

## JavaScript 練習問題

### 目次

1. はじめに	1
2. 基本構文	2
3. オブジェクト	7
4. DOM	12

## 1. はじめに

HTML、JavaScript、CSS の各ファイルは、それぞれ以下のフォルダ内に格納してあります。

- ・ html フォルダ : HTML ファイル (.html) の格納先
- ・ js フォルダ : JavaScript ファイル (.js) の格納先
- ・ css フォルダ : CSS ファイル (.css) の格納先

ファイルを新規作成する際は、該当するフォルダに保存してください。

## 2. 基本構文

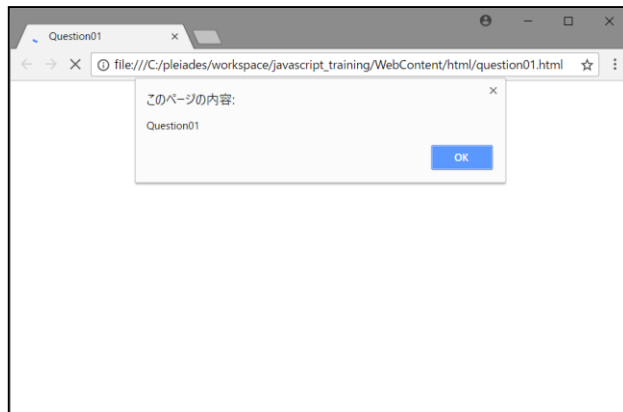
### Question 01

question01.html の<script>タグに下記の JavaScript のコードを追記し、実行結果と同じ警告ダイアログが出力されることを確認してください。

追記するコード

```
window.alert("Question01")
```

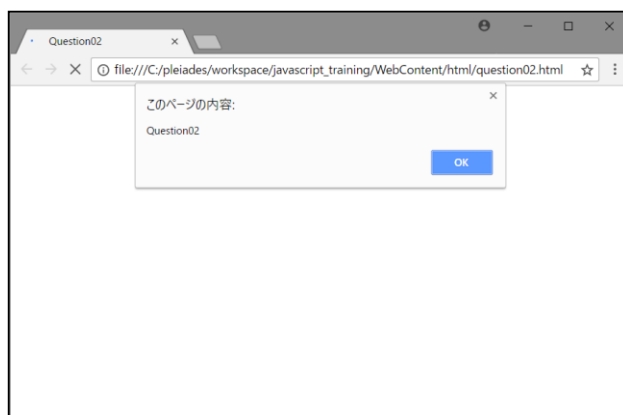
実行結果



### Question02

question02.html を編集し、外部ファイル quesiton.js を読み込めるようにしてください。  
そして、実行結果と同じ警告ダイアログが出力されることを確認してください。

実行結果

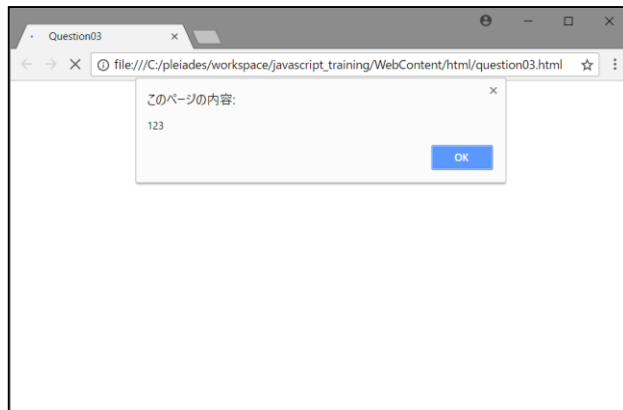


### Question03

question03.html の<script>タグ内に以下の処理を順番通りに実装してください。  
そして、実行結果と同じ警告ダイアログが出力されることを確認してください。

1. 変数 number を宣言する。
2. 変数 number に数値 123 を代入する。
3. 変数 number の値を警告ダイアログに出力する。

#### 実行結果

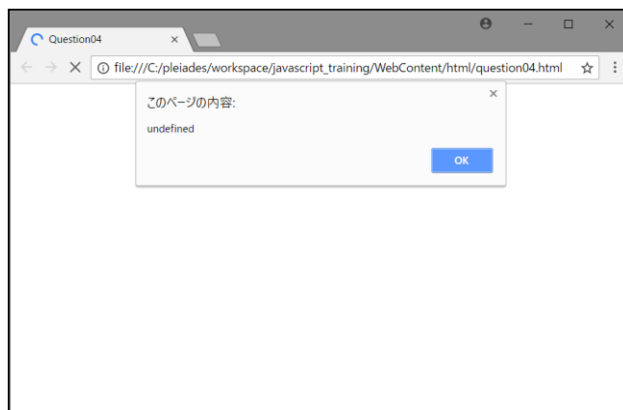


### Question04

question04.html の<script>タグ内に以下の処理を順番通りに実装してください。  
そして、実行結果と同じ警告ダイアログが出力されることを確認してください。

1. 変数 number を宣言する。
2. 変数 number の値を警告ダイアログに出力する。

#### 実行結果



## Question05

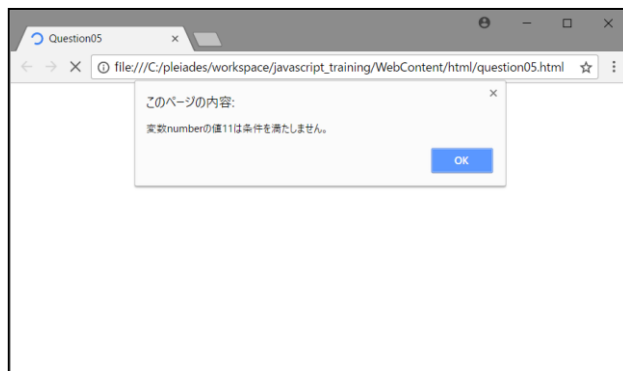
question05.html の<script>タグ内に以下の処理を順番通りに実装してください。  
そして、実行結果と同じ警告ダイアログが出力されることを確認してください。

1. 変数 number を宣言し任意の数値を代入する。
2. 変数 number の値が「1 以上 10 以下」である場合、条件を満たす旨を警告ダイアログで出力する。
3. 変数 number の値が「1 以上 10 以下」ではない場合、条件を満たさない旨を警告ダイアログで出力する。

実行結果（変数 number の値が条件を満たす場合）



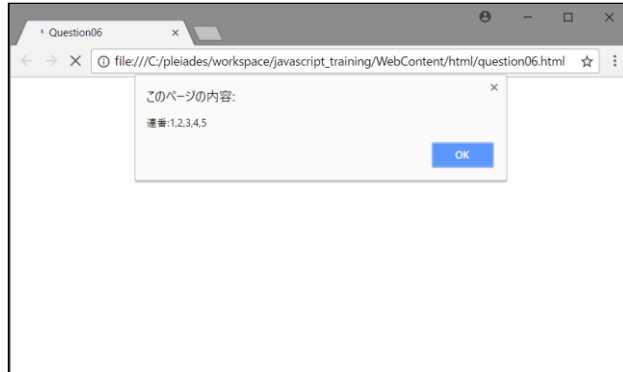
実行結果（変数 number の値が条件を満たさない場合）



## Question06

question06.html の<script>タグ内を編集し、1 から 5 までの連番を警告ダイアログで出力してください。  
ただし、ソースコード内では for 文を必ず使用してください。

### 実行結果

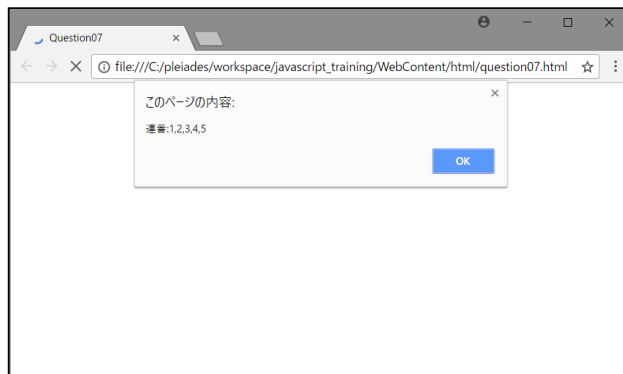


## Question07

question07.html の<script>タグ内を編集し、配列に格納された 1 から 5 までの連番を警告ダイアログで出力してください。

ただし、ソースコード内では for 文を必ず使用してください。

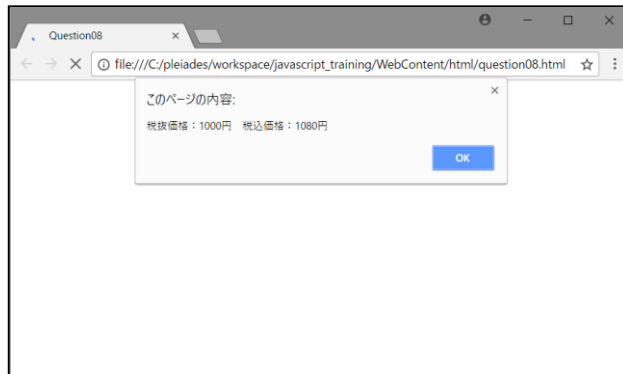
### 実行結果



## Question08

question08.js を編集し、引数で指定された金額の税込価格を戻り値として返す関数 tax() を定義してください。  
また、question08.html の<script>タグ内に、tax() を呼び出して、その処理結果を警告ダイアログで出力する処理を記述してください。

### 実行結果

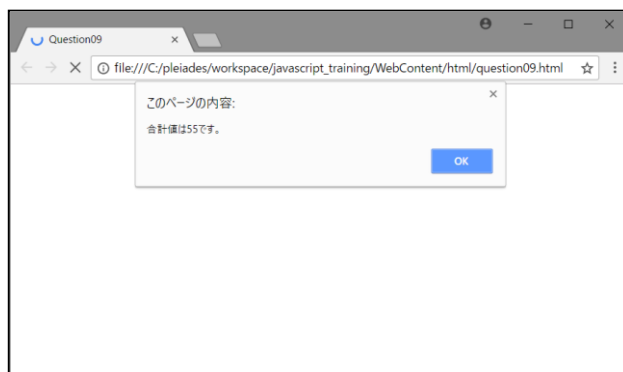


## Question09

question09.js を編集し、引数で指定された配列の各要素の値を合算し、その合計値を戻り値として返す関数 arraySum() を定義してください。

また、question09.html の<script>タグ内に、arraySum () を呼び出して、その処理結果を警告ダイアログで出力する処理を記述してください。

### 実行結果（各要素の値が 20、30、5 の場合）



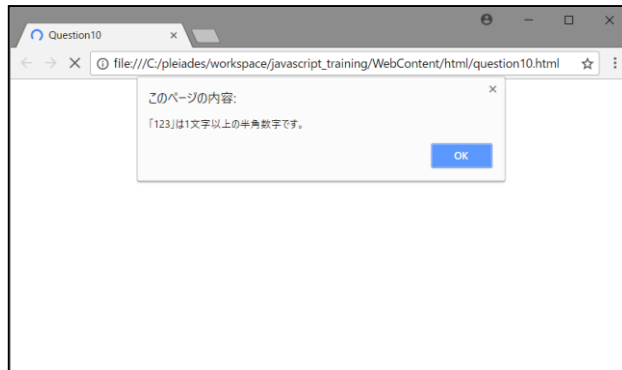
### 3. オブジェクト

#### Question10

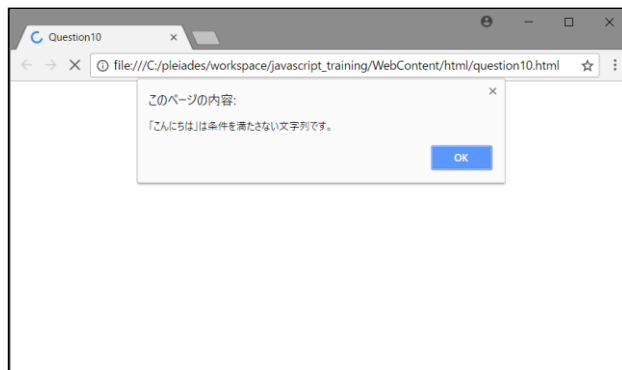
question10.js を編集し、引数で渡された値が 1 文字以上の半角数字である場合は true を、そうでない場合は false を返す関数 stringCheck() を定義してください。

また、question10.html の<script>タグ内に、stringCheck () を呼び出して、その処理結果を警告ダイアログで出力する処理を記述してください。

実行結果（1 文字以上の半角数字の場合）



実行結果（1 文字以上の半角数字ではない場合）





## Question11

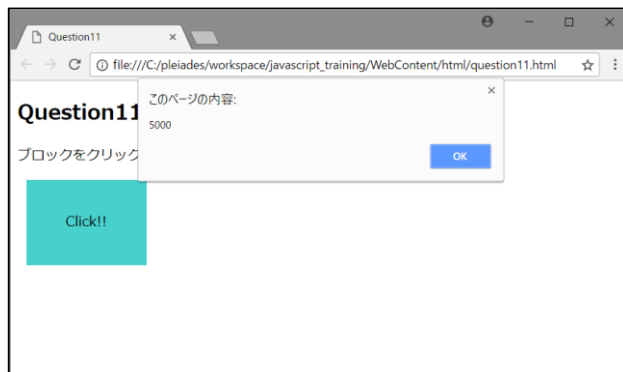
question11.js を編集し、「100 \* 50」の計算結果を警告ダイアログで出力する関数 showCalc() を定義してください。

また、question11.html にて、class 属性の値が "calc" である <div> タグのブロックに、クリックされると関数 showCalc() を呼び出すイベントハンドラを追加してください。

実行結果（クリック前）



実行結果（クリック後）

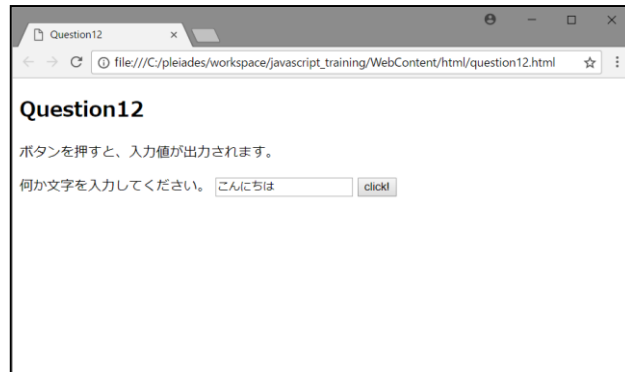


## Question12

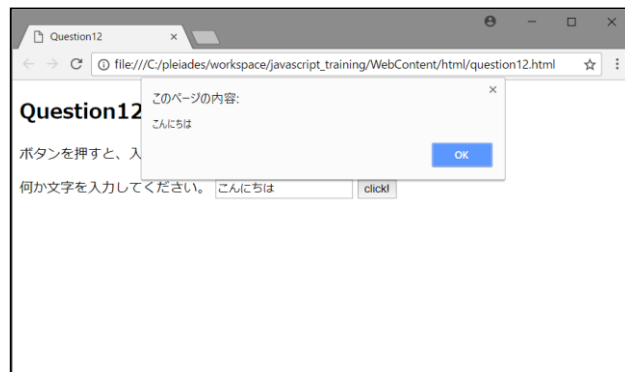
question12.js を編集し、name 属性の値が"message"であるテキストボックスの入力値を取得し、警告ダイアログに出力する関数 showMessage()を定義してください。

また、question12.html にて、value 属性の値が"Click!"であるボタンに、クリックされると関数 showMessage()を呼び出すイベントハンドラを追加してください。

### 実行結果（クリック前）



### 実行結果（クリック後）

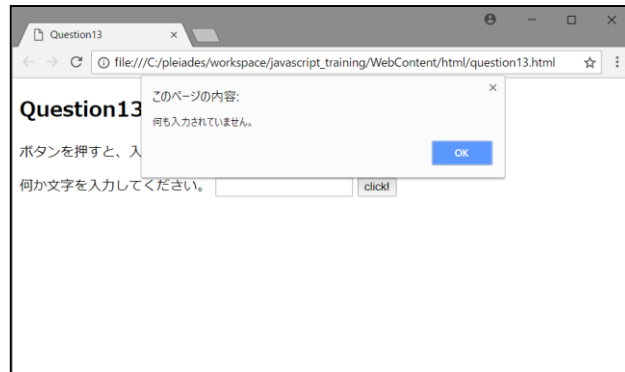


## Question13

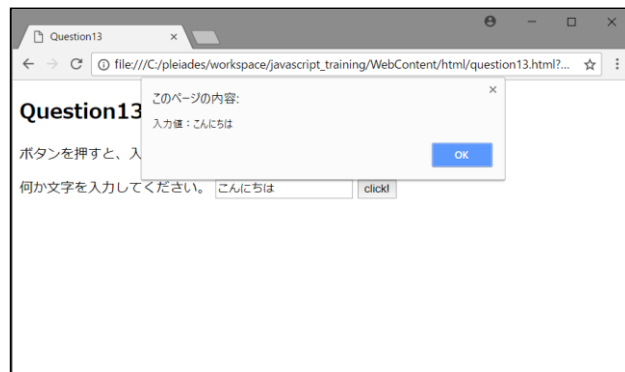
question13.js を編集し、name 属性の値が"message"であるテキストボックスの入力値を取得し、入力値の文字数に応じた警告ダイアログを出力する関数 showMessage()を定義してください。

また、question13.html にて、value 属性の値が"Click!"であるボタンに、クリックされると関数 showMessage()を呼び出すイベントハンドラを追加してください。

実行結果（入力値の文字数が 0 文字だった場合）



実行結果（入力値の文字数が 1 文字以上だった場合）



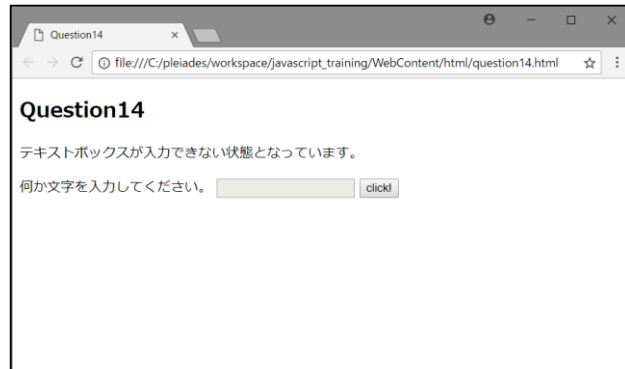
## Question14

question14.js を編集し、name 属性の値が"message"であるテキストボックスを操作無効にする関数 controlForm()を定義してください。

また、question14.html の<body>タグに、ページが読み込まれたときに controlForm()を呼び出すイベントハンドラを追加してください。

最後に、実行結果中のテキストボックスが入力できない状態であることを確認してください。

### 実行結果



## 4. DOM

### Question15

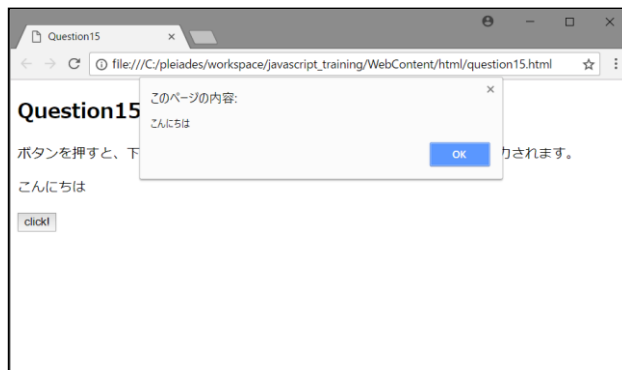
question15.js を編集し、id 属性の値が"message"である<p>タグに囲まれたテキストを取得し、警告ダイアログで出力する関数 getText()を定義してください。

また、question15.html にて、value 属性の値が"Click!"であるボタンに、クリックされると関数 getText()を呼び出すイベントハンドラを追加してください。

実行結果（クリック前）



実行結果（クリック後）

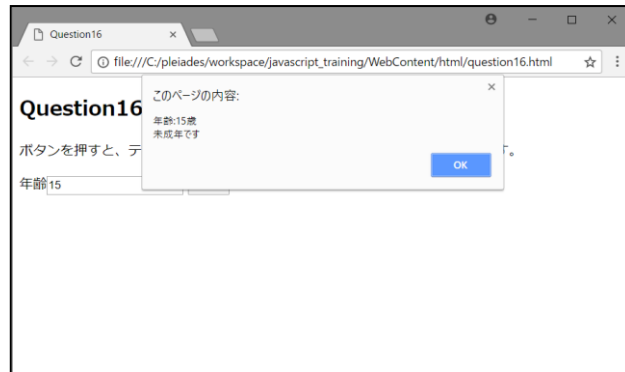


## Question16

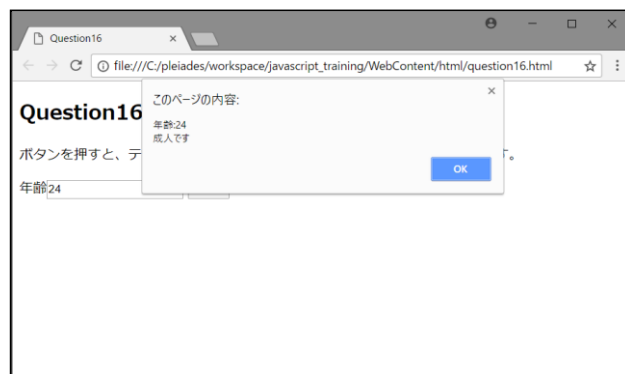
question16.js を編集し、テキストボックスに入力された年齢に応じた結果を警告ダイアログで出力する関数 ageCheck() を定義してください。ただし、入力値の取得には DOM を使用してください。

また、question16.html にて、value 属性の値が "Click!" であるボタンに、クリックされると関数 ageCheck() を呼び出すイベントハンドラを追加してください。

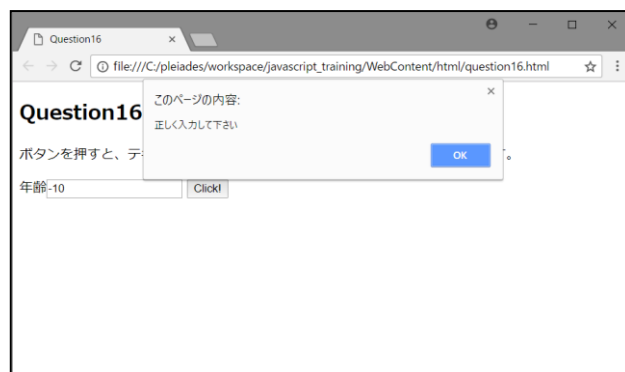
実行結果（年齢が 0 以上 20 未満の場合）



実行結果（年齢が 20 以上の場合）



実行結果（年齢が 0 未満の場合）



## Question17

question17.js を編集し、id 属性の値が"message"である<p>タグに囲まれたテキストの内容を、引数で指定された文字列で出力されるリンクに置換する関数 replaceText()を定義してください。

置換後の HTML タグ : <a href='http://www.3sss.co.jp/tis/'>〇〇〇</a>

※「〇〇〇」には引数で指定された文字列が入ります。

また、question17.html にて、value 属性の値が"Click!"であるボタンに、クリックされると関数 replaceText() を呼び出すイベントハンドラを追加してください。※引数には任意の文字列を指定してください。

### 実行結果（クリック前）



### 実行結果（クリック後）



## Question18

question18.js を編集し、以下の関数を定義してください。

- ・ id 属性の値が"detaild"である<p>タグ内に、りんごの詳細情報に挿入する関数 showDetail()
- ・ id 属性の値が"detaild"である<p>タグ内に、空文字を挿入する関数 hideDetail()

また、question18.html にて、「価格:108 円」というテキストを囲む<p>タグに対して、以下のイベントハンドラを追加してください。

- ・ マウスカーソルの位置がテキストに重なると、関数 showDetail ()を呼び出すイベントハンドラ
- ・ マウスカーソルの位置がテキストの範囲から外れると、関数 hideDetail()を呼び出すイベントハンドラ

実行結果（「価格:108 円」のテキストにマウスカーソルが重なった場合）



実行結果（「価格:108 円」のテキストからマウスカーソルが外れた場合）





## Question19

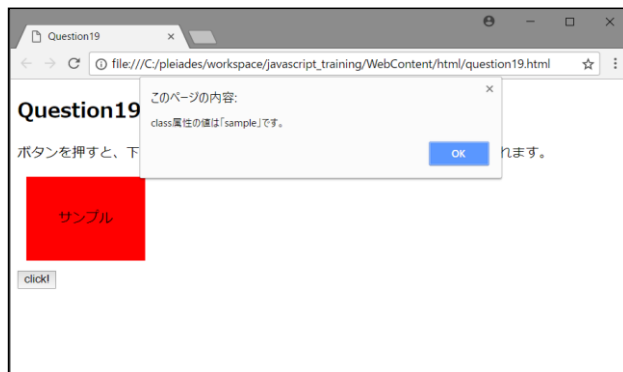
question19.js を編集し、id 属性の値が"block"である<div>タグから class 属性の値を取得し、警告ダイアログで出力する関数 getClassValue ()を定義してください。

また、question19.html にて、value 属性の値が"Click!"であるボタンに、クリックされると関数 getClassValue ()を呼び出すイベントハンドラを追加してください。

### 実行結果（クリック前）



### 実行結果（クリック後）



## Question20

question20.js を編集し、id 属性の値が"block"である<div>タグの class 属性の値を"change"に変更する関数 changeClassValue ()を定義してください。

また、question20.html にて、value 属性の値が"Click!"であるボタンに、クリックされると関数 changeClassValue ()を呼び出すイベントハンドラを追加してください。

最後に、実行結果中のブロックの背景色が赤色から青色に変化することを確認してください。

### 実行結果（クリック前）



### 実行結果（クリック後）

