ソフトウェア開発 第**7**回目授業

このファイルは

http://www.hilano.org/hilano-lab/dev2018-07.pdf で 得られます。

平野 照比古

2018/11/16

ソフトウェア開発 第7回目授業

平野 照比古

則凹の演習

正規表現を作る

正規表現のマッチを確認

HTML 文書の構成

CSS の利

DOM の利用

DOM とは
DOM のメソッド

- ▶ C 言語の変数名 (正確には識別子) は英字で始まり、そ のあとに英数字が並んだもの(正確にはもう少し使え る文字がある)
- ▶ 先頭の文字は文字クラスを使うと [A-Za-z] A-z という解答は文字 [も含まれるので不可
- ▶ そのあとは英数字
- ▶ その文字クラスは \w
- ▶ 0 個でもよいので、\w*
- ▶ 全体がこれだけであることを保証するためには位置指 定子をつける
- ^[A-Za-z]\w*\$

浮動小数リテラルをにマッチする正規表現 (1)

ソフトウェア開発 第7回目授業

平野 照比古

則凹の演首

正規表現を作る

正規表現のマッチを研

HTMI 文聿の構成

CSS の≢

DOM OF HE

DOM とは

DOM のメソッド

浮動小数リテラルは次の部分から成り立っている。 [符号][整数部][小数点][小数部][指数部] このうち、[符号] や小数点以下の部分はなくてもよい。

川田(リ)) 東省 正規表現を作る

正規表現を作る日付を得る

正規表現の利用

IIIIVIL 人古V.

CSS 0)7

DOM の利用 DOM とは DOM のメソッド DOM のプロパティ

- ▶ 符号部は+または-からなる一文字からなる。一度だけまで現れてよいので、この部分は [+-]?または (+|-)?で表される。
- 整数部は10進数の並びであり最低1文字は必要であるので反復の指定は+となる。したがって、この部分は d+で表される。
- ▶ 小数点. は正規表現では任意の文字にマッチするので エスケープする必要がある。したがってこの部分は [\.] となる。
- ▶ 小数部は数字が並べられる。全くなくてもよいので反復の指定は*となる。

前回の演

正規表現を作る

正規表現のマッチを確正規表現の利用

HTML 文書の料

CSS の利

DOM の利用

DOM のメソッド

- ▶ 指数部は指数の開始を表す文字 E または e で始まる 10 進数である。数字は最低一つ必要であるのでこの部分 は (E|e)\d+となる。
- ► これらを合わせると求める正規表現が得られる。小数 部などがなくてもよいのでそれらの部分には反復指 定?を付ければよい。
 - ^[+-]?\d+(\.\d*)?((E|e)[+-]?\d+)?\$

浮動小数リテラルをにマッチする正規表現 (4)

ソフトウェア開発 第7回日授業

平野 照比古

則凹の演首

正規表現を作る

正規表現のマッチを確

HTML 文書の構成

CSS の利

DOM ONE

OM とは

OM のメソッド

正式な数値リテラルでは小数点の前に整数部がない .1 など も許しているが、ここではマッチしない。

24 時間制の時刻の表し方

時、分、秒はすべて2桁とし、それらの区切りは:

- 時間は 00 から 23 までであるので時間の初めの文字が 0と1のときと、2のときで分ける必要がある。(数字 として比較はできない)
- ▶ 時間の先頭が0と1のときはそのあとの文字は0から 9 まで取れるので、[01]\d となる。
- ▶ 2ではじまるときは0から3まで取れるので、2[0-3] となる。
- ▶ 同様に、分と秒は先頭の文字が0から5までであるの で [0-5] \d となる。
- ▶ |の範囲を限定するため時間のところの()を忘れない こと。

求めるものは次のとおりである。

- ^([01]\d|2[0-3]):[0-5]\d:[0-5]\d\$
- $([01]\d[2[0-3])(:[0-5]\d){2}$

日付を分ける-ほしい部分を指定 (match を利用)

ソフトウェア開発 第7回目授業

平野 照比古

可回の演習

規表現を作る

日付を得る

正規表現の利用

HIML 文書の作

CSS の利

DOM の利用

DOM とは DOM のメソッド

- ▶ /(\d+) 年 (\d+) 月 (\d+) 日/
 - ▶ 戻り値の 1, 2,3番目に求める値が入る
 - ▶ したがって分割代入 [,y,m,d] で変数に代入ができる。
- ▶ この形ならば/\d+/g で十分

日付を分ける-区切る部分を指定 (split を利用)

ソフトウェア開発 第7回目授業

平野 照比古

前回の演習

E規表現を作る

日付を得る

正規表現の利用

TIML 文音の

CSS の利

DOM の利田

DOM とは DOM のメソッド

- ▶ /年|月|日/ 戻り値の 0, 1, 2番目に求める値が入る
- ▶ 文字クラスでもよいが「平成」が付いた場合には対応 できない
- ▶ /\D+/(数字以外で分ける)

```
"aaaabaaabb".match(/.*b/);
```

1 2 3 4 5 6 7 8 9 10 a a a a b a a a b b

["aaaabaaabb"]

- ▶ 最後にbが来る文字列がマッチ
- ▶ .*は貪欲であるのでできるだけ長い任意の文字列と マッチする
- ▶ 最後の b は 10 番目のものが対応

ソフトウェア開発 第7回目授業

平野 照比古

正規表現のマッチを確認する

前回の演習

規表現を作る 付を得る

正規表現の利用

UTMI +++

CCC DEI

DOM の利用

DOM とは DOM のメソッド DOM のプロパティ "aaaabaaabb".match(/.*b/g);

ソフトウェア開発 第7回目授業

平野 照比古

前回の演

正規表現を作る

正規表現のマッチを確認する 正規表現の利用

HIML 文書の

CSS の利

DOM ONE

JOIM の利用

DOM とは
DOM のメソッド
DOM のプロパティ

["aaaabaaabb"]

前と同じ理由により、結果は同じ

"aaaabaaabb".match(/.*?b/);

1 2 3 4 5 6 7 8 9 10 a a a a b a a a b b

["aaaab"]

- ▶ b で終わる文字列を表している
- ▶ その前の.*?は非貪欲なマッチをする
- ▶ 最後のbは5番目のもの

ソフトウェア開発 第7回目授業

平野 照比古

前回の演習

E規表現を作る 日付を得る

正規表現のマッチを確認する正規表現の利用

HTML 文書の構成

CSS の利

DOM の利用

DOM とは DOM のメソッド DOM のプロバティ

```
"aaaabaaabb".match(/.*?b/g);
```

1 2 3 4 5 6 7 8 9 10 a a a a b b

["aaaab", "aaab", "b"]

- ▶ 前と同様に非貪欲なマッチング
- ▶ gオプションがついているので一つ目以降見つかった 位置から再度マッチするものを探す
- 全体で3つ答えを返す

ソフトウェア開発 第7回目授業

平野 照比古

前回の演習
正規表現を作る

E規表現を作る I付を得る

正規表現のマッチを確認する 正規表現の利用

HIML 文書

CSS の利

DOM の利用

DOM とは DOM のメソッド DOM のプロパティ "abccbcckkccaaMMaacc".match(/((.) $\2$).* $\1$ /);

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 a b c c b c c k k c c a a M M a a c c

["ccbcckkccaaMMaacc", "cc", "c"]

- ▶ 正規表現の(.)は左から数えて2番目の括弧になる
- ▶ この部分にマッチした文字は \2 で参照できる
- ▶ (.)\2 は同じ文字が 2 つ並ぶものにマッチ
- ► この部分全体が再び括弧でくくられているので、この 部分が \1 で参照できる
- ▶ この正規表現は同じ2つの文字で挟まれた文字列に マッチする
- ▶ 中央部の正規表現は任意の文字列を表す
- ▶ 貪欲なマッチなのではじめに現れる同じ文字が2つ続く3,4番目のccが一番最後に現れる18番目と19番目のccと組み合わされる
- ▶ グローバルな検索ではないときには戻り値に \1 と \2 が含まれる。

ソフトウェア開発 第7回目授業

平野 照比古

前回の演習 正規表現を作る

正規表現のマッチを確認する正規表現の利用

ITTIVIL 人言切悟以

C33 (7) (1)

DOM のオリッド

回の演習

E規表現を作る 日付を得る

正規表現のマッチを確認する 正規表現の利用

TITIVIL X = 0.

CSS の利

DOM の利用 DOM とは

 DOM
 とは

 DOM
 のメソッド

 DOM
 のプロパティ

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 a b c c b c c k k c c a a M M a a c c

["ccbcckkccaaMMaacc"]

- ▶ g フラグが付いているが、条件に合うものは一つしかない
- ▶ 戻り値に \1 と \2 が含まれない

CCC OFFIE

DOM の利用

DOM とは DOM のメソッド DOM のプロバティ

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 a b c c b c c k k c c a a M M a a c c

["ccbcc", "cc", "c"]

- ▶ 前問と異なり、中央部の任意の文字列が非貪欲になっている
- ▶ 3, 4番目と 6, 7番目の cc が対応
- ▶ 戻り値の配列の2番目と3番目は \1と \2

"abccbcckkccaaMMaacc".match(/((.) $\2$).*? $\1/g$);

ソフトウェア開発 第7回目授業

平野 照比古

前回の演

規表現を作る I付を得る

正規表現のマッチを確認する 正規表現の利用

HTML 文書の構成

000 07 13713

DOM の利用

DOM とは
DOM のメソッド
DOM のプロパティ

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 abccbcckkccaa M M a a c c

["ccbcc", "ccaaMMaacc"]

g が付いているのでさらに 10 番目と 11 番目、18 番目と 19 番目の cc が対応 "abccbcckkccaaMMaa".match(/((.)\2).*\1/);

ソフトウェア開発 第7回日授業

平野 照比古

正規表現のマッチを確認する

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 abccbcckkccaa M M a a

["ccbcckkcc", "cc", "c"]

- ▶ これまでの文字列から最後の2文字を取り除いた文字 列で同じことを行っている
- ▶ 3.4 番目の cc に対応するのは 11. 12 番目のもの

"abccbcckkccaaMMaa".match(/((.) $\2$).* $\1/g$);

ソフトウェア開発 第 7 回目授業

平野 照比古

回の演習 E規表現を作る

正規表現のマッチを確認する 正規表現の利用

ITTWIL X TOTAL

CSS の利

DOM の利用 DOM とは

DOM とは DOM のメソッド DOM のプロパティ

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 a b c c b c c k k c c a a M M a a

["ccbcckkcc", "aaMMaa"]

- ▶ gフラグが付いているのでマッチした部分列が返される
- ▶ 3,4 番目のと 10,11 番目の cc が対応
- ▶ そのあとの部分列で 12,13 番目と 16,17 番目の aa が対 応する
- ▶ マッチした部分列は2つ

"abccbcckkccaaMMccaa".match(/((.) $\2$).* $\1/g$);

ソフトウェア開発 第7回目授業

平野 照比古

可回の演習 5世末翌5年2

:現表現を作る |付を得る

正規表現のマッチを確認する正規表現の利用

TITIVIL 文音の構成

DOM OTHER

DOM とは DOM のメソッド

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 a b c c b c c k k c c a a M M c c a a

["ccbcckkccaaMMcc"]

- ▶ これまでの文字列と最後の4つが入れ替わっている
- ▶ 3,4 番目と 16,17 番目の cc が対応

回の演習 規表現を作る

正規表現のマッチを確認する

HTML 文書の構成

CSS の利

DOM の利用 DOM とは

DOM とは DOM のメソッド DOM のプロパティ

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 a b c c b c c k k c c a a M M c c a a

["ccbcc", "ccaaMMcc"]

- ▶ 前問と異なり、非貪欲な任意の部分文字列を途中に 取る
- ▶ 3, 4番目と 6, 7番目の cc、10,11番目と 16,17番目の cc が対応する

2回前の授業で次のエラー処理を紹介した。

```
function Person(name, y, m, d){
  if(name === "") throw new Error("名前がありません");
 this.name = name;
 this.year = y;
 if(m<1 || m>12) throw new Error("月が不正です");
 var date = new Date(y,m,0);
 if(d<1 || d>date.getDate()) throw new Error("日が不正です");
 this.month = m,
 this.day = d
Person.prototype = {
```

正規表現の利用

正規表現の利用

HTML 文書の構成

DOM の利用

DOM とは
DOM のメソッド
DOM のプロパティ

このリストでは十分なエラー処理がなされていない。

- ▶ 数に変換されない文字列の入力に対しては値が NaN になる。
- ▶ その結果、たとえば月の値の評価 m<1 は false となり、エラーチェックを通り抜けてしまう。</p>
- ▶ 対処法としては!(m>=1 && m<=12)とすることも考えられるが、これでも小数点付きの数が排除できない。</p>
- ▶ Web アプリケーションではテキストボックスの入力は 文字列になるので、文字列の段階でチェックするほう が楽

Person2の修正

ソフトウェア開発 第7回日授業

平野 昭比古

正規表現の利用

Person オブジェクトの範囲を外部から入力させるときに文 字列を数に変換する前に数値リテラルになっているかを判 定することでプログラムが不正な値を受け付けないように できる。

整数値だけにするのであれば正規表現は/^\d+\$/である。

4日ト4間ト4日ト4日ト ヨーの9个

正規表現の利用

CSS の利用

DOM の利用

DOM とは DOM のメソッド DOM のプロパティ

```
<title>エラーオブジェクト(改良)</title>
function Person(name, y, m, d){
  if(name === "") throw new Error("名前がありません");
 this.name = name;
 this.year = checkNum(y,1900,2020,"年");
 this.month = checkNum(m,1,12,"月");
 var date = new Date(y,this.month,0);
 this.day = checkNum(d,1,date.getDate(),"日");
function checkNum(s, low, high, mes) {
 if((s+"").match(/^\d+$/)) {
   if(s>=low &&s<=high) return s-0;//文字列を数字に変換
 }
 throw new Error(mes+"が不正です"):
```

. . .

前回の演

規表現を作る 付を得る

正規表現の利用

HIVIL 文書の博成

C33 0741/TI

DOM の利用

DOM とは DOM のメソッド

OM のメソッド OM のプロパティ

prompt()で戻ってきた文字列が数字だけからなっているかをチェックしたうえで、与えられた範囲内にあるかを調べている。

外部入力のチェックを!

ソフトウェア開発 第7回目授業

平野 照比古

前回の演

正規表現を作る 日付を得る

正規表現の利用

HTML 文書の構成

CSS の利

DOM の利用

DOM & Lt

DOM のメソッド DOM のプロパティ

- ▶ 外部からのデータの入力に対しては、データを吟味してから利用する
- ▶ 特に Web ページのテキストボックスからのデータ入力 を利用して不正行為を行う手法が知られている

Google Maps の利用 (1)

次のリストは Google Maps を利用して地図を表示するもの

ソフトウェア開発 第7回日授業

平野 照比古

HTML 文書の構成

<meta http-equiv="Content-Type" content="text/html; charset=utf-8\bar{n}/>

src="http://maps.google.com/maps/api/js?sensor=false"></script>

<script type="text/javascript"</pre>

<title>初めてのGoogleMaps</title>

である。 1 <!DOCTYPE html>

2 <html> <head>

6

◆□ ▶ ◆□ ▶ ◆■ ▶ ◆□ ▶ ◆○ ◆○

```
8
       <script type="text/javascript">
         window.onload = function() {
9
          let latlng = new google.maps.LatLng(35.486210,139.341443);
10
          let myOptions = {
11
12
             zoom: 10,
13
             center: latlng,
14
            mapTypeId: google.maps.MapTypeId.ROADMAP
15
        };
        let mapCanvas = document.getElementById("map_canvas")
16
17
         let map = new google.maps.Map(mapCanvas, myOptions);
18
19
      </script>
20
       <link rel="stylesheet" type="text/css" href="map.css" />
21
    </head>
22
    <body>
23
       <div id="map_canvas" ></div>
24
    </body>
25 </html>
```

前回の演

正規表現を作る 日付を得る 正規表現のマッチを確認する 正規表現の利用

HTML 文書の構成

CSS の利

DOM の利用 DOM とは DOM のメソッド

- ▶ 1 行目は HTML 文書の DOCTYPE 宣言である。この形は HTML5 におけるもの
- ▶ 2 行目はこの HTML 文書のルート要素と呼ばれるものである。最後の 26 行目の </html>までが有効となる。
- すべての要素はこの範囲になければならない。

3 行目から始まる<head>はブラウザに表示されない、いろいろな HTML 文書の情報を表す。

- ▶ 4,5 行目はこの文書の形式や文字集合を記述している。 ここでは内容は text/html の形式、つまり、テキスト で書かれた html の形式で書かれていることを表す。¹
- ▶ 6 行目の<title>はブラウザのタブに表示される文字 列を指定している。
- ▶ 6,7 行目は Google Maps のライブラリーを読み込むためのものである。このように JavaScript のプログラムは外部ファイルとすることができる。
- ▶ 9 行目から 19 行目は HTML 文書内に書かれた JavaScript である。詳しい解説は後の授業で行う。
- ▶ 20 行目は HTML 文書の見栄えなどを規定する CSS ファイルを外部から読み込むことをしている。

ソフトウェア開発 第7回目授業

平野 照比古

前回の演習 E規表現を作る

日付を得る 正規表現のマッチを確認する 正規表現の利用

HTML 文書の構成

DOM Ett

DOM のメソッド DOM のプロバティ

¹このような方法でファイルのデータ形式を表すことを MIME(Multipurpose Internet Mail Extension) タイプと呼ぶ。元来、テキストデータしか扱えない電子メールに様々なフォーマットのデータを 扱えるようにする規格である。

HTML 文書で実際にブラウザ内で表示される情報 は<body>要素内に現れる。

- ▶ このリストでは Google Maps を表示するため の<div>要素が一つあるだけである。このとき、 <div>は<body>の子要素であるといい、 <body>は<div>の親要素という。
- ▶ 各要素名または要素の終了を示すタグ (<...>) の間に 文字列がある場合、その部分はテキストノードと呼ば れるノードが作成されている。

各要素は<との中に現れる。初めに現れる文字列が要素名で あり、そのあとに属性と属性値の組がいくつか並ぶ。

- 属性とその属性値は=で結ばれる。
- 属性値は"ではさまれた文字列として記述
- ▶ <script>要素では属性 type と src が設定
- ▶ 24 行目の<div>要素では属性 id に属性値 map_canvas を設定している。なお、この要素は CSS によっても属 性が定義されている。

```
20 行目で参照している CSS ファイル
```

1 #map_canvas{ 2 width:500px; 3 height:500px; float:left; 5 margin:5px 10px 5px 10px; 6 }

- ► CSS の各構成要素は HTML 文書の要素を選択するセレ クタ (ここでは#map_canvas) とそれに対する属性値の 並び([属性]:[属性値];)からなる。
- ▶ #で始まるセレクタはそのあとの文字列を<id>の属性 値に持つ要素に適用される。
- ▶ したがって、ここの規則は24行目の<div>要素に適用
- ▶ その内容は Google Maps が表示される画面の大きさ (width と height)、配置の位置 (float) と要素の外に 配置される空白 (margin) を指定

DOM の利用

DOM とは DOM のメソッド DOM のプロパティ

- ▶ カスケーディングスタイルシート (CSS) は HTML 文書 の要素の表示方法を指定するもの
- ▶ CSS は JavaScript からも制御可能
- 文書のある要素に適用されるスタイルルールは、複数の異なるルールを結合(カスケード)したもの
- スタイルを適用するためには要素を選択するセレクタ で選ぶ。

セレクタの種類は配布資料を参照のこといくつか注意する 点を挙げる。

- ▶ 属性 id の属性値の前に#をつけることでその要素が選ばれる。
- ▶ 属性 class の属性値の前に. をつけることでその要素 が選ばれる。
- ▶ nth-child(n) には単純な式を書くことができる。このセレクタは複数書いてもよい。
- ► EFとE>Fの違いを理解しておくこと。たとえば div div というセレクタは途中に別の要素が挟まれて いてもよい。また、<div>要素が3つある場合にはど のような2つの組み合わせも対象となる。

HTML 文書の構成

000 0711711

DOM の利用 DOM とは

DOM のメソッド DOM のプロバティ

- ▶ Document Object Model(DOM) は HTML 文書などの 要素をノードとしたツリー構造で管理する方法
- ▶ DOM のメソッドやプロパティを使うことで各要素に アクセスしたり、属性値やツリーの構造を変化させる ことが可能
- ▶ DOM の構造は開発者ツールなどで見ることができる。

平野 昭比古

DOM とは

- ▶ Google Chrome では開発者ツールから Elements タブで 確認する。ここで要素上で右クリックして Edit as HTML を選択するとテキストとして編集できる。
- ▶ FireFox では開発ツールから「開発ツールを表示」を選 択し、インスペクタタブで DOM ツリーが確認できる。 要素上で右クリックから「HTMLとして編集」とする とテキストとして編集できる。
- ▶ IE では開発者ツールを開き左にあるタブの一番上にあ る DOM Explorer で同様のことができる。

DOM のメソッド

DOM では DOM ツリーを操作するためにメソッドやプロ パティが規定されている。メソッドとはそのオブジェクト に対する操作である。次のような手段を提供している。

- 条件に合う要素または要素のリストを得る。
- ▶ 要素の属性を参照、変更ができる。
- ▶ 要素を新規に作成する。
- ある要素に子要素を追加したり、取り除いたりする。

回の演習

正規表現を作る 日付を得る 正規表現のマッチを確認

HTML 文書の構成

CSS の利

DOM の利用 DOM とは DOM のメソッド

DOM のメソッド DOM のプロパティ

▶ getElementById(id) 属性 id の値が引数 id である要素を得る。

- ▶ getElementsByTagName(Name) 要素名が Name である要素のリストを得る。リストの 各要素は配列と同様に[]で参照できる。
- ▶ getElementsByClassName(Name) 属性 class の値が Name である要素のリストを得る。 リストの各要素は配列と同様に[]で参照できる。
- ▶ getElementsByName(Name) 属性 name が Name である要素のリストを得る。得られた各要素は配列と同様に「〕で参照できる。

条件に合う要素または要素のリストを得る(2)

ソフトウェア開発 第7回目授業

平野 照比古

前回の演習

正規表現を作る 日付を得る 正規表現のマッチを確

HTML 文書の構成

CSS の利

DOM の利用 DOM とは

DOM のメソッド DOM のプロパティ

querySelector(selectors)
 selectors で指定された CSS のセレクタに該当する一番初めの要素を得る

▶ querySelectorAll(selectors) selectors で指定された CSS のセレクタに該当する要素のリストを得る。配列と同様に[]で参照できる

HTML 文書の

C55 の利用

DOM の利用 DOM とは DOM のメソッド

- ▶ getAttribute(Attrib) 対象要素の属性 Attrib の値を得る。戻り値の型は文字列である。
- ▶ setAttribute(Attrib, Val) 対象要素の属性 Attrib の値を Val にする。数を渡しても文字列に変換される。
- ► hasAttribute(Attrib) 対象要素に属性 Attrib がある場合は true を、ない場合は false を返す。
- ▶ removeAttribute(Attrib)
 対象要素の属性 Attrib を削除

要素を新規に作成する

- ▶ createElement(Name)
 Name で指定した要素を作成
- ▶ createElementNS(NS,Name)
 名前空間 NS で定義されている要素 Name を作成
- ▶ createTextNode(text) text を持つテキストノードを作成
- ▶ cloneNode(bool) bool が true のときは対象要素の子要素すべてを、 false のときは対象要素だけの複製を作る。

ソフトウェア開発 第7回目授業

平野 照比古

前回の演習正規表現を作る

正規表現のマッチを確認では表現の利用

HTML 文書の構成

CSS の利

DOM の利用

DOM をは

DOM のメソッド

DOM のプロパティ

DOM のメソッド

- ▶ 指定した要素が定義されている規格を指定するもの
- ▶ 一つの文書内で複数の規格を使用する場合、作成する 要素がどこで定義されているのかを指定
- 異なる規格で同じ要素名が定義されていてもそれらを 区別することが可能
- ▶ 通常の HTML 文書では http://www.w3.org/1999/xhtml を指定

▶ insertBefore(newElm, PElm) 対象要素の子要素 PElm の前に newElm を子要素として 付け加える。Elm がすでに対称要素の子要素のときは 元の位置から指定された位置に移動

- ▶ removeChild(Elm) 対象要素の子要素 Elm を取り除く。
- ▶ replaceChild(NewElm, OldElm) 対象要素に含まれる 子要素 OldElm を NewElm で置き換える。
- ▶ setValue(value)
 対象のテキストノードの値を value にする。

別 回 の 決 音 正規表現を作る 日付を得る 正規表現のマッチを確認す。

HTML 文書の構成

CSS の利)

DOM とは **DOM のメソッド** DOM のプロパティ

DOM のメソッド

```
<body>
  <form id="menu">
   <select>
      <option value="1">1 月</option>
      <option value="2">2 月</option>
      <option value="3">3 月</option>
      <option value="4">4 月</option>
      <option value="5">5 月</option>
      <option value="6">6 月</option>
      <option value="7">7 月</option>
      <option value="8">8 月</option>
      <option value="9">9 月</option>
      <option value="10">10 月</option>
      <option value="11">11 月</option>
      <option value="12">12 月</option>
   </select>
  </form>
</body>
```

- ▶ ユーザからの入力を受け付ける要素は通常、<form>要素内に記述
- ▶ プルダウンメニュー の要素名は<select>
- ▶ 選択する内容は<option>要素
- ➤ <option>要素の属性 value の値が選択した値として 利用できる。
- ▶ <option>要素内の文字列 (テキストノード) がプルダウンメニューに表示される
- ► <select>は<form>の子要素であり、 各<option>は<select>の子要素

プルダウンメニュ——プログラムで作成 (1)

7 //<! [CDATA [

8

10

window.onload = function(){

```
ソフトウェア開発
第7回日授業
```

平野 照比古

```
let Select = document.createElement("select");
```

document.getElementById("menu").appendChild(Select);

4日ト4間ト4日ト4日ト ヨーの9个

プルダウンメニュ——プログラムで作成 (2)

ソフトウェア開発 第7回目授業

平野 照比古

```
11
     for(let i=1;i<=12;i++) {
12
       13
       Option.setAttribute("value",i);
       Select.appendChild(Option);
14
       Option.appendChild(document.createTextNode(i+"月"));
15
16
17
                                                 DOM のメソッド
18 //11>
19 </script>
20 </head>
21 <body>
   <form id="menu"></form>
22
23 </body>
24 </html>
```

▶ 8 行目の window.onload はファイルのロードが終わっ た後に発生するイベントを表す。function()が設定 されているのでこの関数がロード後、実行

- ▶ 9行目では<select>要素を作成している。
- ▶ 10 行目では 22 行目にある<form>要素を得て 9 行目で 作成した<select>要素を<form>要素の子要素に設定 している。
- ▶ 11 行目から始まる for ループで 12 個の<option>要素 を作成し、<select>要素の子要素としている。
 - ▶ 12 行目で<option>要素を新規に作成している。
 - ▶ 13 行目で、その要素の属性 value に値を設定して いる。
 - ▶ 14 行目ではその<option>要素を<select>要素の子要 素としている。
 - さらに、15 行目では表示する文字列をもつテキスト ノードを作成し、それを<option>要素の子要素として いる。

DOM のプロパティ

Table: DOM 要素に対するプロパティ(1)

プロパティ名	説明
firstChild	指定された要素の先頭にある子要素
lastChild	指定された要素の最後にある子要素
nextSibling	指定された子要素の次の要素
previousSibling	現在の子要素の前にある要素
parentNode	現在の要素の親要素
hasChildNodes	その要素が子要素を持つかどうか
nodeName	その要素の要素名前
nodeType	要素の種類 (1 は普通の要素、3 はテ
	キストノード)
nodeValue	(テキスト) ノードの値
childNodes	子要素の配列

DOM のプロパティ

Table: DOM 要素に対するプロパティ(2)—DOM4

プロパティ名	説明
children	子要素のうち通常の要素だけ からなる要素の配列
firstElementChild	指定された要素の先頭にある
1	通常の要素である子要素 指定された要素の最後にある
lastElementChild	通常の要素である子要素
nextElementSibling	指定された子要素の次の通常 の要素
previousElementSibling	現在の子要素の前にある通常 の要素