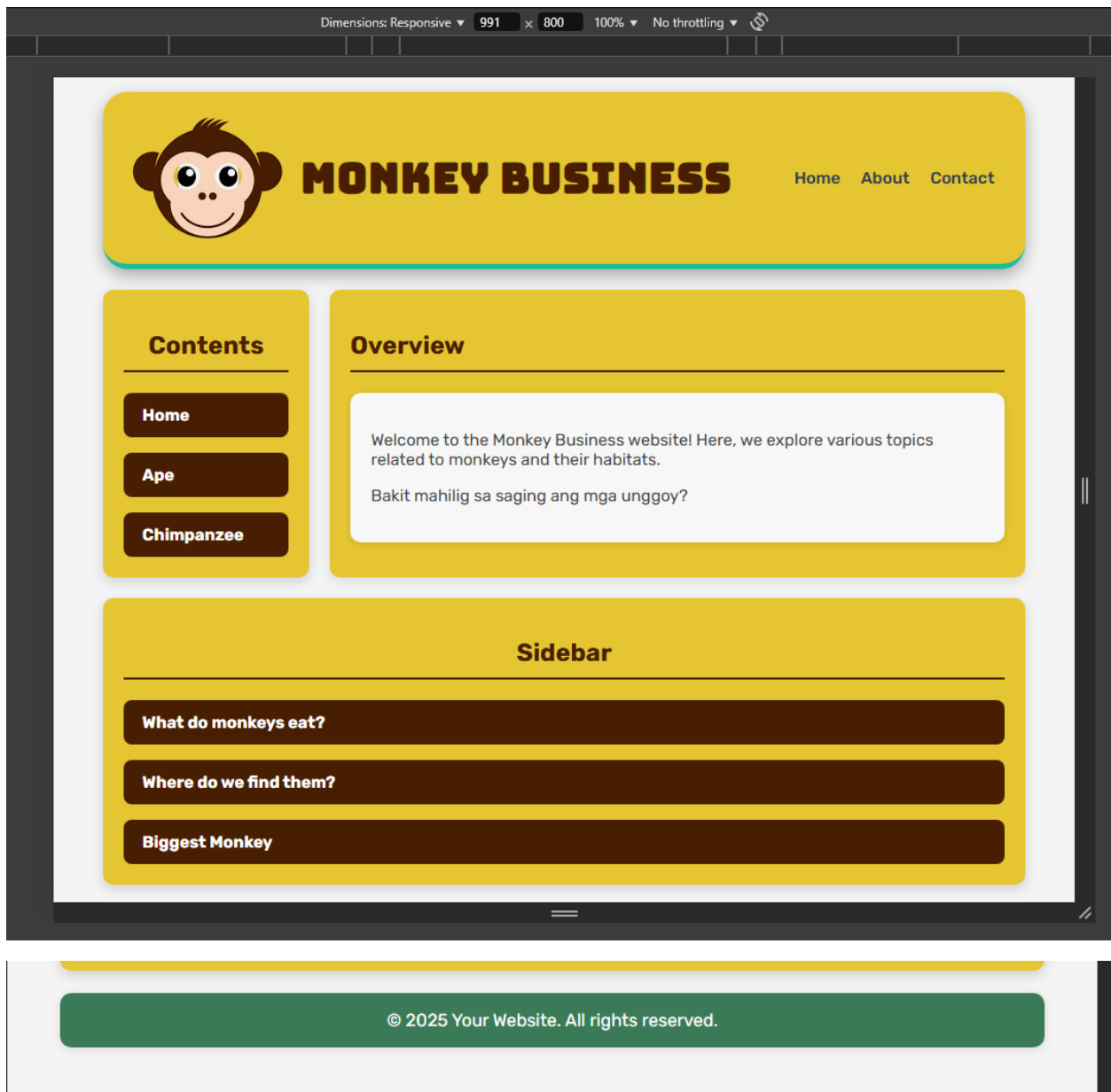
```
.logo {  
  margin:20px  
}  
  
body {  
  padding: 10px;  
}  
  
.main-header h1 {  
  font-size: 2rem;  
}  
}
```

OUTPUT

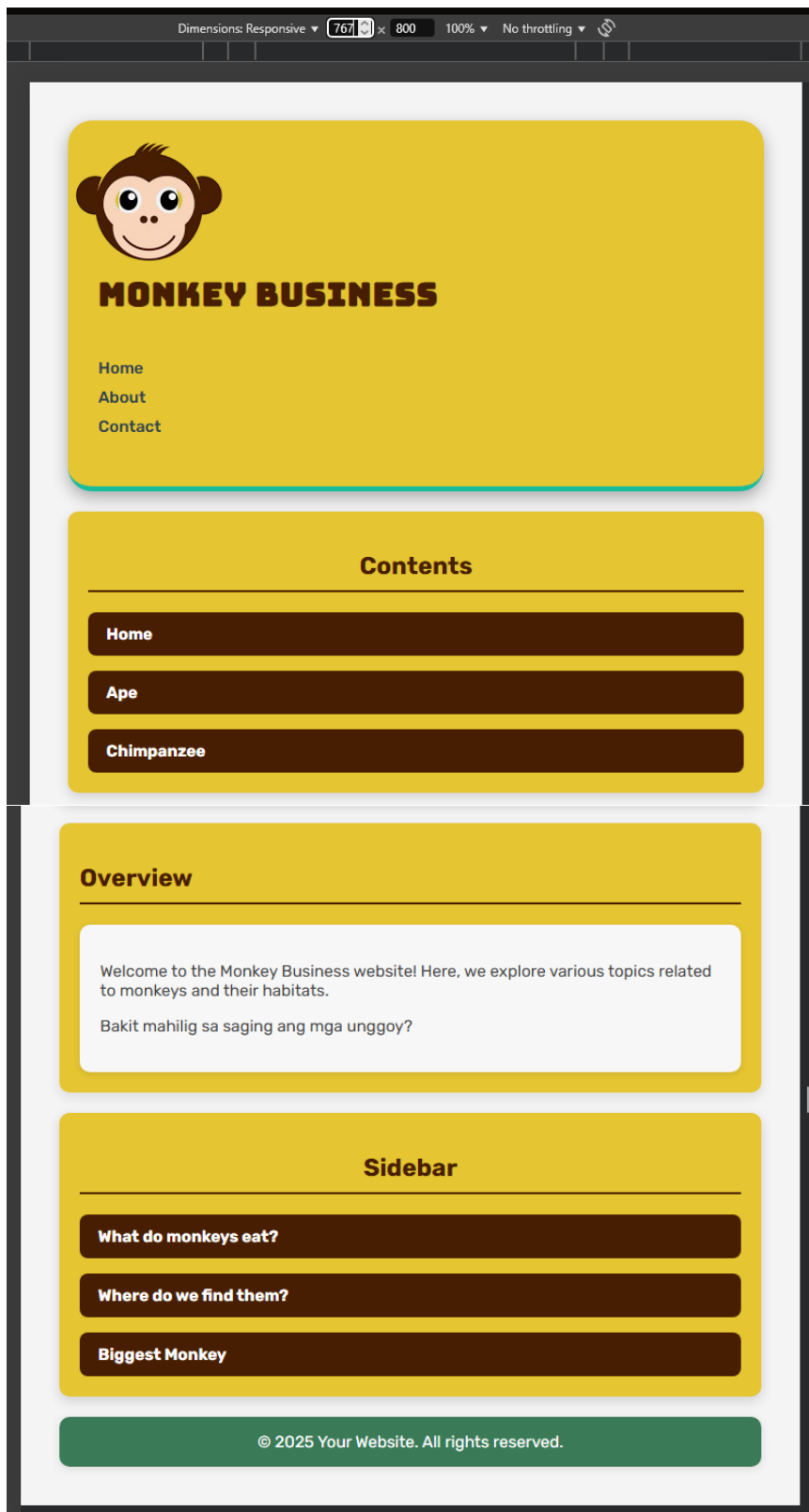
MY SCREEN



991 X 800



767 x 800



REFLECTION

In building this responsive webpage layout, I used CSS Grid to structure the main layout because it allowed me to define clear and organized areas like the header, navigation, main content, sidebar, and footer. I chose a three-column grid template for larger screens, with the left side reserved for the vertical nav, the center for the main content, and the right for the sidebar. I liked how grid-template-areas made the structure easy to understand and manage. To make the layout responsive, I applied media queries that adjusted the layout based on screen width. For example, below 992px, the sidebar automatically stacks below the main content, and for screens under 768px, everything stacks vertically in a single-column layout — from the header down to the footer. I also used Flexbox specifically in the header to align the logo on the left and the navigation links on the right. Flexbox helped a lot in keeping the header layout clean, even when the screen resized.

In terms of styling, I grouped related selectors (like .side-nav and .sidebar) together to avoid repeating code and to keep everything clean. I also added hover effects, shadows, and padding to give the layout a polished look. What I found interesting during this project was how flexible CSS Grid and Flexbox can be when combined. I did have some struggles at first with making the layout behave the way I wanted at different breakpoints, especially with stacking order, but once I clearly defined the grid areas inside each media query, it worked smoothly. Although, I had trouble making the main title and the logo to sit beside each other when its below 768px. Overall, I learned a lot about responsive design — especially how to control layout behavior using only CSS, without needing JavaScript. I'm really happy with how the layout turned out both on desktop and mobile views.