

## 01\_트랜지션기본.html

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>트랜지션</title>
7   <link rel="stylesheet" href="../css/main.css">
8   <style>
9     /* 1초 동안 투명도 변화 */
10    .v-enter-active, .v-leave-active {
11      transition: opacity 1s;
12    }
13    /* 더 이상 보이게 되지 않았을 때의 투명도 */
14    .v-enter, .v-leave-to {
15      opacity: 0;
16    }
17  </style>
18 </head>
19 <body>
20   <div id="app">
21     <p><button v-on:click="show=!show">변경</button></p>
22     <transition>
23       <div v-show="show">
24         트랜지션 요소
25       </div>
26     </transition>
27   </div>
28
29   <!-- Vue.js -->
30   <script
31     src='https://cdnjs.cloudflare.com/ajax/libs/vue/2.6.9/vue.min.js'></script>
32   <script>
33     var app = new Vue( {
34       el: '#app',
35       data: {
36         show: true
37       }
38     } );
39   </script>
40 </body>
</html>
```

## 02\_클래스이름과프리픽스.html

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>트랜지션</title>
7   <link rel="stylesheet" href="../css/main.css">
8   <style>
9     /* 1초 동안 투명도 변화 */
10    .gt-enter-active, .gt-leave-active {
11      transition: opacity 1s;
12    }
13    /* 더 이상 보이게 되지 않았을 때의 투명도 */
14    .gt-enter, .gt-leave-to {
15      opacity: 0;
16    }
17  </style>
18 </head>
19 <body>
20   <div id="app">
21     <p><button v-on:click="show=!show">변경</button></p>
22     <transition name="gt">
23       <div v-if="show">
24         트랜지션 요소
25       </div>
26     </transition>
27   </div>
28
29   <!-- Vue.js -->
30   <script
31     src='https://cdnjs.cloudflare.com/ajax/libs/vue/2.6.9/vue.min.js'></script>
32   <script>
33     var app = new Vue( {
34       el: '#app',
35       data: {
36         show: true
37       }
38     } );
39   </script>
40 </body>
41 </html>
```

## 03\_초기렌더링트랜지션.html

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>트랜지션</title>
7   <link rel="stylesheet" href="../css/main.css">
8   <style>
9     /* 1초 동안 투명도 변화 */
10    .v-enter-active, .v-leave-active {
11      transition: opacity 2s;
12    }
13    /* 더 이상 보이게 되지 않았을 때의 투명도 */
14    .v-enter, .v-leave-to {
15      opacity: 0;
16    }
17  </style>
18 </head>
19 <body>
20   <div id="app">
21     <!-- appear 속성을 추가하면 최초 렌더링시 트랜지션이 적용된다! -->
22     <transition appear>
23       <div v-if="show">
24         처음 렌더링 시 트랜지션 적용
25       </div>
26     </transition>
27   </div>
28
29   <!-- Vue.js -->
30   <script
31     src='https://cdnjs.cloudflare.com/ajax/libs/vue/2.6.9/vue.min.js'></script>
32   <script>
33     var app = new Vue( {
34       el: '#app',
35       data: {
36         show: true
37       }
38     } );
39   </script>
40 </body>
</html>
```

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>트랜지션</title>
7   <link rel="stylesheet" href="../css/main.css">
8   <style>
9
10    .v-enter-active,
11    .v-leave-active {
12      transition: opacity 1s, transform 1s;
13    }
14    /* 왼쪽에서 오른쪽으로 나타나기*/
15    .v-enter {
16      opacity: 0;
17      transform: translateX(-30px);
18    }
19    /* 아래로 사라지기 */
20    .v-leave-to {
21      opacity: 0;
22      transform: translateY(30px);
23    }
24  </style>
25 </head>
26 <body>
27   <div id="app">
28     <p><button v-on:click="show=!show">변경</button></p>
29     <transition>
30       <div v-show="show">
31         트랜지션 요소
32       </div>
33     </transition>
34   </div>
35
36   <!-- Vue.js -->
37   <script
38     src='https://cdnjs.cloudflare.com/ajax/libs/vue/2.6.9/vue.min.js'></script>
39   <script>
40     var app = new Vue( {
41       el: '#app',
42       data: {
43         show: true
44       }
45     } );
46   </script>
47 </body>
48 </html>
```

## 05\_그룹화.html

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>트랜지션</title>
7   <link rel="stylesheet" href="../css/main.css">
8   <style>
9     .v-enter-active, .v-leave-active {
10       transition: opacity 1s;
11     }
12     .v-enter, .v-leave-to {
13       opacity: 0;
14     }
15
16     /* 오류 수정 */
17     .v-leave-active {
18       position: absolute;
19     }
20   </style>
21 </head>
22 <body>
23   <div id="app">
24     <p><button v-on:click="show=!show">변경</button></p>
25     <transition>
26       <div v-if="show" key="a">TRUE</div>
27       <div v-else key="b">FALSE</div>
28     </transition>
29   </div>
30
31   <!-- Vue.js -->
32   <script
33     src='https://cdnjs.cloudflare.com/ajax/libs/vue/2.6.9/vue.min.js'></script>
34   <script>
35     var app = new Vue( {
36       el: '#app',
37       data: {
38         show: true
39       }
40     } );
41   </script>
42 </body>
43 </html>
```

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>트랜지션</title>
7   <link rel="stylesheet" href="../css/main.css">
8   <style>
9     .v-enter-active, .v-leave-active {
10       transition: opacity 1s;
11     }
12     .v-enter, .v-leave-to {
13       opacity: 0;
14     }
15
16     .trans {
17       width: 100px;
18       border: 2px solid lightcoral;
19       float: left;
20       margin-right: 10px;
21     }
22   </style>
23 </head>
24 <body>
25   <div id="app">
26     <p><button v-on:click="show=!show">변경</button></p>
27
28     <div class="trans">
29       <h3>normal</h3>
30       <transition>
31         <div v-if="show" key="a">TRUE</div>
32         <div v-else key="b">FALSE</div>
33       </transition>
34     </div>
35
36     <div class="trans">
37       <h3>out-in</h3>
38       <transition mode="out-in">
39         <div v-if="show" key="a">TRUE</div>
40         <div v-else key="b">FALSE</div>
41       </transition>
42     </div>
43
44     <div class="trans">
45       <h3>in-out</h3>
46       <transition mode="in-out">
47         <div v-if="show" key="a">TRUE</div>
48         <div v-else key="b">FALSE</div>
49       </transition>
50     </div>
51   </div>
52
53   <!-- Vue.js -->
```

```
54 <script
    src='https://cdnjs.cloudflare.com/ajax/libs/vue/2.6.9/vue.min.js'></script>
55 <script>
56     var app = new Vue( {
57         el: '#app',
58         data: {
59             show: false
60         }
61     } );
62 </script>
63 </body>
64 </html>
```

## 07\_키변화로트랜지션발동.html

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>트랜지션</title>
7   <link rel="stylesheet" href="../css/main.css">
8   <style>
9     .trans {
10       width: 50px;
11       height: 50px;
12       background-color: deeppink;
13       color: white;
14       text-align: center;
15       line-height: 50px;
16       font-size: 30px;
17     }
18
19     .v-enter-active {
20       transition: opacity 1s;
21     }
22
23     /* 딜레이(0.2s)를 부여하여 문자 투명도 조정 */
24     .v-leave-active {
25       transition: opacity 0.8s ease 0.2s;
26       position: absolute;
27     }
28     .v-enter, .v-leave-to {
29       opacity: 0;
30     }
31   </style>
32 </head>
33 <body>
34   <div id="app">
35     <p><button v-on:click="count++">변경</button></p>
36     <div class="trans">
37       <transition>
38         <div v-bind:key="count">{{ count }}</div>
39       </transition>
40     </div>
41   </div>
42
43   <!-- Vue.js -->
44   <script
45     src='https://cdnjs.cloudflare.com/ajax/libs/vue/2.6.9/vue.min.js'></script>
46   <script>
47     var app = new Vue( {
48       el: '#app',
49       data: {
50         count: 0
51       }
52     } );
53   </script>
```



## 07\_키변화로트랜지션발동.html

---

53 </body>

54 </html>

## 08\_이동트랜지션-1.html

```

1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>트랜지션</title>
7   <link rel="stylesheet" href="../css/main.css">
8   <style>
9     /*
10     요소가 추가/삭제/정렬 순서 변경으로 요소가 움직(이동)일 때 해당 요소에 .v-move
      트랜지션 클래스가 적용된다!
11
12     1초 동안 요소 이동 */
13     .v-move {
14       transition: transform 10s;
15     }
16   </style>
17 </head>
18 <body>
19   <div id="app">
20     <p><button v-on:click="order=!order">변경</button></p>
21
22     <!-- transition-group으로 사용할 태그는 속성으로 지정 -->
23     <transition-group tag="ul" class="list">
24       <li v-for="item in sortedList" v-bind:key="item.id">
25         {{ item.name }} {{ item.price }}원
26       </li>
27     </transition-group>
28   </div>
29
30   <!-- Vue.js -->
31   <script
32     src='https://cdnjs.cloudflare.com/ajax/libs/vue/2.6.9/vue.min.js'></script>
33   <!-- lodash -->
34   <script
35     src='https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.17.21/lodash.min.js'></sc
36     ript>
37   <script>
38     new Vue( {
39       el: '#app',
40       data: {
41         order: false,
42         list: [
43           { id: 1, name: '사과', price: 1000 },
44           { id: 2, name: '바나나', price: 4000 },
45           { id: 3, name: '수박', price: 5000 },
46           { id: 4, name: '딸기', price: 3000 }
47         ]
48       },
49       computed: {
50         // order 값에 따라 리스트의 순서를 반전하는 산출 속성
51         sortedList: function () {
52           // Lodash의 orderBy 메서드 사용

```

```
50         return _.orderBy(this.list, 'price', this.order ? 'desc' : 'asc')
51     }
52 }
53 } );
54
55 </script>
56 </body>
57 </html>
```

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>트랜지션</title>
7   <link rel="stylesheet" href="../css/main.css">
8   <style>
9     /* 박스 스타일 정의 */
10    .list {
11      width: 240px;
12      padding: 0;
13    }
14    .item {
15      display: inline-flex;
16      justify-content: center;
17      align-items: center;
18      margin: 4px;
19      width: 40px;
20      height: 40px;
21      background: #f5f5f5;
22    }
23    /* 트랜지션 전용 스타일 */
24    .v-enter-active, .v-leave-active, .v-move {
25      transition: all 1s;
26    }
27    .v-leave-active {
28      position: absolute;
29    }
30    .v-enter, .v-leave-to {
31      opacity: 0;
32      background: #f9a3b1;
33      transform: translateY(-30px);
34    }
35  </style>
36 </head>
37 <body>
38   <div id="app">
39     <p>
40       <button v-on:click="doShuffle">셔플</button>
41       <button v-on:click="doAdd">추가</button>
42     </p>
43     <transition-group tag="ul" class="list">
44       <li v-for="(item, index) in list"
45         v-bind:key="item"
46         class="item"
47         v-on:click="doRemove(index)">{{ item }}</li>
48     </transition-group>
49   </div>
50
51   <!-- Vue.js -->
52   <script
53     src='https://cdnjs.cloudflare.com/ajax/libs/vue/2.6.9/vue.min.js'></script>
```

```
53 <!-- lodash -->
54 <script
   src='https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.17.21/lodash.min.js'></sc
   ript>
55 <script>
56   new Vue( {
57     el: '#app',
58     data: {
59       list: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
60     },
61     methods: {
62       doShuffle: function () {
63         this.list = _.shuffle(this.list)
64       },
65       doAdd: function() {
66         var newNumber = Math.max.apply(null, this.list) + 1
67         var index = Math.floor(Math.random() * (this.list.length + 1))
68         this.list.splice(index, 0, newNumber)
69       },
70       doRemove: function (index) {
71         this.list.splice(index, 1)
72       }
73     }
74   } );
75 </script>
76 </body>
77 </html>
```

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>트랜지션</title>
7   <link rel="stylesheet" href="../css/main.css">
8   <style>
9     /* 박스 스타일 정의 */
10    .list {
11      width: 240px;
12      padding: 0;
13    }
14    .item {
15      display: inline-flex;
16      justify-content: center;
17      align-items: center;
18      margin: 4px;
19      width: 40px;
20      height: 40px;
21      background: #f5f5f5;
22    }
23    /* 트랜지션 전용 스타일 */
24    .v-enter-active, .v-leave-active, .v-move {
25      transition: all 1s;
26    }
27    .v-leave-active {
28      position: absolute;
29    }
30    .v-enter, .v-leave-to {
31      opacity: 0;
32      background: #f9a3b1;
33      transform: translateY(-30px);
34    }
35  </style>
36 </head>
37 <body>
38   <div id="app">
39     <p>
40       <button v-on:click="doShuffle">셔플</button>
41       <button v-on:click="doAdd">추가</button>
42     </p>
43     <!-- before-leave 혹은 추가 -->
44     <transition-group tag="ul" class="list" v-on:before-leave="beforeLeave">
45       <li v-for="(item, index) in list"
46         v-bind:key="item"
47         class="item"
48         v-on:click="doRemove(index)">{{ item }}</li>
49     </transition-group>
50   </div>
51
52   <!-- Vue.js -->
53   <script
```

```
src='https://cdnjs.cloudflare.com/ajax/libs/vue/2.6.9/vue.min.js'></script>
54 <!-- lodash -->
55 <script
src='https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.17.21/lodash.min.js'></sc
ript>
56 <script>
57   new Vue( {
58     el: '#app',
59     data: {
60       list: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
61     },
62     methods: {
63       doShuffle: function () {
64         this.list = _.shuffle(this.list)
65       },
66       doAdd: function() {
67         var newNumber = Math.max.apply(null, this.list) + 1
68         var index = Math.floor(Math.random() * (this.list.length + 1))
69         this.list.splice(index, 0, newNumber)
70       },
71       doRemove: function (index) {
72         this.list.splice(index, 1)
73       },
74       // leave 시 X&Y 위치 보정 메소드
75       beforeLeave: function (el) {
76         var style = window.getComputedStyle(el)
77         el.style.left = el.offsetLeft - parseFloat(style.marginLeft, 10) + 'px'
78         el.style.top = el.offsetTop - parseFloat(style.marginTop, 10) + 'px'
79       }
80     }
81   } );
82 </script>
83 </body>
84 </html>
```

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>트랜지션</title>
7   <link rel="stylesheet" href="../css/main.css">
8   <style>
9     .v-enter-active, .v-leave-active {
10       transition: all 1s;
11     }
12     .v-leave-active {
13       position: absolute;
14     }
15     .v-enter, .v-leave-to {
16       opacity: 0;
17       transform: translateX(-20px);
18     }
19
20     button {
21       display: block;
22     }
23   </style>
24 </head>
25 <body>
26   <div id="app">
27     <button v-on:click="toggle=!toggle">변경</button>
28     <svg xmlns="http://www.w3.org/2000/svg" version="1.1">
29       <!-- SVG 요소에 트랜지션 사용 -->
30       <transition>
31         <my-circle v-bind:fill="fill" v-bind:key="fill"></my-circle>
32       </transition>
33     </svg>
34   </div>
35
36   <!-- Vue.js -->
37   <script
38     src='https://cdnjs.cloudflare.com/ajax/libs/vue/2.6.9/vue.min.js'></script>
39   <script>
40     // SVG 요소 컴포넌트 정의
41     Vue.component( 'my-circle', {
42       template: '<circle cx="80" cy="75" r="50" v-bind:fill="fill"/>',
43       props: {
44         fill: String
45       }
46     } );
47
48     new Vue( {
49       el: '#app',
50       data: {
51         toggle: false
52       },
53       computed: {
```



09\_SVG트랜지션.html

---

```
53         fill: function () {  
54             return this.toggle ? 'lightpink' : 'skyblue'  
55         }  
56     }  
57 } );  
58 </script>  
59 </body>  
60 </html>
```

## 10\_트랜지션혹.html

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>트랜지션</title>
7   <link rel="stylesheet" href="../css/main.css">
8   <style>
9     .v-enter-active, .v-leave-active {
10       transition: all 1s;
11     }
12     .v-enter, .v-leave-to {
13       opacity: 0;
14       transform: translateX(-20px);
15     }
16   </style>
17 </head>
18 <body>
19   <div id="app">
20     <p><button v-on:click="show=!show">변경</button></p>
21     <transition
22       v-on:before-enter="beforeEnter"
23       v-on:enter="enter"
24       v-on:after-enter="afterEnter"
25       v-on:enter-cancelled="enterCancelled"
26       v-on:before-leave="beforeLeave"
27       v-on:leave="leave"
28       v-on:after-leave="afterLeave"
29       v-on:leave-cancelled="leaveCancelled">
30       <div v-if="show">콘솔 로그 확인</div>
31     </transition>
32   </div>
33
34   <!-- Vue.js -->
35   <script
36     src='https://cdnjs.cloudflare.com/ajax/libs/vue/2.6.11/vue.min.js'></script>
37   <script>
38     new Vue( {
39       el: '#app',
40       data: {
41         show: true
42       },
43       methods: {
44         // Enter
45         beforeEnter: function (el) {
46           // DOM에 요소가 추가되기 전
47           console.log('before-enter')
48         },
49         enter: function (el, done) {
50           // .v-enter가 있는 DOM에 요소가 추가된 후
51           console.log('enter')
52           setTimeout(done, 1000) // 1초 후에 enter를 종료하고 after-enter로 변경
53         },
```

## 10\_트랜지션혹.html

```
53     afterEnter: function (el) {
54         // 트랜지션이 종료되거나 enter에서 done() 호출 후
55         console.log('after-enter')
56     },
57     enterCancelled: function (el) {
58         // Enter 페이지가 중간에 취소되었을 때
59         console.log('enter-cancelled')
60     },
61     // Leave
62     beforeLeave: function (el) {
63         // 클래스가 추가되기 전
64         console.log('before-leave')
65     },
66     leave: function (el, done) {
67         // .v-leave가 추가된 후
68         console.log('leave')
69         setTimeout(done, 1000)
70     },
71     afterLeave: function (el) {
72         // DOM에서 요소가 제거된 후 또는 leave에서 done() 호출 후
73         console.log('after-leave')
74     },
75     // v-show와 함께 사용하는 경우에만 leaveCancelled 사용 가능
76     leaveCancelled: function (el) {
77         // Leave 페이지가 중간에 취소되었을 때
78         console.log('leave-cancelled')
79     }
80 }
81 } );
82 </script>
83 </body>
84 </html>
```