

演習（空き席状況チェック）

■ 課題

以下の処理、実行例に従って、空き席状況チェックに以下の機能を追加せよ。

- ・ 講座の表示機能の修正（JSON データから読み込みを実施）
- ・ 講座の新規登録機能の追加
- ・ 講座の編集機能の追加
- ・ 講座の削除機能の追加
- ・ 講座情報のダウンロード機能の追加

■ 処理（講座の表示機能の修正）

- ① 画面表示時に JSON から講座情報を取得し、画面に表示する。

■ 処理（講座の新規登録機能）

- ① 新規登録ボタンをクリック → 講座名の入力フォームをモーダルで表示
- ② 講座名を入力し、書き込むボタンをクリック → モーダルが閉じられ、登録した講座を画面に表示。講座情報オブジェクトに対してもデータ追加。

■ 処理（講座の編集機能）

- ① 編集ボタンをクリック → 講座名の入力フォームをモーダルで表示
- ② 講座名を入力し、書き込むボタンをクリック → モーダルが閉じられ、該当の講座名称を変更。講座情報オブジェクトに対してもデータ変更を実施。

■ 処理（講座の削除機能）

- ① 削除ボタンをクリック → 該当の講座を画面、講座情報オブジェクトから削除。

■ 処理（講座情報のダウンロード機能の追加）

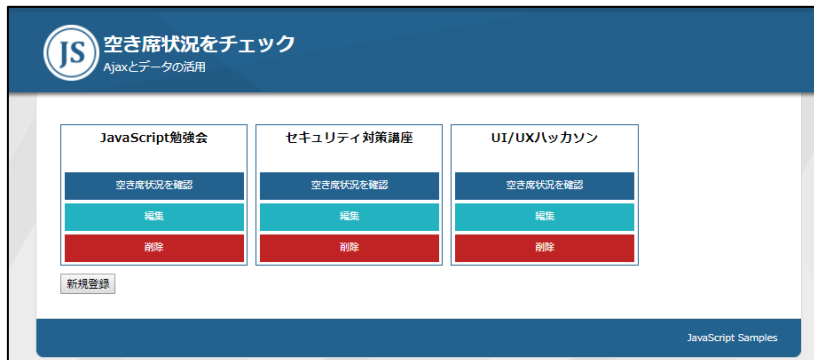
- ① ダウンロードボタンをクリック → 講座情報オブジェクトを JSON データとしてダウンロード。（サンプルコードは 3 ページ目に記載）

■ 条件

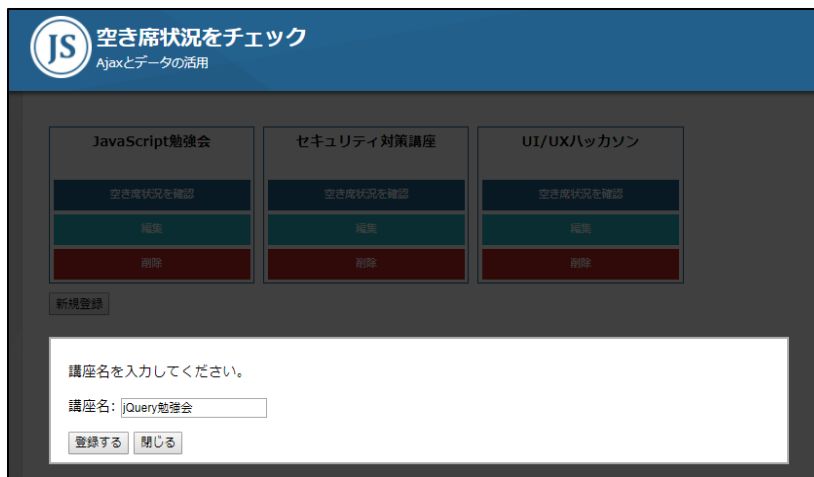
- ・ 新規登録・編集の入力フォームをモーダルで表示すること
- ・ 講座の名称に対し以下の入力チェック実装すること
必須チェック、最大桁チェック（20 文字）
- ・ 講座情報オブジェクトを生成し、講座情報を保持すること(id, name, crowded)
- ・ 空き席状況 (crowded) の初期値は "no"

■ 実行例

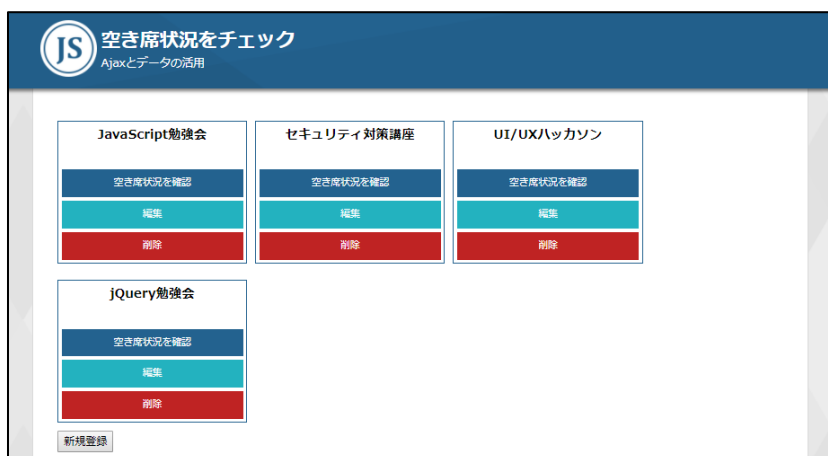
- 画面表示時



- 新規登録、編集



- 登録完了後



■ダウンロード機能サンプル

```
$('#download').on('click',function(){ //ダウンロードボタン
    /** ダウンロードデータの作成 **/
    var text = JSON.stringify({"a":"a"});
    var blob = new Blob([text], {type: "application/json"});

    /** ダウンロードリンクの作成 **/
    var a = document.createElement("a");
    a.href = URL.createObjectURL(blob);
    a.target = '_blank';
    a.download = 'data.json';
    a.click();
});
```