

## SVG 資料第 5 回目 (その 1) テキストの挿入と新しい要素の追加

メディア専門ユニット I(SVG)

2017/5/23

クリックした位置を SVG の図形内に表示する

# クリックした位置を SVG 内に表示

## ソースコード (1)-JavaScript の部分

第 5 回目 (その 1)

メディア専門ユニット I(SVG)

クリックした位置を  
SVG 内に表示

新しい要素を追加  
する

```
1<?xml version="1.0" encoding="UTF-8" ?>
2<svg xmlns="http://www.w3.org/2000/svg"
3    xmlns:xlink="http://www.w3.org/1999/xlink"
4    height="100%" width="100%">
5  <title>クリックした位置を SVG 内に表示</title>
6  <script type="text/ecmascript">
7//    <![CDATA[
8    window.onload = function() {
9      document.getElementById("Rect").addEventListener("click",click, false);
10 }
11 function click(event) {
12   document.getElementById("XP").firstChild.nodeValue=event.clientX;
13   document.getElementById("YP").firstChild.nodeValue=event.clientY;
14   document.getElementById("Circle").setAttribute("cx",event.clientX);
15   document.getElementById("Circle").setAttribute("cy",event.clientY);
16 }
17//    ]]></script>
```

# クリックした位置を SVG 内に表示

## ソースコード (2)-JavaScript の部分

第 5 回目 (その 1)

メディア専門ユニット I(SVG)

クリックした位置を SVG 内に表示

新しい要素を追加する

```
18 <style type="text/css">
19   .textStyle {
20     font-size:20px; text-anchor:end;
21   }
22 </style>
23 <circle id="Circle" cx="200" cy="50" r="20" fill="red"/>
24 <text class="textStyle" x="50" y="50"> X</text>
25 <text class="textStyle" id="XP" x="100" y="50"> </text>
26 <text class="textStyle" x="50" y="100"> Y</text>
27 <text class="textStyle" id="YP" x="100" y="100"> </text>
28 <rect id="Rect" x="0" y="0" width="100%" height="100%" opacity="0" />
29 </svg>
```

## クリックした位置を SVG 内に表示-ソースコード解説

- ▶ 「クリックした位置に円が移動」に文字を表示する部分が追加 (24 行目から 27 行目)
- ▶ 25 行目と 27 行目の<text>要素の中には 1 個空白が置いてあることに注意
- ▶ 28 行目に不透明度が 0 である (まったく見えない) 長方形がある。
- ▶ click イベントはこの<rect>要素に付けている (9 行目)
- ▶ クリックの処理でテキストノードの値を書き直すのは 12 行目と 13 行目で行っている
  - ▶ firstChild は対象の要素の先頭の子要素を意味する。
  - ▶ ここでは<text>要素の中の文字列を指す
  - ▶ これは通常の SVG の要素ではなく、テキストノードになる
  - ▶ nodeValue はこのテキストノードの文字列を参照する
  - ▶ ここに空白がないと firstChild が存在しないので、エラーが発生 (確かめよう)
  - ▶ 空白を入れ忘れても動くコードが配布資料の SVG リスト 7.6(143 ページ)

## やってみよう

第 5 回目 (その 1)

メディア専門ユニット I(SVG)

クリックした位置を  
SVG 内に表示

新しい要素を追加  
する

- ▶ 表示する図形を円から別なものに変えて、その位置を示す属性の値を表示する
- ▶ 28 行目にある<rect>要素を 23 行目にある<circle>要素の前に置くと起こる不備な点を指摘しなさい。
- ▶ SVG リスト 7.6 では空白がなくてもエラーが起きないことを確認する。

## 直線を引く

### 新しい要素を追加する例

- ▶ 初期画面は何も要素がない
- ▶ ドラッグすると直線が引ける

# 直線を引く

## ソースコード (1)

第 5 回目 (その 1)

メディア専門ユニット I(SVG)

クリックした位置を  
SVG 内に表示

新しい要素を追加  
する

```
1<?xml version="1.0" encoding="UTF-8" ?>
2<svg xmlns="http://www.w3.org/2000/svg"
3    height="100%" width="100%">
4  <title>画面上に直線を引く</title>
5  <script type="text/ecmascript">
6//<![CDATA[
7  let svgNS ="http://www.w3.org/2000/svg";
8  let C, NewLine = null;
9  window.onload = function() {
10    C = document.getElementById("Canvas");
11    C.addEventListener("mousedown",mousedown, false);
12    C.addEventListener("mouseup",mup, false);
13  }
14  function mdown(E) {
15    NewLine = document.createElementNS(svgNS, "line");
16    NewLine.setAttribute("x1",E.clientX);
17    NewLine.setAttribute("y1",E.clientY);
18    NewLine.setAttribute("x2",E.clientX);
19    NewLine.setAttribute("y2",E.clientY);
20    NewLine.setAttribute("stroke","red");
21    NewLine.setAttribute("stroke-width","4");
22    C.appendChild(NewLine);
23    C.addEventListener("mousemove",mmove, false);
24  }
```



# 直線を引く

## ソースコード (2)

第 5 回目 (その 1)

メディア専門ユニット I(SVG)

クリックした位置を  
SVG 内に表示

新しい要素を追加  
する

```
25  function mmove(E) {
26      NewLine.setAttribute("x2",E.clientX);
27      NewLine.setAttribute("y2",E.clientY);
28  }
29  function mup(E) {
30      C.removeEventListener("mousemove",mmove, false);
31  }
32//]]></script>
33  <g id="Canvas">
34      <rect x="0" y="0" width="100%" height="100%" fill="white"/>
35  </g>
36</svg>
```

## 直線を引く-ソースコード解説

「円をドラッグする」とプログラムの構造はほとんど同じ

- ▶ 9 行目から 13 行目でファイルのロードが終了したときに呼ばれる関数を定義
- ▶ 画面全体を覆う長方形 (34 行目) を子要素に持つ<g>要素に mousedown と mouseup のイベント処理関数を登録
- ▶ 14 行目から 24 行目で mousedown のイベント処理関数を定義
  - ▶ 名前空間を指定して要素を新規に作成するメソッド (createElementNS) で<line>要素を生成 (15 行目)
  - ▶ その要素の直線の始点と終点をイベントが発生したカーソル位置で設定 (16 行目から 19 行目)
  - ▶ 20 行目と 21 行目で色と直線の幅を設定
  - ▶ 23 行目で mousemove のイベント処理関数の登録
- ▶ 25 行目から 28 行目で mousemove のイベント処理関数で終点の位置をイベントが発生したカーソル位置に設定
- ▶ 29 行目から 31 行目で mouseup のイベント処理関数で mousemove のイベント処理関数を取り除く

## やってみよう

第 5 回目 (その 1)

メディア専門ユニット I(SVG)

クリックした位置を SVG 内に表示

新しい要素を追加する

- ▶ 何本かの直線を引いた後で次のことを確認
  - ▶ 右ボタンでクリックしてソースコードが変化しているか
  - ▶ DOM ツリーを見て、作成した直線の要素があるか
- ▶ SVG 内に小さな正方形で塗りの色が異なるものをいくつか置き、その正方形をクリックした後では、クリックした正方形の塗りの色で直線が引けるようにする (配布資料 155 ページ図 7.14 参照)
- ▶ 長方形をドラッグで描く