

# GSAP Animation

UNDERSTANDING THE  
GREENSOCK  
ANIMATION PLATFORM





# Cos'è GSAP?

**GSAP STA PER GREENSOCK ANIMATION PLATFORM.**

**È UNA LIBRERIA JAVASCRIPT ROBUSTA PER CREARE ANIMAZIONI AD ALTE PRESTAZIONI.**

**GSAP PUÒ ANIMARE QUALSIASI PROPRIETÀ NUMERICA DI QUALSIASI OGGETTO JAVASCRIPT, NON SOLO PROPRIETÀ CSS.**





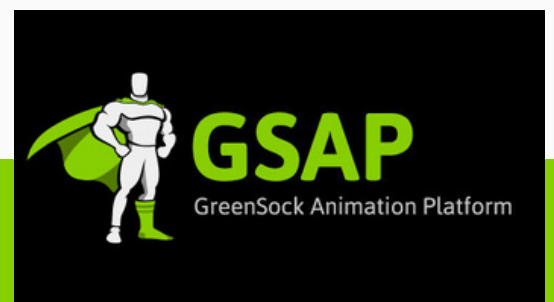
# Perché usare GSAP?

**ALTE PRESTAZIONI: OTTIMIZZATO PER VELOCITÀ ED EFFICIENZA.**

**COMPATIBILITÀ TRA BROWSER: FUNZIONA SU TUTTI I BROWSER MODERNI.**

**AMPIA GAMMA DI FUNZIONALITÀ: FORNISCE UN CONTROLLO PRECISO SULLE ANIMAZIONI.**

**FACILITÀ D'USO: SINTASSI SEMPLICE E DOCUMENTAZIONE COMPLETA.**



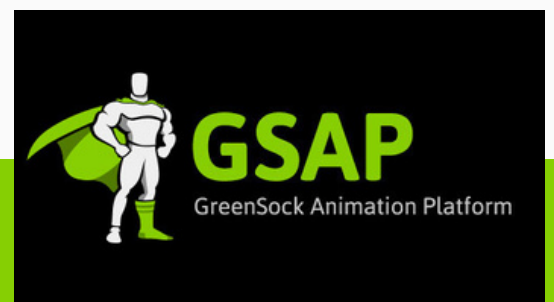


# gsap.to()

GENERA UN'ANIMAZIONE PER L'ELEMENTO TARGET A PARTIRE DALLE PROPERITÀ INIZIALI DELL'ELEMENTO

```
gsap.to(element, {duration: 1, x: 100, y: 50, opacity: 1});
```

L'ELEMENTO TARGET NELL'ARCO DI UN SECONDO SARÀ SPOSTATO DI 100PX SULL'ASSE X E DI 50 PX SULL'ASSE Y E LA SUA OPACITÀ DIVENTERÀ 100%



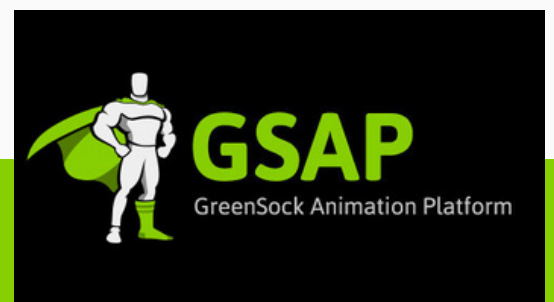


## gsap.from()

GENERA UN'ANIMAZIONE PER L'ELEMENTO TARGET A PARTIRE DAI VALORI FORNITI DA GSAP FINO AD ARRIVARE ALLA POSIZIONE ORIGINALE DELL'ELEMENTO

```
gsap.from(element, {duration: 1, x: -100, opacity: 0});
```

L'ELEMENTO TARGET NELL'ARCO DI UN SECONDO PARTIRÀ DA -100PX SULL'ASSE X E CON UNA OPACITÀ DELLO 0% E ARRIVERÀ AI VALORI DESCRITTI DAL CSS



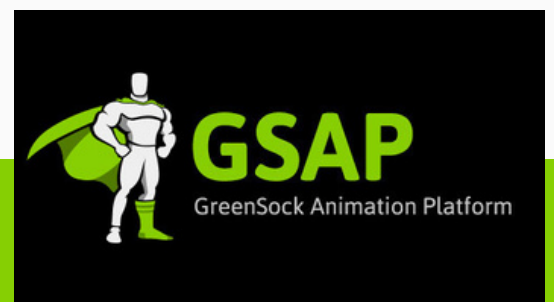


# gsap.fromTo()

GENERA UN'ANIMAZIONE PER L'ELEMENTO TARGET FORNENDO I PARAMETRI DI INIZIO E FINE

```
gsap.fromTo(element, {x: -100, opacity: 0}, {duration: 1, x: 100, opacity: 1});
```

L'ELEMENTO TARGET NELL'ARCO DI UN SECONDO PARTIRÀ DA -100PX SULL'ASSE X E CON UNA OPACITÀ DELLO 0% E ARRIVERÀ A +100PX SULL'ASSE X E UNA OPACITÀ DEL 100%





# Duration

- TIPO: NUMBER
- DESCRIZIONE: DURATA DELL'ANIMAZIONE IN SECONDI.

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



# Delay

- TIPO: NUMBER
- DESCRIZIONE: QUANTITÀ DI RITARDO PRIMA CHE L'ANIMAZIONE INIZI, IN SECONDI.

```
gsap.to("#element", {duration: 1, delay: 0.5, x: 100});
```





# Ease

- TIPO: STRING | FUNCTION
- DESCRIZIONE: FUNZIONE DI EASING PER CONTROLLARE LA VELOCITÀ DEL CAMBIAMENTO DURANTE L'ANIMAZIONE.

```
gsap.to("#element", {duration: 1, ease: "power1.inOut", x: 100});
```

- [HTTPS://GSAP.COM/DOCS/V3/EASES/](https://gsap.com/docs/v3/eases/)



x, y, z

- TIPO: NUMBER | STRING
- DESCRIZIONE: VALORI FINALI PER LE PROPRIETÀ X, Y E Z.

```
gsap.to("#element", {duration: 1, x: 100, y: 50});
```



# opacity

- TIPO: NUMBER
- DESCRIZIONE: VALORE FINALE PER LA PROPRIETÀ DI OPACITÀ.

```
gsap.to("#element", {duration: 1, opacity: 0.5});
```



# scale, scaleX, scaleY

- TIPO: NUMBER
- DESCRIZIONE: VALORI FINALI PER SCALARE L'ELEMENTO.

```
gsap.to("#element", {duration: 1, scale: 1.5});
```



# rotation, rotationX, rotationY

- TIPO: NUMBER | STRING
- DESCRIZIONE: VALORI FINALI PER LA ROTAZIONE IN GRADI

```
gsap.to("#element", {duration: 1, rotation: 360});
```



# transformOrigin

- TIPO: STRING
- DESCRIZIONE: IMPOSTA L'ORIGINE DELLE TRASFORMAZIONI.

```
gsap.to("#element", {duration: 1, rotation: 360, transformOrigin: "center center"});
```

- PARAMETRI: CENTER CENTER, TOP LEFT, TOP CENTER, TOP RIGHT E COSÌ VIA...



## backgroundColor, color

- TIPO: STRING
- DESCRIZIONE: VALORI FINALI PER IL COLORE DI SFONDO E IL COLORE DEL TESTO.

```
gsap.to("#element", {duration: 1, backgroundColor: "#ff0000", color: "#ffffff"});
```



# repeat

- TIPO: NUMBER
- DESCRIZIONE: NUMERO DI VOLTE CHE L'ANIMAZIONE DEVE RIPETERSI. UTILIZZA -1 PER RIPETIZIONI INFINITE.

```
gsap.to("#element", {duration: 1, x: 100, repeat: 2});
```





# yoyo

- TIPO: BOOLEAN
- DESCRIZIONE: SE TRUE, L'ANIMAZIONE INVERTIRÀ DIREZIONE A OGNI RIPETIZIONE

```
gsap.to("#element", {duration: 1, x: 100, repeat: 2, yoyo: true});
```



# stagger

- TIPO: NUMBER | OBJECT
- DESCRIZIONE: RITARDA I TEMPI DI INIZIO DELL'ANIMAZIONE DI PIÙ ELEMENTI.

```
gsap.to(".box", {duration: 1, x: 100, stagger: 0.2});
```

- [HTTPS://GSAP.COM/RESOURCES/GETTING-STARTED/STAGGERS/](https://gsap.com/resources/getting-started/staggers/)



# onComplete, onStart, onUpdate, onRepeat, onReverseComplete

- TIPO: FUNCTION
- DESCRIZIONE: CALLBACK PER VARI STADI DELL'ANIMAZIONE.

```
gsap.to("#element", {duration: 1, x: 100, onComplete: () => console.log("Animazione completata")});
```



# scrollTrigger

- TIPO: OBJECT
- DESCRIZIONE: SINCRONIZZA L'ANIMAZIONE CON LA POSIZIONE DI SCORRIMENTO.

```
gsap.to("#element", {  
  duration: 1,  
  x: 100,  
  scrollTrigger: {  
    trigger: "#element",  
    start: "top center",  
    end: "bottom top",  
    scrub: true  
  }  
});
```

[HTTPS://CODEPEN.IO/GRE  
ENSOCK/PEN/ZYMPMGN](https://codepen.io/greentsock/pen/zympmgn)



# xPercent, yPercent

- TIPO: NUMBER
- DESCRIZIONE: SPOSTA L'ELEMENTO DI UNA PERCENTUALE DELLA SUA LARGHEZZA O ALTEZZA.

```
gsap.to("#element", {duration: 1, xPercent: 50, yPercent: 50});
```



# skewX, skewY

- TIPO: NUMBER
- DESCRIZIONE: INCLINA L'ELEMENTO LUNGO L'ASSE X O Y.

```
gsap.to("#element", {duration: 1, skewX: 45, skewY: 45});
```



# padding, paddingLeft, paddingRight, paddingTop, paddingBottom

- TIPO: NUMBER | STRING
- DESCRIZIONE: ANIMA IL PADDING DI UN ELEMENTO.

```
gsap.to("#element", {duration: 1, padding: "20px"});
```



# borderWidth, borderColor, borderRadius

- TIPO: NUMBER | STRING
- DESCRIZIONE: ANIMA LE PROPRIETÀ DEL BORDO.

```
gsap.to("#element", {duration: 1, borderWidth: "10px", borderColor: "#000000", borderRadius: "50%"});
```





## width, height

- TIPO: NUMBER | STRING
- DESCRIZIONE: ANIMA LA LARGHEZZA E L'ALTEZZA DI UN ELEMENTO.

```
gsap.to("#element", {duration: 1, width: "200px", height: "100px"});
```



# SCROLLTRIGGER approfondimento

SCROLL TRIGGER VA IMPORTATO A PARTE IN QUANTO PLUGIN DI GSAP

E VA POI ATTIVATO

```
gsap.registerPlugin(ScrollTrigger);
```





# SCROLLTRIGGER approfondimento

```
gsap.to("#element", {  
  scrollTrigger: {  
    trigger: "#element",  
    start: "top center",  
    end: "bottom 100px",  
    scrub: true  
  },  
  x: 500  
});
```

- TRIGGER: L'ELEMENTO CHE ATTIVERÀ L'ANIMAZIONE.
- START: LA POSIZIONE DI INIZIO DELL'ANIMAZIONE.
- END: LA POSIZIONE DI FINE DELL'ANIMAZIONE.
- SCRUB: SINCRONIZZA L'ANIMAZIONE CON LA POSIZIONE DI SCORRIMENTO.



# trigger

- TIPO: STRING | ELEMENT
- DESCRIZIONE: L'ELEMENTO CHE ATTIVERÀ LO SCROLLTRIGGER.

```
trigger: "#element"
```



# start

- TIPO: STRING | NUMBER | FUNCTION
- DESCRIZIONE: LA POSIZIONE DI INIZIO DELL'ANIMAZIONE RELATIVA AL TRIGGER.

```
start: "top center"
```



end

- TIPO: STRING | NUMBER | FUNCTION
- DESCRIZIONE: LA POSIZIONE DI FINE DELL'ANIMAZIONE RELATIVA AL TRIGGER.

```
end: "bottom 100px"
```



# markers

- TIPO: BOOLEAN | OBJECT
- DESCRIZIONE: MOSTRA I MARCATORI PER IL DEBUG.

```
markers: true
```



## pin

- TIPO: BOOLEAN | STRING | ELEMENT
- DESCRIZIONE: FISSA L'ELEMENTO NEL VIEWPORT DURANTE L'ANIMAZIONE.

```
pin: true
```

- PINSPACING
- TIPO: BOOLEAN | STRING
- DESCRIZIONE: CONTROLLA LO SPAZIO LASCIATO MENTRE UN ELEMENTO È PINNING. SE FALSE, NON LASCIA SPAZIO.

```
pinSpacing: false
```





# horizontal

- TIPO: BOOLEAN
- DESCRIZIONE: SE TRUE, SCROLLTRIGGER RISPONDE ALLO SCORRIMENTO ORIZZONTALE INVECE CHE VERTICALE.

```
horizontal: true
```



# horizontal

- TIPO: BOOLEAN
- DESCRIZIONE: SE TRUE, SCROLLTRIGGER RISPONDE ALLO SCORRIMENTO ORIZZONTALE INVECE CHE VERTICALE.

```
horizontal: true
```

