Assignment Code: FSD-AG-003

HTML & CSS | **Assignment**

Instructions: Carefully read each question. Use Google Docs, Microsoft Word, or a similar tool to create a document where you type out each question along with its answer. Save the document as a PDF, and then upload it to the LMS. Please do not zip or archive the files before uploading them. Each question carries 20 marks.

Total Marks: 160

Question 1 : Create a layout with three boxes side by side using Flexbox. The first box should have twice the width of the other two. Provide the HTML and CSS code snippets.

Answer:

```
<!-- HTML -->

<div class="container">

  <div class="box box1">Box 1</div>

  <div class="box box2">Box 2</div>

  <div class="box box3">Box 3</div>
</div>




/* CSS */
.container {
  display: flex;
  gap: 10px; /* optional spacing between boxes */
}

.box {
  background-color: lightblue;
  padding: 20px;
  text-align: center;
  border: 1px solid #333;
}

/* First box twice the width of others */
.box1 {
  flex: 2;
}

.box2,
.box3 {
  flex: 1;
}
```

Question 2: Using CSS Grid, create a 2x2 grid layout where the first row spans two columns, and the second row has two equal columns. Provide the HTML and CSS code snippets.

Answer:

```
<!-- HTML -->
<div class="grid-container">
  <div class="item item1">Item 1 (spans 2 columns)</div>
  <div class="item item2">Item 2</div>
  <div class="item item3">Item 3</div>
</div>



/* CSS */
.grid-container {
  display: grid;
  grid-template-columns: 1fr 1fr; /* two equal columns */
  grid-template-rows: auto auto; /* two rows */
  gap: 10px;
}

.item1 {
  grid-column: 1 / span 2; /* spans both columns */
  background-color: #8ecae6;
  padding: 20px;
  text-align: center;
}

.item2, .item3 {
  background-color: #219ebc;
  padding: 20px;
  text-align: center;
}
```

Question 3: Write a media query that changes the background color of the body to light grey when the screen width is less than 600px.

Answer:

Media query that changes the background color of the body to light grey when the screen width is less than 600px

```
@media (max-width: 599px) {

  body {

    background-color: lightgrey;

  }
}
```

Question 4: Create a button that changes its background color to blue and scales up slightly when hovered over, using CSS transitions. Provide the HTML and CSS code snippets.

Answer:

```
<!-- HTML -->
<button class="animated-btn">Hover Me</button>


/* CSS */
.animated-btn {
  background-color: #3498db;
  color: white;
  border: none;
  padding: 12px 24px;
  font-size: 16px;
  border-radius: 6px;
  cursor: pointer;
  transition: background-color 0.3s ease, transform 0.3s ease;
}

.animated-btn:hover {
  background-color: #1e90ff;
  transform: scale(1.05);
}
```

Question 5: Use the **::before** pseudo-element to add a decorative icon before each list item in an unordered list with the class name '**decorated-list**'. Provide the HTML and CSS code snippets.

Answer:

```
<!-- HTML -->
<ul class="decorated-list">
  <li>First item</li>
  <li>Second item</li>
  <li>Third item</li>
</ul>



/* CSS */
.decorated-list li {
  position: relative; /* for positioning the ::before element */
  padding-left: 25px; /* space for the icon */
}

.decorated-list li::before {
  content: "✓"; /* Unicode checkmark icon, you can change this */
  position: absolute;
  left: 0;
  top: 50%;
  transform: translateY(-50%);
  color: #2a9d8f;
  font-weight: bold;
  font-size: 16px;
}
```

Question 6: Create a simple animation that moves a square box from left to right infinitely. Provide the HTML and CSS code snippets.

Answer:

```
<!-- HTML -->
<div class="moving-box"></div>



/* CSS */
.moving-box {
  width: 50px;
  height: 50px;
  background-color: #e76f51;
  position: relative;
  animation: moveRight 3s linear infinite;
}

@keyframes moveRight {
  0% {
    left: 0;
  }
  50% {
    left: 200px;
  }
  100% {
    left: 0;
  }
}
```

Question 7: Explain the difference between **position: absolute;** and **position: fixed;** in CSS.

Answer:

position: absolute;

- Positioning context: The element is positioned relative to its nearest positioned ancestor (an ancestor with position set to relative, absolute, or fixed). If no such ancestor exists, it’s positioned relative to the initial containing block (usually the <html> element or viewport).
- Behavior: The element moves as you scroll the page because it’s tied to the document flow via its ancestor.
- Use case: Useful for positioning elements inside a container or layout, e.g., dropdown menus, tooltips inside a section.

position: fixed;

- Positioning context: The element is positioned relative to the viewport (the visible part of the browser window), regardless of any ancestors.
- Behavior: The element stays fixed in the same position on the screen even when the page is scrolled.
- Use case: Commonly used for sticky headers, floating buttons, or any UI element that must stay visible at all times.

Question 8: Create a layered layout where a semi-transparent overlay covers an image using **z-index**. Provide the HTML and CSS code snippets.

Answer:

```
<!-- HTML -->
<div class="image-container">
  <img src="https://via.placeholder.com/400x250" alt="Sample Image" />
  <div class="overlay"></div>
</div>



/* CSS */
.image-container {
  position: relative;
  width: 400px; /* same as image width */
  height: 250px; /* same as image height */
}

.image-container img {
  display: block;
  width: 100%;
  height: 100%;
  object-fit: cover;
}

.overlay {
  position: absolute;
  top: 0;
  left: 0;
  width: 100%;
  height: 100%;
  background-color: rgba(0, 0, 0, 0.5); /* semi-transparent black */
  z-index: 1; /* higher than image */
}
```