START
YOUR TECH JOURNEY
WITH ADA

Agenda

➢Grid
➢Shadow
➢Transition
➢Animations
➢Variables
➢Media Queries

Grid

The display: grid property creates a grid-based layout with rows and columns.

```css
.container {
  display: grid;
  grid-template-columns: 1fr 2fr; /* Two columns: first 1x, second 2x width */
  grid-template-rows: auto; /* Automatic row height */
  gap: 10px; /* Space between grid items */
}
.item {
  background-color: lightgray;
  padding: 10px;
}
```

Shadow

CSS shadow properties add shadow effects to elements. Common properties:
- box-shadow: For element shadows.
- text-shadow: For text shadows.

```css
.box {
  box-shadow: 5px 5px 10px rgba(0, 0, 0, 0.5); /* Box shadow */
}
.text {
  text-shadow: 2px 2px 5px gray; /* Text shadow */
}
```

Transition

CSS transitions allow smooth changes between property values. Use transition to define what property to animate and its duration.

```css
.button {
  background-color: lightblue;
  transition: background-color 0.3s ease;
}
.button:hover {
  background-color: darkblue; /* Changes smoothly */
}
```

Animations

CSS animations define keyframes to animate properties. Use @keyframes to create animations and animation to apply them.

```css
@keyframes slide {
  from {
    transform: translateX(0);
  }
  to {
    transform: translateX(100px);
  }
}
.box {
  animation: slide 2s infinite; /* Infinite sliding animation */
}
```

Variables

CSS variables (--variable-name) store reusable values for colors, sizes, and other properties.

```css
:root {
  --primary-color: lightblue;
  --padding-size: 10px;
}
.box {
  background-color: var(--primary-color);
  padding: var(--padding-size);
}
```

Media Queries

Media queries allow for responsive designs that adapt to screen size and resolution.

```css
/* Mobile devices */
@media (max-width: 600px) {
  body {
    background-color: lightgray;
  }
}
/* Larger screens */
@media (min-width: 601px) {
  body {
    background-color: white;
  }
}
```