

4주차 GreenSock 기초 개념

1. GreenSock : <https://greensock.com/>

- 복잡한 Animation 효과를 몇 가지 명령어를 사용하여 쉽게 구현할 수 있도록 도와주는 자바스크립트 라이브러리
- 사용 방법은 공식 홈페이지에서 GreenSock 라이브러리를 다운로드하여 HTML 문서와 연결하거나 아래 CDN 을 사용하여 웹상에 있는 라이브러리에 접속하여 이용할 수 있음
- latest 는 항상 최신 버전을 불러오겠다는 의미
- `<script src="http://cdnjs.cloudflare.com/ajax/libs/gsap/latest/TweenLite.min.js"></script>`
- `<script src="http://cdnjs.cloudflare.com/ajax/libs/gsap/latest/plugins/CSSPlugin.min.js"></script>`

2. TweenLite 기초 개념

```
<div id="demo">
  <div class="logo"></div>
</div>

#demo {
  width: 100%;
  height: 300px;
  background-color: #333;
  padding: 8px;
}

.logo {
  position: relative;
  width: 200px;
  height: 200px;
  background: yellow url(..img/icon.png) no-repeat;
  border: solid 10px red;
}

$(function() {

  // TweenLite.to(target, duration, {vars});
  TweenLite.to(".logo", 10, {
    left:600,
    backgroundColor: "F00",
    borderColor: "white",
    borderRadius: "50%"
  });

});
```

```
$(function() {  
  
    TweenLite.to(".logo", 10, {  
        x: 600,  
        rotation: 360,  
        scale: 0.5  
    });  
  
});
```

```
$(function() {  
  
    TweenLite.from(".logo", 10, {  
        x: 600,  
        rotation: 360,  
        scale: 0.5  
    });  
  
});
```

```
$(function() {  
  
    TweenLite.from(".logo", 10, {  
        opacity: 0,  
        scale: 0  
    });  
  
});
```

3. Selectors : 선택된 header 영역에 opacity : 0 + y 축으로 50 만큼 1초 동안 움직이는 효과 적용

```
$(function() {  
  
    var introHeader = $('.intro-heading');  
    // TweenLite.to(영역, 시간, {효과});  
    // introHeader 의 현재 디자인에서 1초 안에 opacity: 0, y 축으로 50px 만큼 변화  
    TweenLite.to(introHeader, 1, {opacity: 0, y: 50});  
  
});
```

4. TweenLite 의 기본적인 특징 : .to , .from , .fromTo

```
$(function() {  
  
    // 버튼 선택  
    var btnXl = $('.btn-xl');  
  
    // 크기 조절  
    TweenLite.to(btnXl, 1, {width: 100});  
  
    // x 축으로 200px 만큼 이동  
    TweenLite.to(btnXl, 1, {x: 200});  
  
    // x 축 -200px 지점에서 현재 위치까지 이동  
    TweenLite.from(btnXl, 1, {x: -200});  
  
    // x 축 -200px 지점에서 200px 지점까지 이동  
    TweenLite.fromTo(btnXl, 1, {x: -200}, {x: 200});  
  
    // x 축 -200px 지점에 선택된 오브젝트 고정  
    TweenLite.set(btnXl, {x: -200});  
  
    // IT'S NICE TO MEET YOU 선택  
    var introHeader = $('.intro-heading');  
  
    // 텍스트에 투명도 적용  
    TweenLite.from(introHeader, 1, {autoAlpha: 0});  
  
    // delay 적용  
    TweenLite.from(introHeader, 1, {autoAlpha: 0, delay: 1});  
  
});
```

5. Easing : CSS Animation 의 Timing Function 역할을 수행

- Easing CDN : <https://cdnjs.com/libraries/gsap>
- `<script src="https://cdnjs.cloudflare.com/ajax/libs/gsap/1.18.2/easing/EasePack.min.js"></script>`
- Easing Visualizer : <http://greensock.com/ease-visualizer>

```
$(function() {  
  
    var faStack = $('.fa-stack');  
  
    TweenLite.from(faStack, 1, {x: -200, ease: Power0.easeNone});  
  
    // TweenLite.from(faStack, 1, {x: -200, ease: Power1.easeIn});  
  
    // TweenLite.from(faStack, 1, {x: -200, ease: Power2.easeOut});  
  
    // TweenLite.from(faStack, 1, {x: -200, ease: Back.easeOut});  
    // TweenLite.from(faStack, 1, {x: -200, ease: Bounce.easeOut});  
  
    // TweenLite.from(faStack, 1, {x: -200, ease: Elastic.easeOut});  
    // TweenLite.from(faStack, 1, {x: -200, ease: RoughEase.ease});  
    // TweenLite.from(faStack, 1, {x: -200, ease: SlowMo.ease});  
  
    var serviceHeading = $('.service-heading');  
  
    TweenLite.from(serviceHeading, 1, {autoAlpha: 0, delay: 1});  
  
});
```

6. 복수의 영역에 애니메이션 효과 적용하기

```
$(function() {  
  
    var portfolioHeading = $('#portfolio .section-heading'),  
        portfolioSubHeading = $('#portfolio .section-subheading'),  
        portfolioImg = $('#portfolio img'),  
        portfolioCaptionTitle = $('#portfolio .portfolio-caption h4'),  
        portfolioCaptionText = $('#portfolio .portfolio-caption p');  
  
    TweenLite.from(portfolioHeading, 0.3, {y: -15, autoAlpha: 0, delay: 0.2});  
    TweenLite.from(portfolioSubHeading, 0.3, {y: -15, autoAlpha: 0, delay: 0.4});  
    TweenLite.from(portfolioImg, 0.3, {y: -15, autoAlpha: 0, delay: 0.8});  
    TweenLite.from(portfolioCaptionTitle, 0.3, {y: -15, autoAlpha: 0, delay: 0.8});  
    TweenLite.from(portfolioCaptionText, 0.3, {y: -15, autoAlpha: 0, delay: 1});  
  
});
```

<!-- 복수의 애니메이션 효과를 주기 위해서 TimelineLite 사용 -->

```
$(function() {  
  
    var portfolioHeading = $('#portfolio .section-heading'),  
        portfolioSubHeading = $('#portfolio .section-subheading'),  
        portfolioImg = $('#portfolio img'),  
        portfolioCaptionTitle = $('#portfolio .portfolio-caption h4'),  
        portfolioCaptionText = $('#portfolio .portfolio-caption p'),  
  
    tl = new TimelineLite();  
  
    // 여기에서 delay 는 이전 효과를 기준으로 측정  
    tl  
        .from(portfolioHeading, 0.3, {y: -15, autoAlpha: 0, delay: 0.2})  
        .from(portfolioSubHeading, 0.3, {y: -15, autoAlpha: 0, delay: 0.4})  
        .from(portfolioImg, 0.3, {y: -15, autoAlpha: 0, delay: 0.8})  
        .from(portfolioCaptionTitle, 0.3, {y: -15, autoAlpha: 0, delay: 0.8})  
        .from(portfolioCaptionText, 0.3, {y: -15, autoAlpha: 0, delay: 1});  
  
});
```

- Position Parameter : <http://greensock.com/position-parameter>

```
$(function() {  
  
    .....  
  
    tl  
        .from(portfolioHeading, 0.3, {y: -15, autoAlpha: 0})  
        .from(portfoliidSubHeading, 0.3, {y: -15, autoAlpha: 0}, '-=0.15')  
        .from(portfolioImg, 0.3, {y: -15, autoAlpha: 0}, '-=0.15')  
        .from(portfolioCaptionTitle, 0.3, {y: -15, autoAlpha: 0}, '-=0.15')  
        .from(portfolioCaptionText, 0.3, {y: -15, autoAlpha: 0}, '-=0.15');  
  
});
```

7. 선택된 영역의 애니메이션 동작에 맞춰 효과 적용하기

```
$(function() {  
  
    ...  
  
    t1  
        .from(portfolioHeading, 0.3, {y: -15, autoAlpha: 0})  
        .from(portfolidSubHeading, 0.3, {y: -15, autoAlpha: 0}, '+=1')  
  
    // 절대시간 기준  
        .from(portfolioImg, 0.3, {y: -15, autoAlpha: 0}, 3)  
        .from(portfolioCaptionTitle, 0.3, {y: -15, autoAlpha: 0}, 2)  
        .from(portfolioCaptionText, 0.3, {y: -15, autoAlpha: 0}, 2.5);  
  
});
```

```
t1  
    .from(portfolioHeading, 0.3, {y: -15, autoAlpha: 0})  
  
    // SubHeading 을 기준으로 효과 적용  
    .add('portfolidSubHeading')  
    .from(portfolidSubHeading, 0.3, {y: -15, autoAlpha: 0})  
    .from(portfolioImg, 0.3, {y: -15, autoAlpha: 0}, 'portfolidSubHeading')  
    .from(portfolioCaptionTitle, 0.3, {y: -15, autoAlpha: 0}, 'portfolidSubHeading+=3')  
    .from(portfolioCaptionText, 0.3, {y: -15, autoAlpha: 0}, 'portfolidSubHeading');
```