

프로젝트 기술서

작성자: 류경현

테디베어 스크롤 후 숨/천/바늘로 변환

● CSS/JQuery 전 HTML 준비 코드

1. 테디베어의 코드를 작성합니다. 태그를 통해 테디베어 이미지를 삽입하고 자연스러운 이동 효과를 주기 위해서 를 감싸는 <div class="">를 클래스값과 함께 줍니다.
2. 도형영역에 숨/천/바늘 또한 태그를 통해 이미지를 삽입합니다.

```
<main>
<div class="bear_wrapper">
  
</div>
<!-- unique -->
<section class="unique_wrapper">
  <h2 class="title1">Museum is unique <span class="block">to each branch!</span></h2>
  <div class="deco">Out branches</div>
</section>

<!-- fabric -->
<section class="fabric_wrapper">
  <div class="bigText fabric">
    <span>Fabric</span>
    <p>마티와 함께 떠나는 세계문화여행기</p>
  </div>
  
  <div class="bigText cotton">
    <span>Cotton</span>
    <p>테디와 친구들의 전 세계 도시로의 시공간여행!</p>
  </div>
  
  <div class="bigText needle">
    <span>Needle</span>
    <p>테디월슨과 페르난도가 함께 떠나는 세계일주!</p>
  </div>
  
```

테디베어

도형영역

천

숨

바늘

● CSS/jQuery 이용하여 효과 주기

1. 테디베어를 스크롤 이동에 맞춰 떨어지게 합니다.

- 1. 먼저 테디베어를 원하는 위치에 css의 position 값을 이용하여 배치합니다.
- 2. \$(window).on('scroll', 효과함수)를 통해 먼저 스크롤 효과를 준비합니다.
- 3. 테디베어가 움직이고 사라지는 효과들의 기준들은 scrollY를 기준으로 합니다.
- 4. if문을 이용하여 조건문에는 테디베어를 움직일 만큼의 영역을 설정합니다.

```
// 핑크곰 떨어뜨리기
const bearMove = document.querySelector('.unique_wrapper').
clientHeight - 240;
$(window).on('scroll', () => { //Query 스크롤
  console.log(scrollY);
  if (0 < scrollY && scrollY < bearMove + 120) { //테디베어 이동영역
    $('.bear_wrapper').css({
      top: `10%`,
      transform: `translate(-50%, ${scrollY}px)`; //테디베어 Y축이동거리
    });
  } else {
    $('.bear_wrapper').css({
      top: `10%`,
    });
  }
});
```

- 5. if문의 실행문에는 테디베어 태그를 감싸는 <div>에 css를 넣어줍니다.
- 6. <div>태그에 transform: translateY(\${px})를 넣어서 Y축으로 움직일 준비를 합니다.
- 7. 테디베어가 스크롤한 만큼 움직이기 위해 변수란에는 scrollY를 넣어줍니다.

2. 테디베어를 다음 도형영역의 위에서 사라지게 합니다.

- 1. 사라지게 할 위치를 설정해줍니다.
- 2. if문을 이용하여 조건문에는 scrollY가 특정위치보다 더 내려갈 경우 실행문이 실행되게 조건을 줍니다.
- 3. if문의 실행문에는 테디베어태그에 css값으로 opacity 값과 transform: scale의 값을 0으로 줍니다.
- 4. else의 값으로 1의 값을 주며 영역밖에서는 나타나게 합니다.
- 5. 영역에 따라 테디베어가 사라지고 나타나게 됩니다.

```
if (scrollY > bearMove) {
  // 곰 투명해지기
  $('.bear_wrapper .bear1').css({
    opacity: 0,
    transform: `scale(0)`; //줄어들고 사라지기
  });
} else {
  $('.bear_wrapper .bear1').css({
    opacity: 1,
    transform: `scale(1)`; //커지며 나타나기
  });
}
```

3. 도형들이 나타나고 텍스트들의 뒤로 붙게 합니다.

- 1. 도형들이 나타날 위치를 정합니다. 도형들에게 position값을 주어 도형영역의 상단에 세 도형들이 모두 한 곳에 위치할 수 있도록 합니다.
- 2. 스크롤이 특정지점을 지나가면 작동할 수 있도록 if문을 사용합니다.
- 3. if문의 조건문은 (scrollY > 특정지점)의 형식으로 쓰며 특정 지점이 지나가면 실행문으로 도형태그들에 active라는 클래스값이 붙게 해줍니다.
- 4. CSS에서 에 종속 선택자를 붙여 position의 top, left을 이용하여 스크롤했을 시 이동하고 싶은 위치로 지정해 줍니다. 그리고 작아 졌다가 커지는 효과를 주기 위해 위의 테디베어가 동일하게 opacity와 scale의 값을 CSS에 줍니다.

```
main .fabric_img,
main .cotton_img,
main .needle_img {
  position: absolute;
  z-index: -1;
  top: -120px;
  left: 50%;
  transform: translateX(-50%) scale(0);
  opacity: 0;
  transition: 1s;
}

main .fabric_img.active {
  opacity: 1;
  transform: translateX(-50%) scale(1);
  top: -1%;
  left: 43%;
}

main .cotton_img.active {
  opacity: 1;
  transform: translateX(-50%) scale(1);
  top: 35%;
  left: 71%;
}

main .needle_img.active {
  opacity: 1;
  transform: translateX(-50%) scale(1);
  top: 53%;
  left: 25%;
}
```

CSS

JavaScript

```
if (scrollY > bearMove) {
  // 곰 투명해지기
  $('.bear_wrapper .bear1').css({
    opacity: 0,
    transform: `scale(0)`;
  });
  // 도형 움직이기
  $('main .fabric_img').addClass('active');
  $('main .cotton_img').addClass('active');
  $('main .needle_img').addClass('active');
} else {
  $('.bear_wrapper .bear1').css({
    opacity: 1,
    transform: `scale(1)`;
  });
  $('main .fabric_img').removeClass('active');
  $('main .cotton_img').removeClass('active');
  $('main .needle_img').removeClass('active');
}
```

- 5. 기본 에는 0의 값을, .active가 있는 태그에는 1의 값을 줍니다.
- 6. if문의 else 값을 통해 active의 값이 클래스에서 제거되도록 해줍니다.
- 7. 스크롤에 따라 도형들은 특정 지점에서 도형영역의 원하는 곳으로 나타나고 사라지게 됩니다.