Webプログラミング実習1

2017/7/12

Kazuma Sekiguchi
class@cieds.jp

Canvas

- Canvas自体はHTML5で登場したタグ
  - ドキュメント内に絵を描くための場所を生成するタグ
- JSを利用することで、Canvasに対して任意の画像を作成することが可能
  - JSを使って絵を描く、ということ
    - アニメーションも可能
  - イメージ画像をCanvas上に出力することも可能
    - 画像の上にJSで描いた画像を載せる、ということも可能

Canvasのデフォルト

- 結構使いづらい
  - JSを駆使して記述が必要
  - ブラウザーごとの差異が比較的大きい（特に小数点以下の扱いが異なる）
  - レイヤー機能などは存在しない（自分で作成するか、外部ライブラリを使うのが一般的）
  - 描いたものはすべて同一のものとして扱われる
    - ペイントソフトと同じ扱い
    - 異なるものを描いたとしても同一扱いのため、それぞれに違うことをすることが不可能

Canvas

- Canvasタグを利用する際は、width値とheight値、id名を付与する
  - width値とheight値はそのままCanvasの大きさになる
  - id名はJSからcanvasを識別するために必要
    - canvas自体は複数設置が可能
- JSからcanvasを利用する場合は、id名をキーにしてCanvasオブジェクトを取得し、そこに対して、描画を行っていく

図形の描画

- 矩形
  - ctx.fillRect(x, y, width, height)
    - 塗りつぶした矩形の描画
    - x,y：座標位置、width：幅、height:高さ
  - ctx.strokeRect(x, y, width, height)
    - 枠線だけの矩形の描画
- 円
  - ctx.arc(x, y, radius, startAngle, endAngle, anticlockwise)
    - 円や円弧の描画
    - x,y：座標位置（中心）
    - radius：半径
    - startAngle：円弧を書き始める角度（ラジアン指定）
    - endAngle：円弧を書き終える角度（ラジアン指定）
    - anticlockwise：円弧を描く向きを真偽値で指定。trueで反時計周り
  - ラジアン：角度*Math.PI/180

色の指定

- ctx.strokeStyle = color
  - 枠線の色
  - colorはCSS色指定が可能（rgba指定で半透明も可能）
- ctx.fillStyle = color
  - 塗りつぶしの色

- グラデーションの指定の仕方は異なる

画像の読み込み

- 任意の画像ファイルを読み込み可能
  - ブラウザーが扱うことができることは最低条件
  - 実のところ映像も扱う事が可能
- ctx.drawImage(“ファイル名”,貼り付ける場所（X座標）,貼り付ける場所（Y座標）,貼り付けるサイズ（幅）、貼り付けるサイズ（高さ）);
- 縮小したり拡大して貼り付けることが可能
  - 貼り付けた後は、普通のピクセルデータの扱いになる