**GreenSock Animation Platform (GSAP)**

**Introduction**

GSAP (GreenSock Animation Platform) is a robust JavaScript library used for creating high-performance animations. It provides a suite of tools and features to create complex animations with ease, offering precision and flexibility.

**Installation**

**Via CDN**

Include GSAP via a CDN in your HTML file:

<script src="https://cdnjs.cloudflare.com/ajax/libs/gsap/3.11.0/gsap.min.js"></script>

**Via npm/yarn**

npm install gsap

# or

yarn add gsap

**Basic Usage**

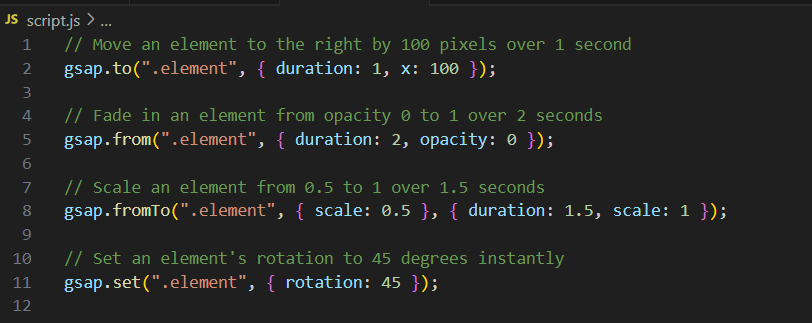
To create a simple animation using GSAP, you can use the **gsap.to** method:

**gsap.to(".element", { duration: 1, x: 100 });**

**Core Methods**

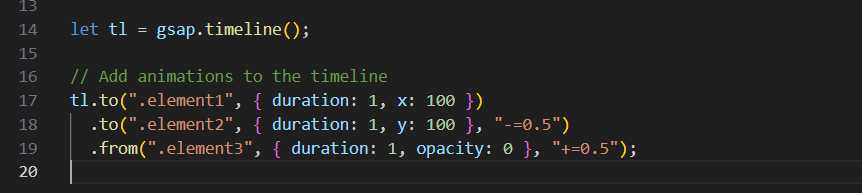
**GSAP offers several core methods to create and control animations:**

* **gsap.to(targets, vars): Animates the target elements to the specified properties.**
* **gsap.from(targets, vars): Animates the target elements from the specified properties.**
* **gsap.fromTo(targets, fromVars, toVars): Animates the target elements from specific properties to another set of properties.**
* **gsap.set(targets, vars): Sets the target elements to the specified properties immediately.**

****

**Timelines**

**Timelines in GSAP allow for more complex sequencing and control of multiple animations.**

****

**Plugins**

GSAP (GreenSock Animation Platform) offers a variety of plugins that extend its capabilities, making it easier to create complex animations and interactions. Here's an overview of some of the most popular GSAP plugins:

**ScrollTrigger**

**Purpose:** Animates elements based on the user's scroll position.

**Features:**

* Trigger animations when elements enter the viewport.
* Pin elements in place while the user scrolls.
* Create scroll-based animations and effects.



**ScrollTrigger Properties:**

**trigger:** The element that triggers the animation when it enters the viewport.

**start**: Defines when the animation should start relative to the viewport and the trigger element.

* Format: **"trigger point viewport point"**
* Default: **"top bottom"**

**end**: Defines when the animation should end.

* Format: Same as **start**.
* Default: **"bottom top"**

**scrub**: Links the progress of the animation to the scrollbar, creating a smoother animation effect.

* Accepts **true** or a duration (in seconds).

**markers**: Displays visual markers for debugging the start and end points of the trigger.

**pin**: Pins an element in place while the scroll position is between the **start** and **end** points.

* Accepts **true** or an element.

**pinSpacing**: Controls whether the pinned element creates space in the layout.

* Accepts **true**, **false**, or a string (**"margin"**, **"padding"**, **"none"**).

**toggleClass**: Adds/removes a class to/from an element when the scroll position reaches the start/end points.

* Format: **{targets: element, className: string}**

**animation**: Links a GSAP animation to the ScrollTrigger, making the animation play when the trigger is activated.

**onEnter**: A callback function that is called when the trigger element enters the viewport.

**onLeave**: A callback function that is called when the trigger element leaves the viewport.

**onEnterBack**: A callback function that is called when the trigger element re-enters the viewport from the bottom.

**onLeaveBack**: A callback function that is called when the trigger element leaves the viewport from the top.

**scroller**: Specifies the element that should be used for scrolling (useful for custom scroll containers).

**horizontal**: Indicates whether the ScrollTrigger should track horizontal scrolling instead of vertical.

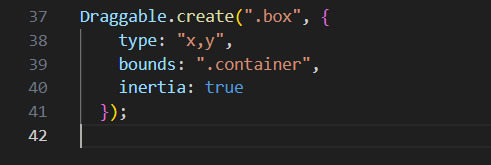
**containerAnimation**: Allows ScrollTrigger to be used inside a container that is being animated by GSAP.

**Draggable**

**Purpose**: Enables dragging functionality for elements.

**Features**:

* Make any element draggable.
* Constrain dragging to certain bounds or axis.
* Throwing and inertia support.



**MotionPathPlugin**

**Purpose**: Animates elements along a specified path.

**Features**:

* Animate elements along SVG paths or custom bezier curves.
* Precise control over path progression and orientation.

