



2D 웹 그래픽 기술

2D web graphic technology

자바스크립트 이벤트



이벤트?

- 키보드 입력 또는 마우스 클릭과 같은 다른 것에 영향을 미치는 것
- 자바스크립트 이벤트 종류
 - > 마우스 이벤트
 - > 키보드 이벤트
 - > HTML 프레임 이벤트
 - > HTML 입력 양식 이벤트
 - > 유저 인터페이스 이벤트
 - > 구조 변화 이벤트
 - > 터치 이벤트

자바스크립트 이벤트



이벤트 모델 1

- 자바스크립트에서 문서 객체의 이벤트 속성으로 이벤트를 연결하는 방법

```
...  
<body>  
  <h1 id="header">Click</h1>  
</body>  
...  
<script>  
window.onload = function() {  
  var header = document.getElementById("header");  
  header.onclick = function () {  
    alert("클릭");  
  };  
};  
</script>
```

자바스크립트 이벤트



이벤트 모델 1 (스타일 변경)

- 이벤트 발생한 객체 스타일 변경

```
...  
<body>  
  <h1 id="header">Click</h1>  
</body>  
...  
<script>  
window.onload = function() {  
  var header = document.getElementById("header");  
  header.onclick = function () {  
    this.style.color = 'orange';  
    this.style.backgroundColor = 'red';  
  };  
};  
</script>
```

자바스크립트 이벤트



이벤트 모델 2 (인라인)

- Script 태그 안에 함수를 만들고 이를 호출하는 방식

```
...  
<script>  
function whenClick(){  
  alert("클릭");  
}  
</script>  
</head>  
  
<body>  
  <h1 onclick="whenClick(event)">Click</h1>  
</body>  
...
```

자바스크립트 이벤트



이벤트 모델 3 (표준)

- 웹 표준을 만드는 단체인 W3C에서 공식적으로 지정한 DOM Level 2 이벤트 모델

```
addEventListener(event Name, handler, userCapture)
removeEventListener(eventName, handler)
```

```
<script>
  window.onload = function() {
    var header = document.getElementById("my-header");

    header.addEventListener('click', function(){
      this.innerHTML += '+';
    });
  };
</script>
</head>

<body>
  <h1 id='my-header'>Click</h1>
</body>
...
```

자바스크립트 이벤트



이벤트 모델 3 (표준)

```
...
<script>
  window.onload = function() {
    var header = document.getElementById("my-header");

    header.addEventListener('click', function(){
      this.innerHTML += '+';
    });
  };
</script>
</head>

<body>
  <h1 id='my-header'>Click</h1>
</body>
...
```

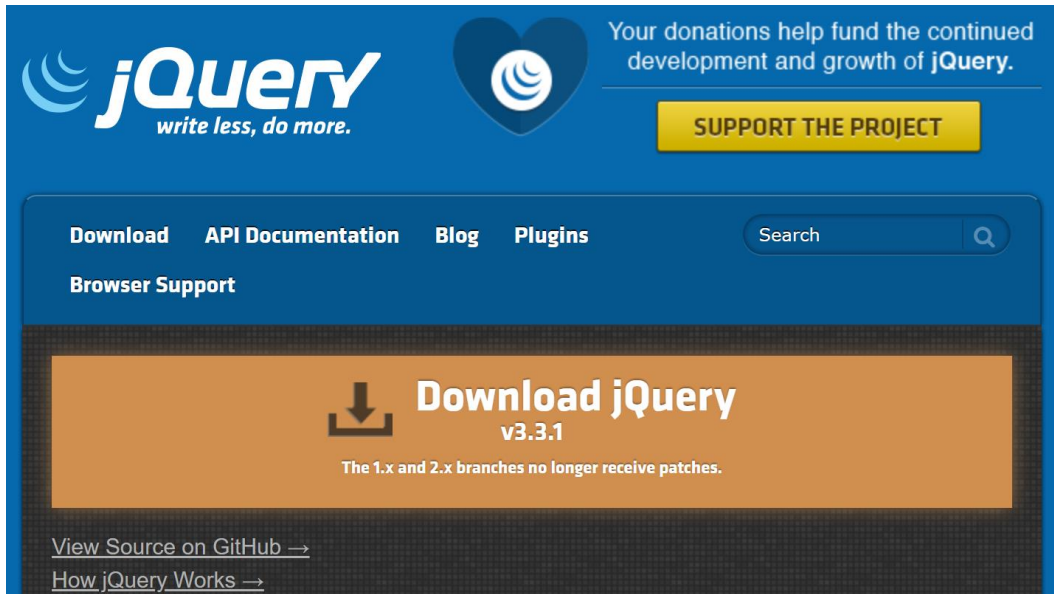
jQuery

- 모든 브라우저에서 동작하는 클라이언트 자바스크립트 라이브러리
- 무료로 사용 가능한 오픈 소스 라이브러리
- 왜?
 - > 문서 객체 모델과 관련된 처리를 쉽게 구현
 - > 일관된 이벤트 연결을 쉽게 구현
 - > 시각적 효과를 쉽게 구현
 - > Ajax 어플리케이션을 쉽게 개발

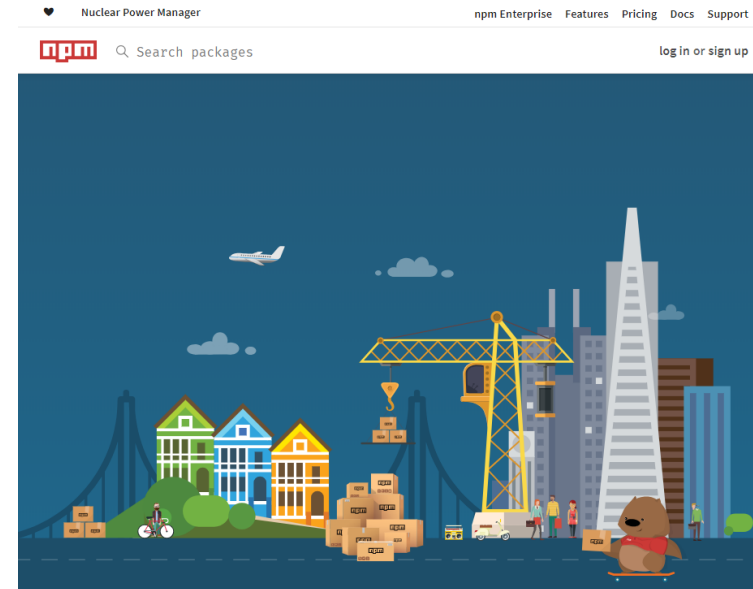
웹 라이브러리

jQuery 다운로드

- jQuery.com 에 접속 후 다운
- npm 방식으로 다운



직접 다운로드



npm install jquery



jQuery 1, 2, 3

- jQuery는 현재 1, 2, 3 버전이 다양하게 사용되고 있음

1.x VS 2.x

- 2.x 부터는 오래된 웹 브라우저 지원 X (IE 6, IE 7, IE 8)
(그러다보니 파일 사이즈가 감소)
- 메소드 차이는 크게 나지 않음

2.x VS 3.x

- HTML5 호환성 개선
- 전반적인 메소드 차이는 크게 없지만 버전이 크게 변경된 만큼 1.x 또는 2.x로 개발한 경우 호환 체크 반드시 필요



jQuery 3 와 일반 JavaScript 비교

- 문서 로드 후 이벤트

```
window.onload = function() {  
};
```

```
$(document).ready(function () {  
});
```

```
$(function () {  
});
```



jQuery 3 와 일반 JavaScript 비교

- 기본 선택자

```
$(선택자).xxx
```

- 예)

- `document.getElementsByTagName('h1');`
- `document.getElementById('idName');`
- `document.getElementsByClassName('className');`
- `$('h1').css('color', 'red');`
- `$('#idName').css('color', 'red');`
- `$('.idName').css('color', 'red');`



jQuery 반복문

- 일반적인 for문 사용해도 무관
- \$.each 라는 메서드 존재

```
<script>
  $(function(){
    var array = [
      {name : 'Kwangseob', link:'http://www.engintruder.com'},
      {name : 'Naver', link:'http://www.naver.com'},
      {name : 'Google', link:'http://www.google.com'}
    ];
    var div = $('#content');
    $.each(array, function (index, item){
      var text = div.html();
      text += '<div>' + item.name + ': ' + item.link + '</div>';
      div.html(text);
    });
  });
</script>
```



jQuery 이벤트

- 기존 자바스크립트 이벤트 연결할 때보다 훨씬 간편하게 이벤트 연결 가능
- on() 메서드 사용

```
$(선택자).on(이벤트이름, function(evt){  
  ...  
});
```

```
document.getElementById('선택자').onclick = function () {  
  ...  
}
```

```
$('선택자').on('click', function() {  
  ...  
});
```



jQuery 이벤트

- 이벤트 이름 종류

| 이벤트 | 설명 |
|------------|--|
| click | 노드(elements)를 마우스 포인터로 눌렀다가 떴을 때에 발생 |
| dblclick | 노드를 더블 클릭 했을 때에 발생 |
| hover | mouseenter와 mouseleave 이벤트를 한번에 bind한다. |
| mousedown | 노드 영역에서 마우스를 눌렀다가 떴을 때에 발생 |
| mouseenter | 노드에 마우스가 진입했을 때에 발생(자식노드에서는 이벤트를 감지 못함) |
| mouseleave | 마우스가 노드에서 벗어났을 때에 발생 |
| mousemove | 노드 영역에서 마우스를 움직였을 때에 발생 |
| mouseout | 노드에서 마우스 포인터가 떠났을 때에 발생 |
| mouseover | 노드 영역에서 마우스를 올려놓았을 때 발생 (내부노드까지 이벤트를 감지) |
| mouseup | 마우스 포인터를 노드에 올려놓고 마우스 버튼을 눌렀다 떴을 때에 발생 |
| toggle | click 이벤트에 핸들러를 바인딩하고 클릭할 때마다 실행될 함수들을 차례대로 실행 |



jQuery 이벤트 예제

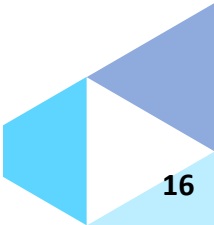
- 타이머 예제 <https://github.com/engintruder/Timer>

현재 시간

11:42:35

남은 시간

01:10:00



Canvas



HTML5 Graphic - Canvas

스크립트를 통해 즉석에서 그래픽을 그리는데 사용

<canvas> 요소는 그래픽 컨테이너.
내부에 실제 그래픽을 그리는 스크립트를 사용

Paths, Boxes, Circles, text, images

A canvas is a rectangular area on an HTML page

```
<canvas id="myCanvas" width="200" height="100">  
</canvas>
```

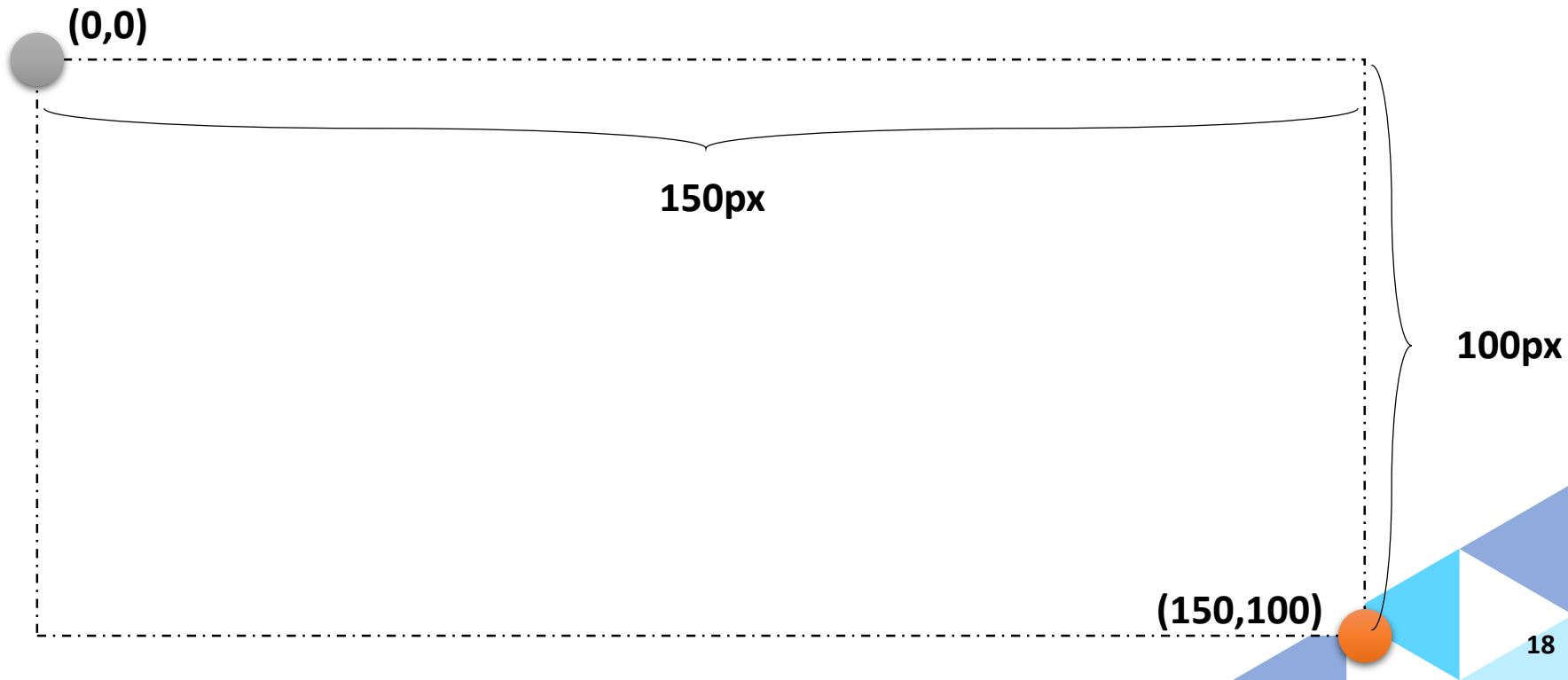
Canvas



HTML5 Graphic - Canvas

- 2차원 그리드 구성
- 왼쪽 상단 코너가 좌표의 첫 시작점 (0, 0)

if) canvas 너비 : 150px 높이 : 100px 의 컨테이너를 생성



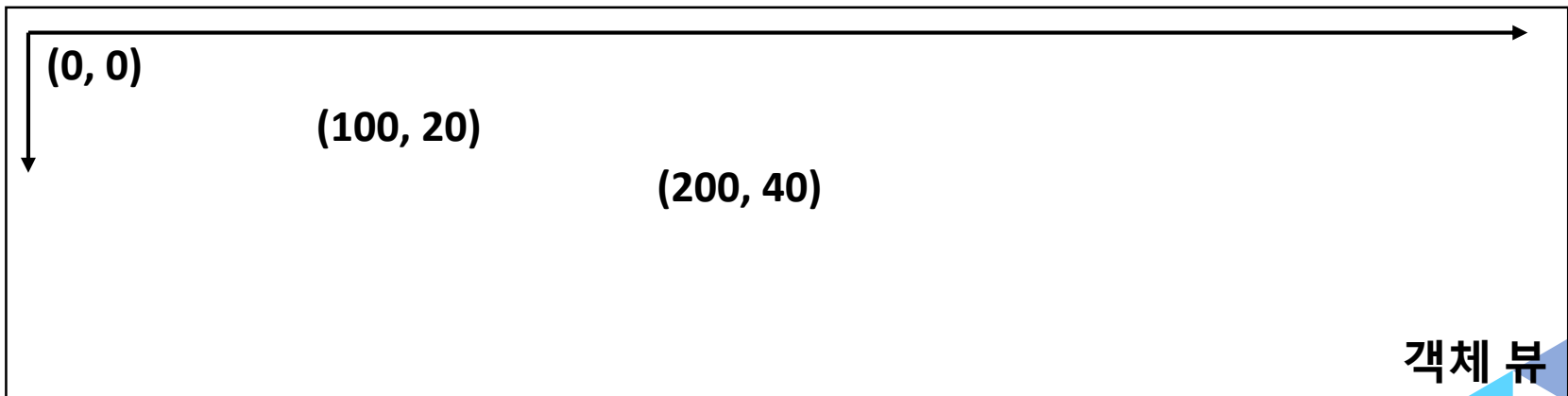
SVG

2차원 벡터 그래픽을 표현하기 위한 XML 기반 파일 형식

- 1999년 W3C 주도하 개발된 오픈 표준 벡터 그래픽
- rect, circle, ellipse, line, path 같은 많은 종류의 시각적 문서 요소를 넣을 수 있음
- 브라우저 간 일관성이 뛰어나며 빠름



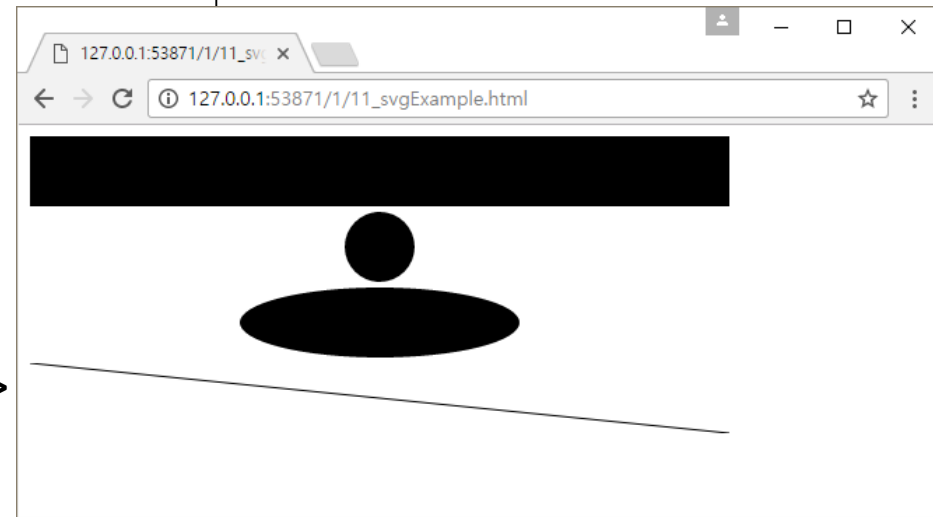
SVG 좌표 시스템



SVG

- 사각형, 원, 타원, 라인 그리기

```
...  
<body>  
  <svg width="500" height=50">  
    <rect x="0" y="0" width="500" height="50"/>  
  </svg>  
  <br>  
  <svg width="500" height=50">  
    <circle cx="250" cy="25" r="25"/>  
  </svg>  
  <br>  
  <svg width="500" height=50">  
    <ellipse cx="250" cy="25" rx="100" ry="25"/>  
  </svg>  
  <br>  
  <svg width="500" height=50">  
    <line x1="0" y1="0" x2="500" y2="50" stroke="black"/>  
  </svg>  
</body>  
...
```



SVG

- 텍스트 그리기

```
...  
<body>  
  <svg width="500" height="50">  
    <rect x="0" y="0" width="500" height="50"/>  
  </svg>  
  <br>  
  <svg width="500" height="50">  
    <circle cx="250" cy="25" r="25"/>  
  </svg>  
  <br>  
  <svg width="500" height="50">  
    <ellipse cx="250" cy="25" rx="100" ry="25"/>  
  </svg>  
  <br>  
  <svg width="500" height="50">  
    <line x1="0" y1="0" x2="500" y2="50" stroke="black"/>  
  </svg>  
</body>  
...
```

Easy-peasy

Easy-peasy

Easv-peasv

SVG

- SVG 요소 스타일 입히기

- 공통 프로퍼티(property) 목록

fill : 바탕색 값 (색상명, 헥사 코드, RGB, RGBA)

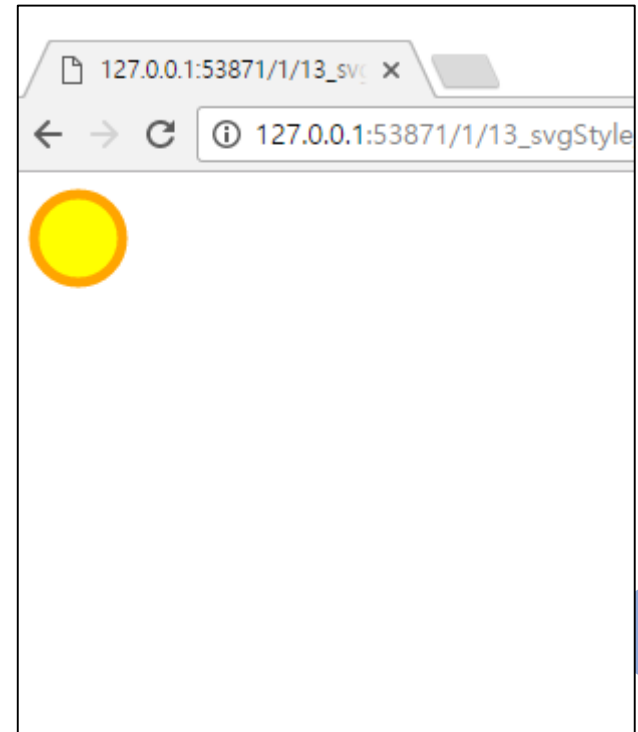
stroke : 선 색상 값

stroke : 선 두께 (단위 : 픽셀)

opacity : 불투명도 값 (0.0 ~ 1.0)

- CSS 클래스 활용 스타일 추가 가능

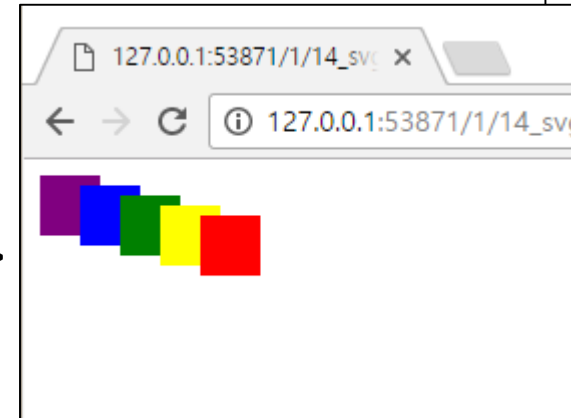
```
...  
<style>  
  .pumpkin {  
    fill: yellow;    stroke : orange;    stroke-width: 5;  
  }  
</style>  
</head>  
<body>  
  <svg width="500" height="50">  
    <circle cx="25" cy="25" r="22" class="pumpkin"/>  
  </svg>  
</body>  
...
```



SVG

- 중첩 순서와 그리는 순서
 - 포토샵 같은 '레이어' 개념 없음
 - CSS z-index 속성 지원하지 않음
 - 하지만 그리는 순서가 깊이를 나타내도록 만들기 때문에 고려해야 함

```
...  
<body>  
  <svg width="500" height="500">  
    <rect x="0" y="0" width="30" height="30" fill="purple"/>  
    <rect x="20" y="5" width="30" height="30" fill="blue"/>  
    <rect x="40" y="10" width="30" height="30" fill="green"/>  
    <rect x="60" y="15" width="30" height="30" fill="yellow"/>  
    <rect x="80" y="20" width="30" height="30" fill="red"/>  
  </svg>  
</body>  
...
```



SVG

- 투명도

- 요소들이 겹쳐 있지만 사라지면 안되거나 일부만 강조하고 다른 요소를 주목 받지 않기 바랄 때 사용

지정 방법

1. RGBA 사용
2. Opacity 값 지정

→ RGBA를 사용하면 fill과 stroke에 투명도를 독립적으로 지정할 수 있음

