

jQuery 1.3 Cheatsheet

<http://www.jquery.com> <http://docs.jquery.com>

Download at: <http://www.gmtaz.com>

SELECTORS

Basics	Content Filters	Forms
#id	:contains(text)	:input
element	:empty	:text
.class	:has(selector)	:password
*	:parent	:radio
selector1, selectorN, ...		:checkbox
	Visibility Filters	:submit
Hierarchy	:hidden	:image
ancestor descendant	:visible	:reset
parent > child		:button
prev + next	Attribute Filters	:file
prev ~ siblings	[attribute]	:hidden
	[attribute=value]	
Basic Filters	[attribute != value]	Form Filters
:first	[attribute ^=value]	:enabled
:last	[attribute \$=value]	:disabled
:not(selector)	[attribute *=value]	:checked
:even	[attributeFilter1][attributeFilterN]	:selected
:odd		
:eq(index)	Child Filters	
:gt(index)	:nth-child(index/even/odd/equation)	
:lt(index)	:first-child	
:header	:last-child	
:animated	:only-child	

ATTRIBUTES / CSS

Attributes	Value
attr(name)	val()
attr(properties)	val(val)
attr(key, value)	val(val)
attr(key, fn)	
removeAttr(name)	
	Positioning
Class	offset()
addClass(class)	position()
hasClass(class)	scrollTop(val)
removeClass(class)	scrollLeft(val)
toggleClass(class)	scrollLeft(val)
toggleClass(class, switch)	
	CSS
Height and Width	css(name)
height()	css(properties)
height(val)	css(name, value)
width()	
width(val)	

EVENTS

Event Helpers			
blur()	error()	keyup()	mouseover(fn)
blur(fn)	error(fn)	keyup(fn)	mouseup(fn)
change()	focus()	load(fn)	resize(fn)
change(fn)	focus(fn)	mousedown(fn)	scroll(fn)
click()	keydown()	mouseenter(fn)	select()
click(fn)	keydown(fn)	mouseleave(fn)	select(fn)
dblclick()	keypress()	mousemove(fn)	submit()
dblclick(fn)	keypress(fn)	mouseout(fn)	submit(fn)
			upload(fn)
Event Handling	Live Events	Page Load	
bind(type, [data], fn)	live(type, fn)	ready(fn)	
one(type, [data], fn)	die([type], [fn])		
trigger(event, [data])	Interaction Helpers		
triggerHandler(event, [data])	hover(over, out)		
unbind([type], [fn])	toggle(fn, fn2, [fn3], [fn4],...)		

EFFECTS

Basics	Settings
show()	\$.fx.off
show(speed, [callback])	
hide()	
hide(speed, [callback])	
toggle()	
toggle(speed, [callback])	
toggle(switch)	
toggle(speed, [callback])	
Sliding	
slideDown(speed, [callback])	
slideUp(speed, [callback])	
slideToggle(speed, [callback])	
Fading	
fadeIn(speed, [callback])	
fadeOut(speed, [callback])	
fadeTo(speed, opacity, [callback])	
Custom	
animate(params, [duration], [easing], [callback])	
animate(params, options)	
stop([clearQueue], [gotoEnd])	

MANIPULATION

Changing Contents	Replacing
html()	replaceWith(content)
html(val)	replaceAll(selector)
text()	
text(val)	Removing
	empty()
Inserting Inside	remove([expr])
append(content)	
appendTo(selector)	Copying
prepend(content)	clone()
prependTo(selector)	clone(bool)
Inserting Outside	Inserting Around
after(content)	wrap(html)
before(content)	wrap(elem)
insertAfter(selector)	wrapAll(html)
insertBefore(selector)	wrapAll(elem)
	wrapInner(html)
	wrapInner(elem)

TRAVERSING

Filtering	Finding
eq(index)	add(expr)
filter(expr)	children([expr])
filter(fn)	closest([expr])
is(expr)	contents()
map(callback)	find(expr)
not(expr)	next([expr])
slice(start, [end])	nextAll([expr])
	offsetParent()
Chaining	parent([expr])
andSelf()	parents([expr])
end()	prev([expr])
	prevAll([expr])
	siblings([expr])

AJAX

Ajax Request
\$.ajax(options)
load(url, [data], [callback])
\$.get(url, [data], [callback], [type])
\$.getJSON(url, [data], [callback])
\$.getScript(url, [callback])
\$.post(url, [data], [callback], [type])
Ajax Events
ajaxComplete(callback)
ajaxError(callback)
ajaxSend(callback)
ajaxStart(callback)
ajaxStop(callback)
ajaxSuccess(callback)
Misc
\$.ajaxSetup(options)
serialize()
serializeArray()

CORE

The jQuery Function	Data
\$(expression, [context])	data(name)
\$(html, [ownerDocument])	data(name, value)
\$(elements)	removeData(name)
\$(callback)	queue([name])
jQuery Object Accessors	queue([name], queue)
each(callback)	dequeue([name])
size()	
length	Plugins
selector	\$.fn.extend(object)
context	\$.extend(object)
eq(position)	Interoperability
get()	\$.noConflict()
get(index)	\$.noConflict(extreme)
index(subject)	

UTILITIES

Browser and Feature D	Test operations
\$.support	\$.isArray(obj)
\$.browser	\$.isFunction(obj)
\$.browser.version	String operations
\$.boxModel	\$.trim(str)
Array and Object operations	URLs
\$.each(object, callback)	\$.param(obj)
\$.grep(array, callback, [invert])	
\$.makeArray(obj)	
\$.map(array, callback)	
\$.inArray(value, array)	
\$.merge(first, second)	
\$.unique(array)	
\$.extend([deep], target, object1, [objectN])	

16. 효과

16.1 저수준의 시각적 효과

❖ jQuery의 기본 효과

표 jQuery의 기본 효과

메서드 이름	설명
show()	문서 객체를 크게 하며 보여줍니다.
hide()	문서 객체를 작게 하며 사라지게 합니다.
toggle()	show() 메서드와 hide() 메서드를 번갈아 실행합니다.
slideDown()	문서 객체를 슬라이드 효과와 함께 보여줍니다.
slideUp()	문서 객체를 슬라이드 효과와 함께 사라지게 합니다.
slideToggle()	slideDown() 메서드와 slideUp() 메서드를 번갈아 실행합니다.
fadeIn()	문서 객체를 선명하게 하며 보여줍니다.
fadeOut()	문서 객체를 흐리게 하며 사라지게 합니다.
fadeToggle()	fadeIn() 메서드와 fadeOut() 메서드를 번갈아 실행합니다.



16.1 저수준의 시각적 효과

❖ jQuery 기본 효과 메서드 사용법

- 1. `$(selector).method();`
- 2. `$(selector).method(speed);`
- 3. `$(selector).method(speed, callback);`
- 4. `$(selector).method(speed, easing, callback);`

❖ 각 매개 변수의 의미

- speed
 - 효과를 진행할 속도 지정
 - 밀리초 단위의 숫자나 문자열 `slow`, `normal`, `fast` 입력
- callback
 - 효과를 모두 완료한 후에 실행할 함수 지정
- easing
 - 애니메이션의 easing 속성 지정
 - 별도의 플러그인 사용 않으면 문자열 `linear`와 `swing`만 입력 가능



16.1 저수준의 시각적 효과

❖ 예제 코드

- h1 태그를 누르면 h1 태그 바로 다음에 있는 div 태그를 사라지게도 생기게도 하는 예제

코드 body 태그 구성

```
<body>
  <h1>OPEN & CLOSE</h1>
  <div>
    <h1>Lorem ipsum</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  </div>
  <h1>OPEN & CLOSE</h1>
  <div>
    <h1>Lorem ipsum</h1>
    <p>Aenean vel augue dolor, at rhoncus tortor.</p>
  </div>
</body>
```



16.1 저수준의 시각적 효과

❖ 예제 코드

- h1 태그에 클릭 이벤트 연결
 - h1 태그가 클릭됐을 때 클릭한 문서 객체의 바로 다음 위치 문서 객체에 toggle() 메서드 사용
- toggle() 메서드의 매개 변수
 - 메서드가 두 개 이상 입력
 - **이벤트와 관련된 메서드로 사용**
 - 그 이외의 경우 효과 메서드로 적용

코드 toggle() 효과 메서드

```
<script>
$(document).ready(function () {
    $('h1').click(function () {
        $(this).next().toggle('slow', function () {
            alert('EFFECT END..!');
        });
    });
});
</script>
```



16.1 저수준의 시각적 효과

❖ toggle() 메서드

- show() 메서드와 hide() 메서드 번갈아 실행하는 메서드
- 코드 실행하고 OPEN & CLOSE를 누른 결과

그림  OPEN & CLOSE를 눌러 사라졌다 나타나게 하는 효과



16.1 저수준의 시각적 효과

❖ hide() 메서드와 show() 메서드

코드 hide() 메서드와 show() 메서드

```
<script>
    $(document).ready(function () {
        $('h1').toggle(function () {
            $(this).next().hide('slow');
        }, function () {
            $(this).next().show('slow');
        });
    });
</script>
```



16.2 innerFade 플러그인

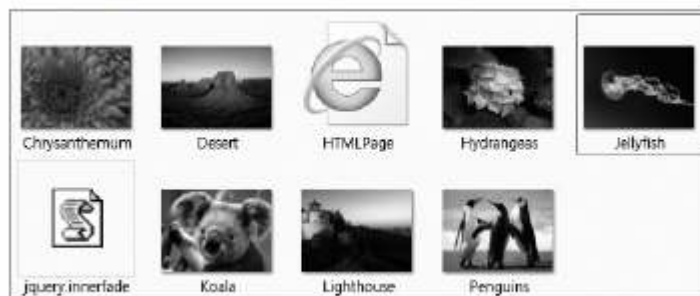
❖ innerFade 플러그인

- 간단하게 화면 전환 효과를 만들 수 있음
- <http://medienfreunde.com/lab/innerfade/> 에서 다운

그림 16-1 InnerFade 플러그인 사이트



그림 16-2 폴더 구성



16.2 innerFade 플러그인

❖ innerFade 플러그인 사용 예제

- head 태그를 구성
 - 모든 jQuery 플러그인은 jQuery.js 파일이 플러그인 파일보다 먼저 추가

코드 head 태그 구성

```
<head>
  <style>

  </style>
  <script src="http://code.jquery.com/jquery-1.7.js"></script>
  <script src="jquery.innerfade.js"></script>
  <script>
    $(document).ready(function () {

    });
  </script>
</head>
```



16.2 innerFade 플러그인

❖ innerFade 플러그인 사용 예제

- Body 태그 & Style 태그 구성

코드 body 태그 구성

```
<body>
  <ul id="inner_fade">
    <li></li>
    <li></li>
    <li></li>
    <li></li>
    <li></li>
    <li></li>
    <li></li>
  </ul>
</body>
```

코드 style 태그 구성

```
<style>
  * { margin:0px; padding:0px; }
  ul { list-style:none; }
  img { width:500px; height:350px; }
</style>
```



16.2 innerFade 플러그인

❖ innerFade 플러그인 사용 예제

- innerFade 플러그인의 가장 핵심적인 메서드는 innerfade()
 - ul 태그를 대상으로 innerfade() 메서드 적용
 - innerfade() 메서드의 첫 번째 매개 변수에는 객체 입력


코드 innerfade() 메서드

```
<script>
$(document).ready(function () {
    // innerfade 플러그인을 적용합니다.
    $('#inner_fade').innerfade({
        animationtype: 'fade',
        speed: 750,
        timeout: 2000,
        type: 'random',
        containerheight: '350px'
    });
});
</script>
```



16.2 innerFade 플러그인

❖ innerfade() 메서드의 옵션 객체 속성

표  innerFade() 메서드의 옵션 객체 속성

옵션 이름	설명	들어갈 수 있는 값
animationtype	내용의 변경 효과	fade, slide
speed	내용의 변경 속도	slow, normal, fast, 숫자
timeout	변경 효과가 적용되는 속도	숫자
type	내용의 변경 방식	sequence, random, random_start
containerheight	내용의 높이	auto, 숫자

그림  실행 결과




16.3 사용자 지정 효과

❖ animate() 메서드

■ 사용법

- 1. \$(selector).animate(object);
- 2. \$(selector).animate(object, speed);
- 3. \$(selector).animate(object, speed, easing);
- 4. \$(selector).animate(object, speed, easing, callback);

표  jQuery 사용자 지정 효과 메서드

메서드 이름	설명
animate()	사용자 지정 효과를 생성합니다.

• 첫 번째 변수에 입력 가능한 속성들

- opacity
- top
- left
- right
- bottom
- height
- width
- margin
- padding



16.3 사용자 지정 효과

❖ 사각형 좌우로 움직이는 간단한 예제

■ Body & Style 태그

- 위치 속성(left 속성과 top 속성) 변경하는 animate() 메서드 사용
 - **position 스타일 속성이 absolute나 relative이어야 !!!**

코드 ☐ body 태그 구성

```
<body>
  <div>/div>
  <div>/div>
  <div>/div>
</body>
```

코드 ☐ style 태그 구성

```
<style>
  div {
    width:50px; height:50px;
    background-color:orange;

    position:relative;
  }
</style>
```



16.3 사용자 지정 효과

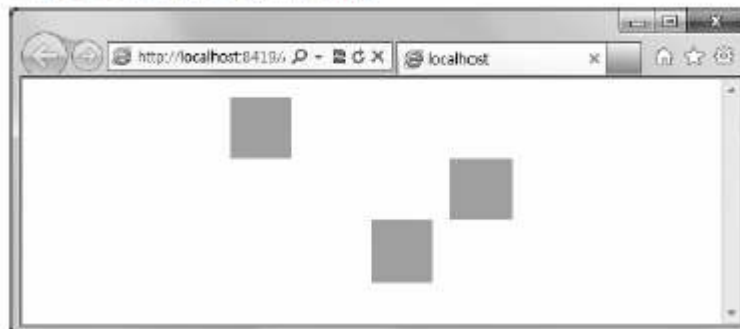
❖ 사각형 좌우로 움직이는 간단한 예제

- animate() 메서드 사용
 - 첫 번째 매개 변수에 변경하고자 하는 속성 입력

코드 animate() 메서드

```
<script>
$(document).ready(function () {
    $('#div').toggle(function () {
        $(this).animate({
            left: 500
        }, 'slow');
    }, function () {
        $(this).animate({
            left: 0
        }, 'slow');
    });
});
</script>
```

그림 클릭하면 이동하는 사각형



16.4 상대적 애니메이션

❖ animate() 메서드

- 스타일 속성을 동적으로 변경해 애니메이션 효과 내는 것
- 현재 스타일 속성 알아내 추가 값 부여
 - 상대적으로 애니메이션 적용
- style 태그 구성
 - width 스타일 속성과 height 스타일 속성 변화시키는 예제
 - position 스타일 속성 지정하지 않음

코드 style 태그 구성

```
<style>
  div {
    width:50px; height:50px;
    background-color:orange;
  }
</style>
```

코드 body 태그 구성

```
<body>
  <div/>
</body>
```



16.4 상대적 애니메이션

❖ animate() 메서드

코드 문서 객체의 스타일 속성 추출

```
<script>
    $(document).ready(function () {
        $('div').click(function () {
            // 변수를 선언합니다.
            var width = $(this).css('width');
            var height = $(this).css('height');

        });
    });
</script>
```

코드 현재 상태를 기준으로한 상대적 애니메이션 (1)

```
$('div').click(function () {
    // 변수를 선언합니다.
    var width = $(this).css('width');
    var height = $(this).css('height');

    // animate() 메서드를 사용합니다.
    $(this).animate({
        width: parseInt(width) + 50,
        height: parseInt(height) + 50
    }, 'slow');
});
```



16.4 상대적 애니메이션

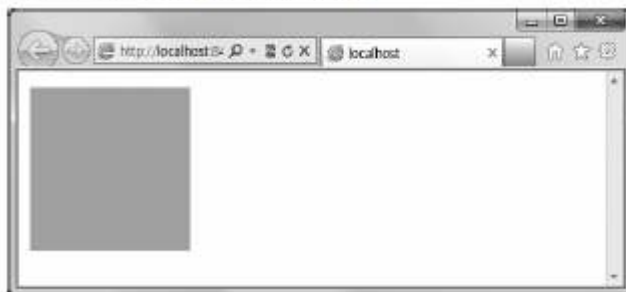
❖ animate() 메서드

- 속성을 상대적으로 변화시키는 형태의 애니메이션
 - + = 연산자나 -= 연산자 사용으로 더 효과적

코드 < > 현재 상태를 기준으로한 상대적 애니메이션 (2)

```
$('#div').click(function () {  
    // animate() 메서드를 사용합니다.  
    $(this).animate({  
        width: '+=50',  
        height: '+=50'  
    }, 'slow');  
});
```

그림 17-6 누르면 점점 커지는 사각형



16.5 애니메이션 큐

❖ jQuery의 효과 메서드는 계속 누적

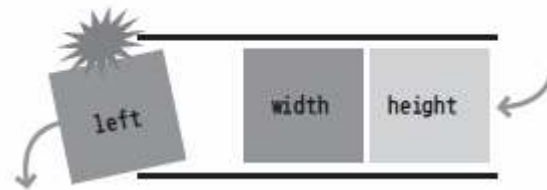
- 누적된 효과 명령은 큐Queue에 누적되어 실행

코드 HTML 페이지 구성

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {
      width:50px; height:50px;
      background-color:orange;

      position:relative;
    }
  </style>
  <script src="http://code.jquery.com/jquery-1.7.js"X/script>
  <script>
    $(document).ready(function () {
      $('div').click(function () {
        $(this).animate({
          left: '+=60'
        }, 2000).animate({
          width: '+=60'
        }, 2000).animate({
          height: '+=60'
        }, 2000);
      });
    });
  </script>
```

그림 애니메이션 큐



16.5 애니메이션 큐

❖ 큐의 내용을 제거하고 싶다면?

표 1 jQuery 메서드

메서드 이름	설명
clearQueue()	큐의 내용을 제거합니다.

코드 clearQueue() 메서드

```
<script>
$(document).ready(function () {
    // animate() 메서드를 사용합니다.
    $('div').animate({ left: '+=60' }, 2000);
    $('div').animate({ width: '+=60' }, 2000);
    $('div').animate({ height: '+=60' }, 2000);

    // 3초 후에 함수를 실행합니다.
    setTimeout(function () {
        // 애니메이션 큐를 제거합니다.
        $('div').clearQueue();

        // 효과 메서드를 사용합니다.
        $('div').hide();
    }, 3000);
});
</script>
```



16.5 애니메이션 정지

❖ clearQueue() 메서드

- 현재 실행되는 애니메이션 정지 기능 없음

표 { } jQuery 메서드

메서드 이름	설명
stop()	효과와 애니메이션을 정지합니다.

- stop() 메서드의 사용
 - 1. \$(selector).stop();
 - 2. \$(selector).stop(clearQueue);
 - 3. \$(selector).stop(clearQueue, goToEnd);
- 매개 변수 clearQueue와 goToEnd는 불리언 입력
 - 입력하지 않으면 자동으로 false를 입력한 것으로 간주
- 첫 번째 매개 변수 clearQueue를 true로 설정
 - clearQueue() 메서드를 실행하는 것과 같은 효과
- 두 번째 매개 변수 goToEnd를 true로 설정
 - 지정한 최종 상태에서 멈춤



16.5 애니메이션 정지

❖ stop() 메서드 사용 예제

- Style 과 Body 태그
 - Eval() 함수 사용하므로 대소문자 구분 확실히 할 것

코드 style 태그 구성

```
<style>
  div {
    width:100px; height:100px;
    background-color:Orange;

    position:relative;
  }
</style>
```

코드 body 태그 구성

```
<body>
  <button>stop()</button>
  <button>stop(true)</button>
  <button>stop(false, true)</button>
  <button>stop(true, true)</button>
  <div></div>
</body>
```



16.5 애니메이션 정지

❖ stop() 메서드 사용 예제

- div 태그는 2초마다 좌우로 한 번씩 이동하게 설정
 - button 태그 클릭하면 각 메서드 실행

코드 stop() 메서드

```
<script>
    $(document).ready(function () {
        // 이벤트를 연결합니다.
        $('button').click(function () {
            // 변수를 선언합니다.
            var html = $(this).html();
            var evalText = " $('div')." + html;

            // 메서드를 실행합니다.
            eval(evalText);

        });

        // 애니메이션을 시작합니다.
        setInterval(function () {
            $('div').animate({
                left: '500'
            }, 1000).animate({
                left: '0'
            }, 1000);
        }, 2000);
    });
</script>
```

그림 실행 결과

stop() stop(true) stop(false, true) stop(true, true)



16.5 애니메이션 정지

❖ stop() 메서드 사용 예제

- 사각형이 오른쪽으로 움직일 때 어떤 식으로 실행하는지 정리
 - 1. stop():
 - » 오른쪽으로 이동하는 것 멈추고 제자리에서 바로 왼쪽으로 다시 이동
 - 2. stop(true):
 - » 오른쪽으로 이동하는 것 멈추고, 다음 setInterval() 함수가 실행될 때까지 대기
 - 3. stop(false, true):
 - » 버튼 누르는 순간 사각형이 오른쪽 끝으로 이동하고, 바로 왼쪽으로 이동함
 - 4. stop(true, true): 버튼을 누르는 순간 사각형이 오른쪽 끝으로 이동하고, 다음 setInterval() 함수가 실행될 때까지 대기



16.7 애니메이션 지연

❖ 지연 메서드

표 Query 메서드

메서드 이름	설명
delay()	큐에 있는 명령을 잠시 중지합니다.

■ 예제코드

- delay() 메서드의 매개 변수에 정지하려는 시간을 밀리초 단위로 입력
- 사각형이 순서대로 오른쪽으로 이동하는 효과

코드 HTML 페이지 구성

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {
      width:100px; height:100px;
      background-color:orange;

      position:relative;
    }
  </style>
  <script src="http://code.jquery.com/jquery-1.7.js"></script>
```



16.7 애니메이션 지연

❖ 지연 메서드

■ 예제코드 (2)

```
<script>
    $(document).ready(function () {
        $('div').each(function (index) {
            // (index * 500)초 후 animate() 메서드를 실행합니다.
            $(this).delay(index * 500).animate({
                left: 500
            }, 'slow');
        });
    });
</script>
</head>
<body>
    <div></div><div></div>
    <div></div><div></div>
    <div></div><div></div>
</body>
</html>
```



16.8 jQuery Approach 플러그인

❖ jQuery Approach 플러그인

- 마우스의 거리 사용해 애니메이션 발생
- <http://srobbin.com/jquery-plugins/jquery-approach/> 에서 다운

그림 16.8.1 jQuery Approach 플러그인 사이트

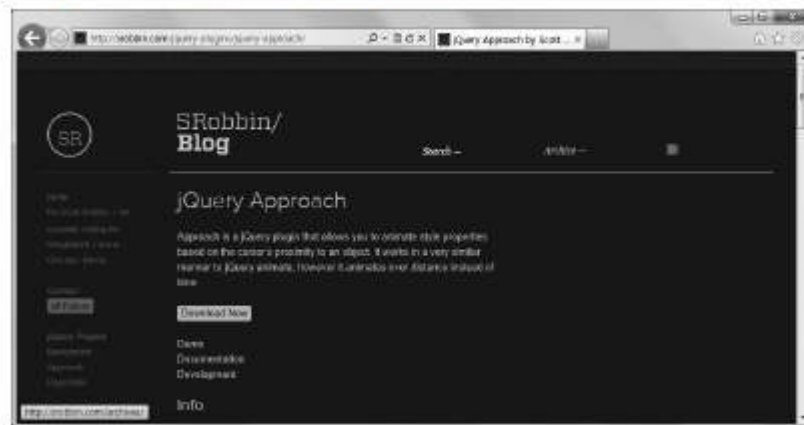
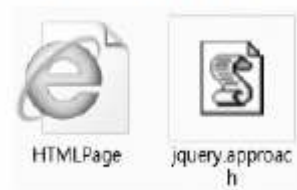


그림 16.8.2 폴더 구성



16.8 jQuery Approach 플러그인

❖ jQuery Approach 플러그인 사용

■ Head & Body 태그

- body 태그에는 circle 클래스를 갖는 div 태그를 40개~80개 정도 놓을 것

코드 head 태그 구성

```
<head>
  <style>

  </style>
  <script src="http://code.jquery.com/jquery-1.7.js"/>
  <script src="jquery.approach.js"/>
  <script>
    $(document).ready(function () {

    });
  </script>
</head>
```

코드 body 태그 구성

```
<body>
  <div id="circle_wrapper">
    <div class="circle"/>
    <div class="circle"/>
    <div class="circle"/>
    <div class="circle"/>
    <div class="circle"/>
  </div>
</body>
```



16.8 jQuery Approach 플러그인

❖ style 태그

- circle 클래스 갖는 div 태그를 원형으로 다량 배치

코드 style 태그 구성

```
<style>
  * { margin:0px; padding:0px; }
  #circle_wrapper {
    padding:10px;
    background-color:Black;
    overflow:hidden;
  }
  .circle {
    margin:5px;
    width:50px; height:50px;
    border-radius:25px;
    background-color:White;
    float:left;
  }
</style>
```



16.8 jQuery Approach 플러그인

❖ jQuery Approach 플러그인

- 핵심 메서드는 approach()
 - approach() 메서드의 첫 번째 매개 변수는 변경하려는 스타일 입력
 - approach() 메서드의 두 번째 매개 변수는 어느 거리부터 효과 발생시킬지 지정

코드 approach() 메서드

```
<script>
$(document).ready(function () {
    $('circle').approach({
        opacity: 0.2
    }, 200);
});
</script>
```

그림 approach() 메서드의 두 번째 매개 변수

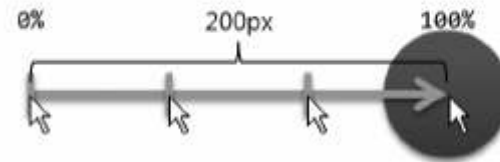
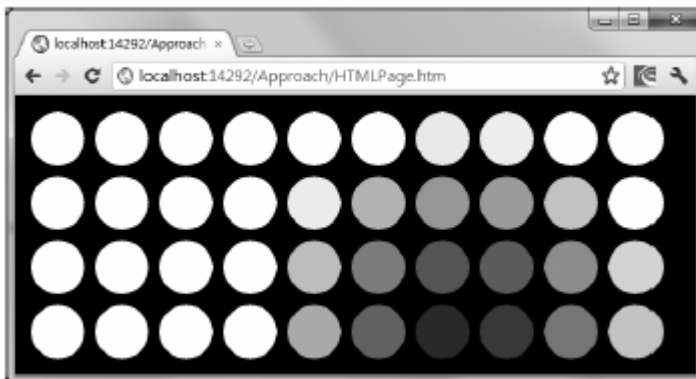


그림 마우스를 움직여보세요.



16.9 jQuery UI Effect 플러그인 다운로드

❖ jQuery UI 플러그인

- jQuery에서 공식적으로 지정한 플러그인
- 2008년 초까지 많이 사용한 유명한 jQuery 플러그인 모아 정리
- jQuery UI 플러그인 공식 사이트
 - <http://jqueryui.com/> 에서 jQuery UI 플러그인 다운

그림 1 jQuery UI Effect 플러그인 공식 사이트



그림 2 폴더 구성



16.9 jQuery UI Effect 플러그인 다운로드

❖ 예제

■ head 태그 구성

코드 head 태그 구성

```
<head>
  <link rel="Stylesheet" href="jquery-ui-1.8.14.custom.css"/>
  <script src="http://code.jquery.com/jquery-1.7.js"X/script>
  <script src="jquery-ui-1.8.14.custom.min.js"X/script>
  <script>
    $(document).ready(function () {

    });
  </script>
</head>
```



16.9 jQuery UI Effect 플러그인 다운로드

❖ jQuery UI Effect 플러그인의 기능

- 색상 애니메이션
- `addClass()`, `removeClass()`, `toggleClass()` 메서드
 - 애니메이션 기능 추가
- 고수준의 시각적 효과
- Easing 효과



16.10 jQuery UI Effect 플러그인 (1)

❖ 색상 변환 애니메이션

- addClass() 메서드와 removeClass() 메서드의 애니메이션 기능 포함
- style 태그 안에 클래스 정의

코드 style 태그 구성

```
<style>
    .reverse {
        color:White;
        background-color:Black;
    }
</style>
```



16.10 jQuery UI Effect 플러그인 (1)

❖ 색상 변환 애니메이션

- body 태그
 - 클래스 적용 태그
 - div 태그가 공간을 차지하려면 내용 있어야
 - 간단하게 h1 태그와 p 태그 입력

```
코드 body 태그 구성

<body>
  <div>
    <h1>Lorem ipsum</h1>
    <p>Lorem ipsum dolor sit amet</p>
  </div>
  <div>
    <h1>Lorem ipsum</h1>
    <p>Lorem ipsum dolor sit amet</p>
  </div>
  <div>
    <h1>Lorem ipsum</h1>
    <p>Lorem ipsum dolor sit amet</p>
  </div>
</body>
```



16.10 jQuery UI Effect 플러그인 (1)

❖ 색상 변환 애니메이션

- jQuery UI Effect 플러그인 사용
 - addClass() 메서드와 removeClass() 메서드에 애니메이션 부여하는 작업
 - 각 메서드의 두 번째 매개 변수에 애니메이션 적용 속도 입력

코드 script 태그 구성

```
<script>
$(document).ready(function () {
    $('div').hover(function () {
        $(this).addClass('reverse', 1000);
    }, function () {
        $(this).removeClass('reverse', 1000);
    });
});
</script>
```

그림 17-15 실행 결과



16.11 jQuery UI Effect 플러그인 (2)

❖ easing 플러그인

- 기존의 jQuery를 사용
 - 효과 관련 메서드의 easing 속성에 문자열 swing이나 linear만 입력 가능
- jQuery UI Effect 플러그인을 사용
 - easing 속성에 훨씬 다양한 형태의 문자열 입력 가능

표 다양한 종류의 easing 속성

linear	swing	easeInQuad	easeOutQuad
easeInOutQuad	easeInCubic	easeOutCubic	easeInOutCubic
easeInQuart	easeOutQuart	easeInOutQuart	easeInQuint
easeOutQuint	easeInOutQuint	easeInSine	easeOutSine
easeInOutSine	easeInExpo	easeOutExpo	easeInOutExpo
easeInCirc	easeOutCirc	easeInOutCirc	easeInElastic
easeOutElastic	easeInOutElastic	easeInBack	easeOutBack
easeInOutBack	easeInBounce	easeOutBounce	easeInOutBounce

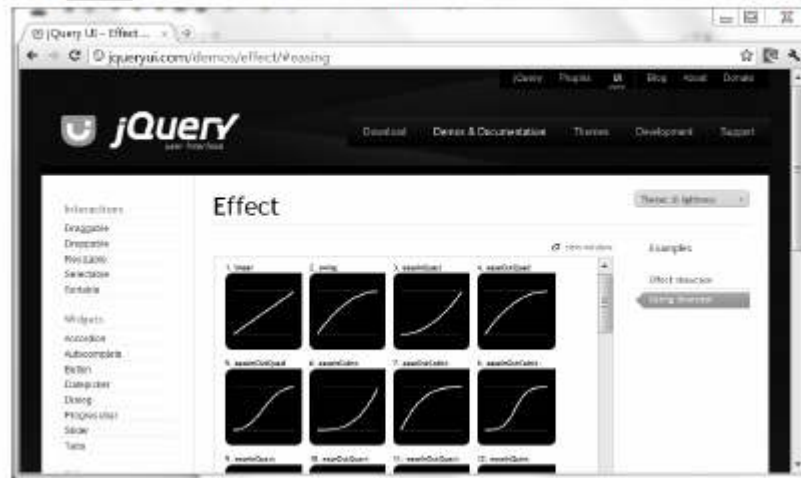


16.11 jQuery UI Effect 플러그인 (2)

❖ easing 플러그인

- jQuery UI 공식 홈페이지의 Demo & Documentation 탭
 - jQuery UI Effect 플러그인
 - 각 easing 속성이 어떤 형태를 의미하는지 설명

그림 다양한 easing 속성(<http://jqueryui.com/demos/effect/#easing>)



16.11 jQuery UI Effect 플러그인 (2)

❖ easing 플러그인

- 모든 easing 속성이 어떤 방식으로 작동하는지 알아볼 수 있는 예제
 - Body 태그
 - 생략된 부분에 표 17-8의 모든 속성을 option 태그에 넣을 것
 - » 총 32개의 option 태그
 - button 태그와 움직이는 사각형이 되어줄 div 태그 작성

코드 body 태그 구성

```
<body>
  <select>
    <option>linear</option>
    <option>swing</option>
    <option>easeInQuad</option>
    <!-- 생략 -->
    <option>easeInBounce</option>
    <option>easeOutBounce</option>
    <option>easeInOutBounce</option>
  </select>
  <button>MOVE</button>
  <div></div>
</body>
```



16.11 jQuery UI Effect 플러그인 (2)

❖ easing 플러그인

- 모든 easing 속성이 어떤 방식으로 작동하는지 알아볼 수 있는 예제
 - Style 태그

코드 style 태그 구성

```
<style>
  div {
    background-color:orange;
    width:150px; height:150px;
    position:relative;
  }
</style>
```



16.11 jQuery UI Effect 플러그인 (2)

❖ easing 플러그인

- 모든 easing 속성이 어떤 방식으로 작동하는지 알아볼 수 있는 예제
 - option 문서 객체의 내부의 문자열 가져옴
 - animate() 메서드의 세 번째 매개 변수에 입력

코드 easing 속성을 사용한 animate() 메서드

```
<script>
$(document).ready(function () {
    // 이벤트를 연결합니다.
    $('button').click(function () {
        // 변수를 선언합니다.
        var easing = $('select > option:selected').html();

        // animate() 메서드를 사용합니다.
        $('div').animate({
            left: '400'
        }, 2000, easing).animate({
            left: '0'
        }, 1000, easing);
    });
});
</script>
```

그림 실행 결과

easeInQuart MOVE

