複数の処理をまとめよう
同じ処理は２度書かない！

★☆☆　　制作目標時間：15分　　js-002

WORK　関数を使用した、効率のよい処理方法を記述してみましょう

課題目的

JavaScriptでのイベント(マウスオーバーイベント、クリックイベント等)のアクションを
体感し、サイト制作時の挙動を制御する基礎を身に付ける
処理をひとまとめ（関数化）にし、より効率の良いソースを記述出来る

課題要項

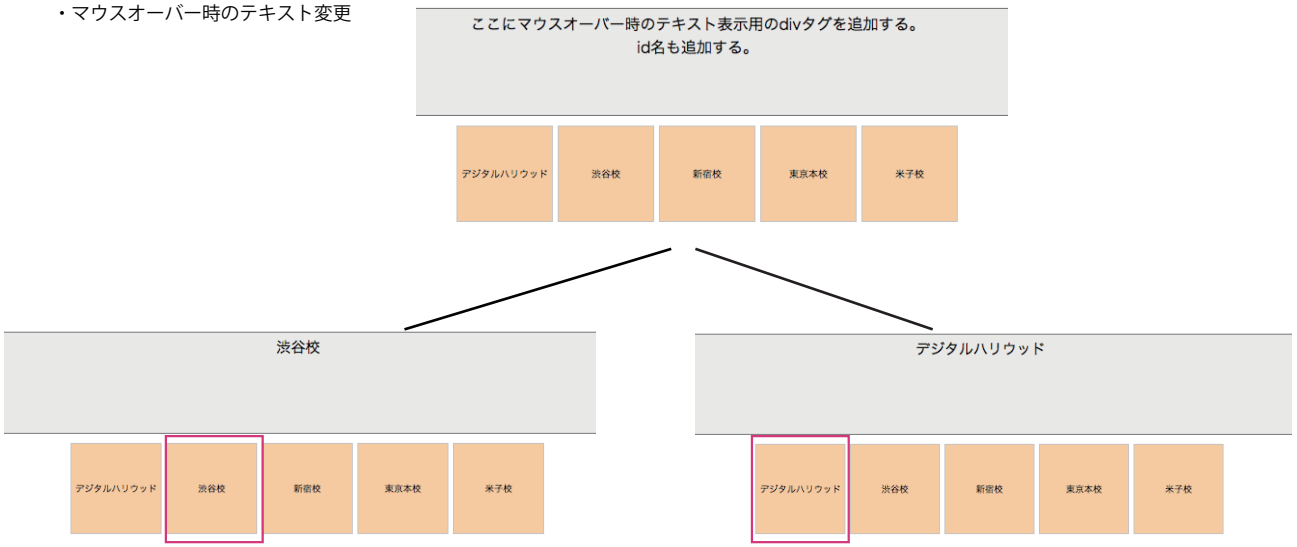
１：cssフォルダ内にあるcssファイル（style.css）を読み込みましょう
２：jsフォルダ内にあるjsファイル（script.js）を読み込みましょう
３：HTML上のliタグにマウスオーバーイベントを登録しましょう
４：今回の課題で使用するjs
- **document.getElementById（id名）←　id名の取得**
- **document.getElementById（id名）.innerText　←　対象idタグ内のテキスト**
- **document.getElementById（id名）.innerHTML　←　対象idタグ内のHTML**

５：不要な改行・インデントなどを意識して、綺麗なソースを心掛けましょう

サンプル

表示完成例

・マウスオーバー時のテキスト変更



このエリアにマウスオーバー　　このエリアにマウスオーバー

作業手順

2ページ目に記載

複数の処理をまとめよう：作業手順（1）

同じ処理は２度書かない！

WORK　関数を使用した、効率のよい処理方法を記述してみましょう

作業手順

1：index.htmlにcssフォルダ内のstyle.cssとjsフォルダ内のscript.jsを読み込む

css、JavaScriptファイルは、js002_dataフォルダ内のcssフォルダ、jsフォルダでそれぞれ管理しています。
※js002_dataフォルダ内で作業してください。

2：HTML上のliタグにマウスオーバーイベントを記述関数名（自由に名前を付ける 例：changeText()）を登録し、引数に自分自身（liタグ）のid名を記述する

引数とは、呼び出す関数に値（主に数字や文字列など）を渡す事が出来ます。
関数に自分のid名を渡し、そのid名を使用して処理をさせます。

記述例）
mouseover="changeText(引数として渡す値);"

3：（2）で登録した関数名と同じ名前で、jsファイルに関数の実態を記述する

引数で、渡された値（今回はid名）を取得する処理を忘れず記述しましょう。
下記の「○○○」には自分で自由な名前を付ける事が出来ます。
その関数の中でその値を使用する場合は、必ず「○○○」と同じ名前を使用しましょう。

記述例）

```
changeText(○○○){
    console.log(○○○);
}
```

4：作った関数の中に、マウスオーバーしたliタグの要素の取得処理を記述する

引数で渡された値をdocument.getElementByld(id名)で使用しましょう。

※JavaScriptでは、「'（シングルコーテーション）」や「"（ダブルコーテーション）」で囲われている文字は文字として、囲われていない文字は変数として扱われますので注意しましょう。下記の表示をコンソールで確認してみましょう。

記述例）

```
var str = "文字列";
console.log('str');  ←　コーテーションあり
console.log(str);  ←　コーテーションなし
```

複数の処理をまとめよう：作業手順（2）

同じ処理は２度書かない！

WORK　関数を使用した、効率のよい処理方法を記述してみましょう

作業手順

5：取得した要素のHTML上のテキストを取得する

HTML上のテキストを取得する場合はinnerTextを使用しましょう。

記述例）
取得した要素.innerText;　←　要素のテキスト取得

6：HTML上のid名「changeArea」のテキストを（5）で取得した値で置き換える

id名「changeArea」の要素を取得し、innerTextで置き換える処理を記述しましょう。
innerTextは、
・HTML上のテキストを取得する場合
・HTML上のテキストを変更する場合
のどちらにも使用できます。

※「innerText」の動作確認に関しては、Firefox以外のブラウザで行なって下さい。
「innerText」はIE独自仕様でFirefox以外のブラウザがサポートしています。
Firefoxでの動作確認を行う場合は、「innerText」の代わりに「textContent」を使用して下さい。
「textContent」はIE以外のブラウザがサポートしています。

7：完成後はトレーナーのチェックを受ける

liを追加してもjsファイルは変更する事なく同じ動作をするかどうかも試してみて下さい。
さらに理解が深まると思います。