

第4章 アナログ時計の制作

Web応用

第6回 日付の取得

第4章

アナログ時計の制作

第4章 学習目標

Canvas APIを活用したアナログ時計の制作方法について学びましょう。

時間を表示するファイル

Canvas APIは描画のしくみです。詳しくは第10回以降で学びます。アナログ時計を制作するのに先行して使用しましょう。

1. ファイルの準備

前回学んだ内容を元に、デジタル時計のファイルを準備しましょう。ファイル名は sample6-4.html です。

■ ソースコード

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <title>サンプル6-4</title>
6     <style>
7     </style>
8   </head>
9   <body>
10    <p id="date1"></p>
11
12    <script>
13      var date1 = document.getElementById("date1");
14
15      setInterval(clock1,1000);
16
17      function clock1(){
18        var now = new Date();
19        var h = now.getHours();
20        var i = now.getMinutes();
21        var s = now.getSeconds();
22        date1.innerHTML = h+"時"+i+"分"+s+"秒";
23      }
24    </script>
25  </body>
26 </html>

```

Canvas要素の追加

文字の表示をcanvasに変更しましょう。

1. canvas要素

■ ソースコード

```

9   <body>
10  <p id="date1"></p>

```

を

■ ソースコード

```

9   <body>
10  <canvas id="date1" width="300" height="300"></canvas>

```

に変更します。

解説：

canvas要素内に描画することができます。

その横幅、高さは開始タグ内のwidth属性、height属性で指定します。

時計、分針、秒針の描画

canvasに針を表示させます。

1. 描画のコードの記述

描画のコードを記述しましょう。

■ ソースコード

```
12 <script>
13     var date1 = document.getElementById("date1");
```

に描画のオブジェクトを追加します。

■ ソースコード

```
12 <script>
13     var date1 = document.getElementById("date1");
14     var ctx1 = date1.getContext("2d");
```

詳細は10回で解説するので今回はそのまま記述してください。

2. 表示の方法

表示の方法を変更します。

■ ソースコード

```
23     date1.innerHTML = h+"時"+i+"分"+s+"秒";
```

を次の描画のコードに入れ替えましょう。

■ ソースコード

```

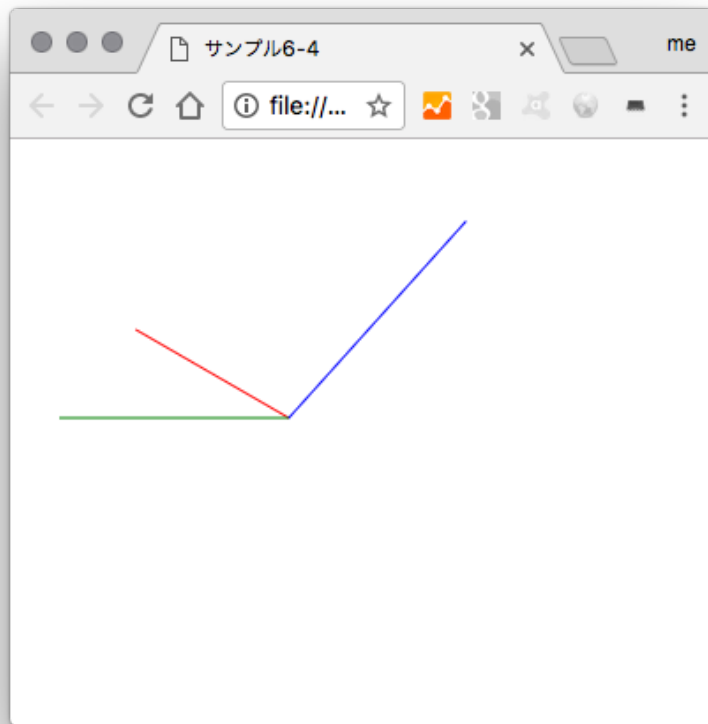
23  hdeg = h/6*Math.PI;
24  ideg = i/30*Math.PI;
25  sdeg = s/30*Math.PI;
26
27  ctx1.clearRect(0,0,300,300);
28
29  ctx1.strokeStyle="red";
30  ctx1.beginPath();
31  ctx1.moveTo(150,150);
32  ctx1.lineTo(150+Math.sin(hdeg)*100,150-Math.cos(hdeg)*100);
33  ctx1.stroke();
34
35  ctx1.strokeStyle="green";
36  ctx1.beginPath();
37  ctx1.moveTo(150,150);
38  ctx1.lineTo(150+Math.sin(ideg)*130,150-Math.cos(ideg)*130);
39  ctx1.stroke();
40
41  ctx1.strokeStyle="blue";
42  ctx1.beginPath();
43  ctx1.moveTo(150,150);
44  ctx1.lineTo(150+Math.sin(sdeg)*140,150-Math.cos(sdeg)*140);
45  ctx1.stroke();

```

解説：

以下については第10回で詳しく説明するので、今回は触れるだけで大丈夫です。
コピーして貼り付けても構いません。以下は参考です。

- `ctx1.clearRect(0,0,300,300)`は、座標(0,0)から横幅300px、高さ300pxの範囲を消去します。
- `ctx1.strokeStyle="red";`は、描画する線の色を指定します（ここでは赤）
- `ctx1.beginPath();`は、描画のパスを開始します。
- `ctx1.moveTo(x,y);`は、開始点を指定します。
- `ctx1.lineTo(x1,y1);`は、直線をひいた終点を指定します。
- `ctx1.stroke();`は線を描画します。



3.背景

背景を画像を使ってCSSで取り付けましょう。

background-imageプロパティを使用します。背景画像「bg.png」を準備しているのでダウンロードしてください。

■ ソースコード

```
6 <style>
7   #date1{
8     background-image:url(bg.png);
9     background-size:100% 100%;
10  }
11 </style>
```

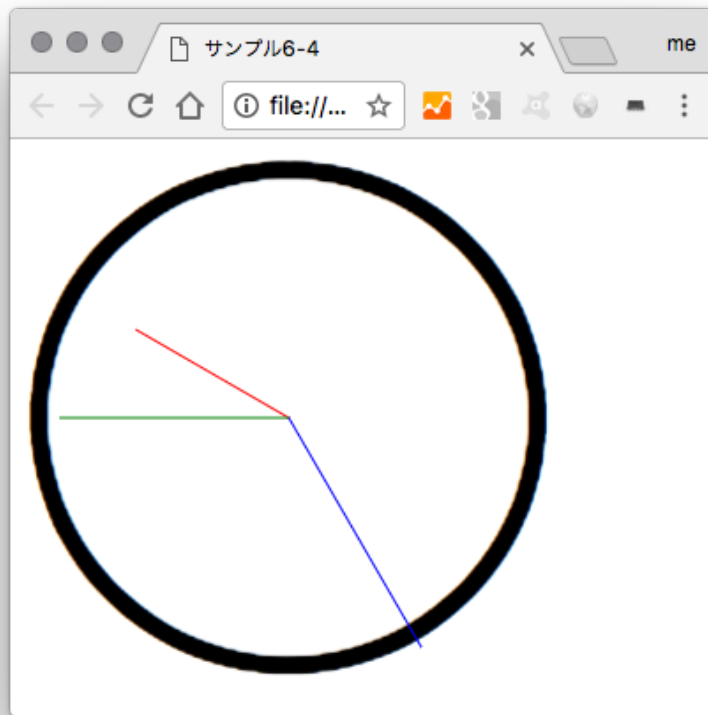
解説：

canvasは透明な要素なので、背景を指定すれば表示されます。

多少大きさが違って画像が収まるように、background-sizeプロパティに「100% 100%」を指定しておきました。

応用：

オリジナルな画像を貼り付けてみてください。



練習問題1

問題

【クイズ】 択一選択（即解答表示）

この章で扱ったcanvas要素の横幅と高さはどこで指定したか。

- ☐ 開始タグ内
- ☐ 終了タグ内
- ☐ CSS
- ☐ ブラウザが最適な大きさに自動調節する

練習問題1の解説

正解は、**開始タグ内** です。

canvas要素は次のように記述しました。

```
<canvas id="date1" width="300" height="300"></canvas>
```

width属性は横幅、height属性は高さを指定します。pxなどの単位は指定しません。

第4章 まとめ

Canvas APIを活用したアナログ時計の制作方法について学びました。

第4章 終わり

Web応用

第6回 日付の取得

第4章

アナログ時計の制作

おわり