

Angular8におけるBootstrap4のTabsetの利用方法

2019.7.16

by *Shuichi Ohtsu*



フルソースコード

<https://github.com/Ohtsu/ng8-bootstrap4-tabset>

ビデオ解説(日本語)

https://youtu.be/IcL_4Xo2lz0

ビデオ解説(英語)

<>

プロジェクトの作成方法について

ここでは、Angular8におけるBootstrap4のTabsetの利用方法についてご紹介します。

なお、プロジェクト作成の段階からのBootstrap4の設定方法につきましては、すでに、*Angular8におけるBootstrap4環境の設定方法*<https://youtu.be/gF xuSi4IZjM>を公開しておりますので、そちらをご覧ください。

今回は、そうした設定済みプロジェクトをGitHubからダウンロードし、さらにそれにTabset機能を組み込む形で進めていきます。

プロジェクトの内容については、そのフルソースコードをご覧ください。

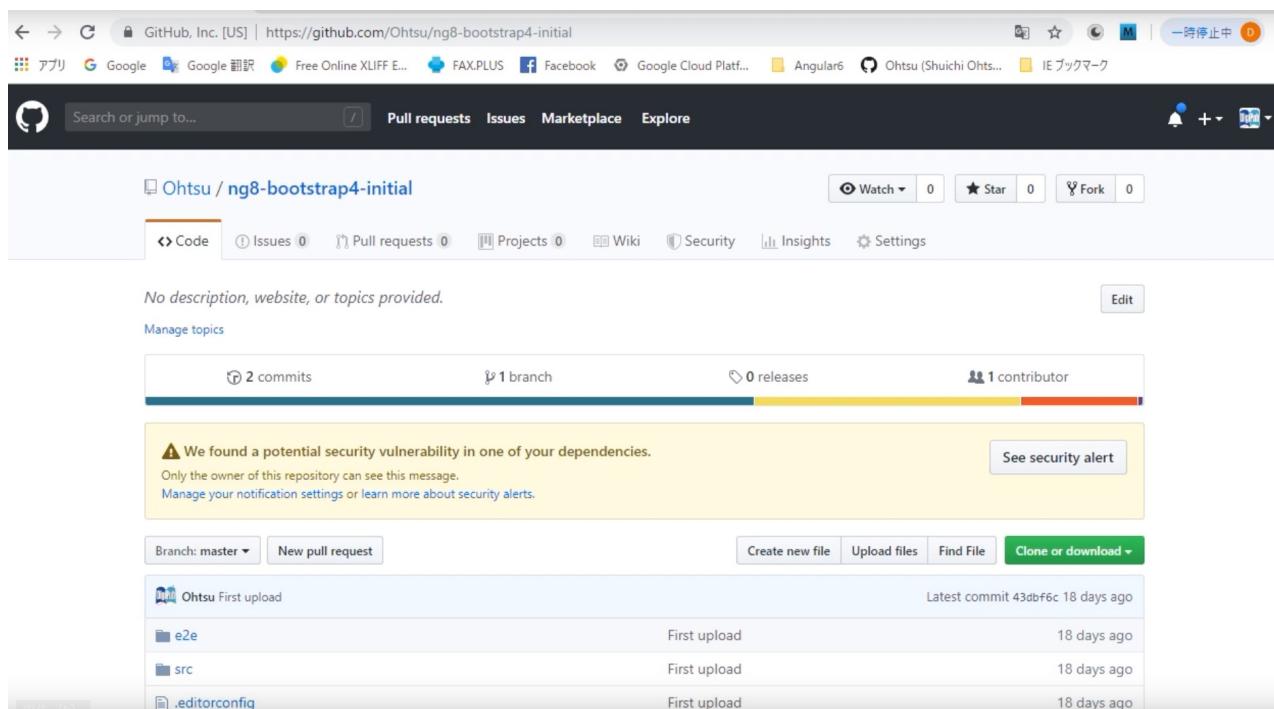
Bootstrap4を組み込んだプロジェクト・ソースのGitHubからのダウンロード

まず、GitHubからBootstrap4を組み込んだプロジェクトのフルソースをダウンロードします。

そのURLは、

<https://github.com/Ohtsu/ng8-bootstrap4-initial>

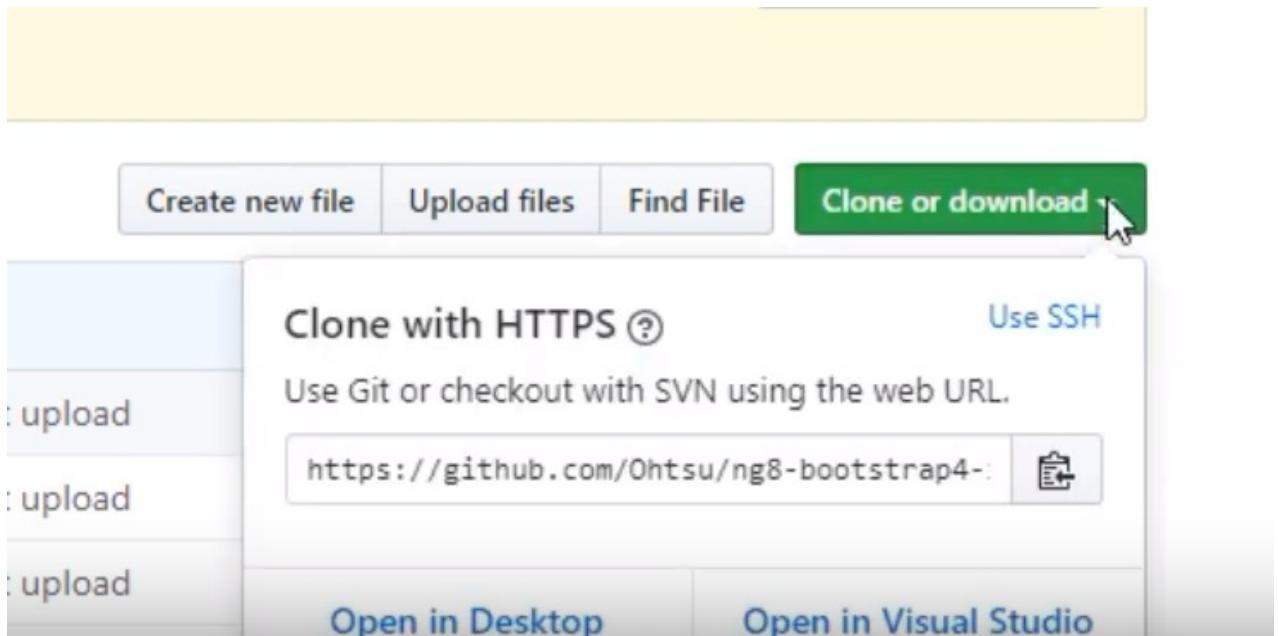
です。



The screenshot shows a GitHub repository page for 'Ohtsu / ng8-bootstrap4-initial'. At the top, there's a navigation bar with links for Pull requests, Issues, Marketplace, and Explore. Below the header, the repository name is displayed along with a search bar. A prominent yellow warning box in the center of the page states: '⚠ We found a potential security vulnerability in one of your dependencies. Only the owner of this repository can see this message. Manage your notification settings or learn more about security alerts.' To the right of this box is a 'See security alert' button. At the bottom of the page, there's a list of files: 'Ohtsu First upload', 'e2e', 'src', and '.editorconfig'. Each file entry includes a small icon, the file name, its status ('First upload'), and the date ('18 days ago'). At the very bottom of the page, there are several buttons: 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find File', and a large green 'Clone or download' button.

ページが表示されたら、右側の緑の*Clone or download*ボタンをクリックします。

するとダウンロード用のアドレスが表示されますので、右側のコピーアイコンをクリックし、そのアドレスをコピーしておきます。



プロジェクトの生成

コマンド・プロンプトを開き、適当なディレクトリに移動した後、そのディレクトリで、`git clone`に続けて、コピーしたアドレスを貼り付けます。

```
git clone https://github.com/Ohtsu/ng8-bootstrap4-initial.git
```

と入力します。

するとダウンロードが開始されます。

```
ohtsu@Tiger64:/mnt/c/mvprz/test/git$ git clone https://github.com/Ohtsu/ng8-bootstrap4-initial.git
Cloning into 'ng8-bootstrap4-initial'...
```

Visual Studio Codeの起動

次に、生成されたプロジェクトのディレクトリに入り、

```
cd ng8-bootstrap4-initial
```

Visual Studio Codeを起動します。

```
code .
```

と入力します。

```
$ cd ng8-bootstrap4-initial  
$ code .
```

VSCodeが起動しましたら、

ctrl+@キーを押し、ターミナル・ウィンドウを開きます。



そして、

```
npm i
```

と入力し、プロジェクトに必要なライブラリをインストールします。

このプロセスには若干時間を要します。

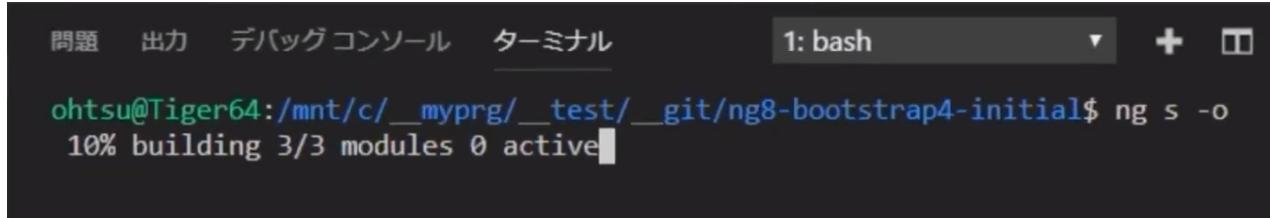
```
問題 出力 デバッグコンソール ターミナル  
1: bash  
ohotsu@Tiger64:/mnt/c/_myprg/_test/_git/ng8-bootstrap4-initial$ npm i  
[██████████] \ extract:@webassemblyjs/helper-wasm-bytecode: sill extract @web
```

プロジェクトの起動の確認

次に、プロジェクトが起動できるかどうかを確認します。

```
ng s -o
```

と入力します。

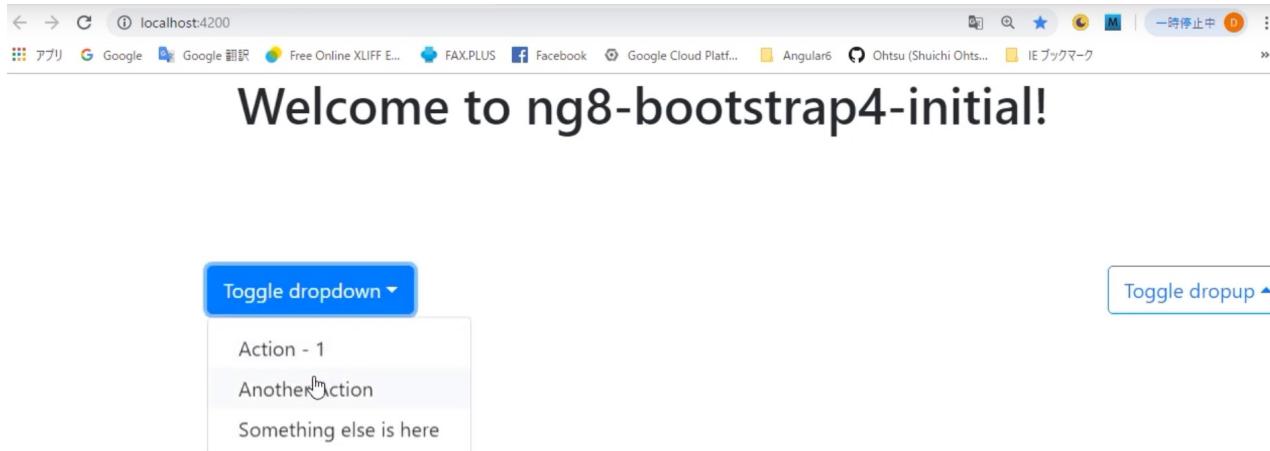


```
問題 出力 デバッグコンソール ターミナル 1: bash ▾ + ■
```

```
ohtsu@Tiger64:/mnt/c/_myprg/_test/_git/ng8-bootstrap4-initial$ ng s -o
```

```
10% building 3/3 modules 0 active
```

ブラウザが自動的に開き、ドロップダウン及びドロップアップが表示されます。



OKです。

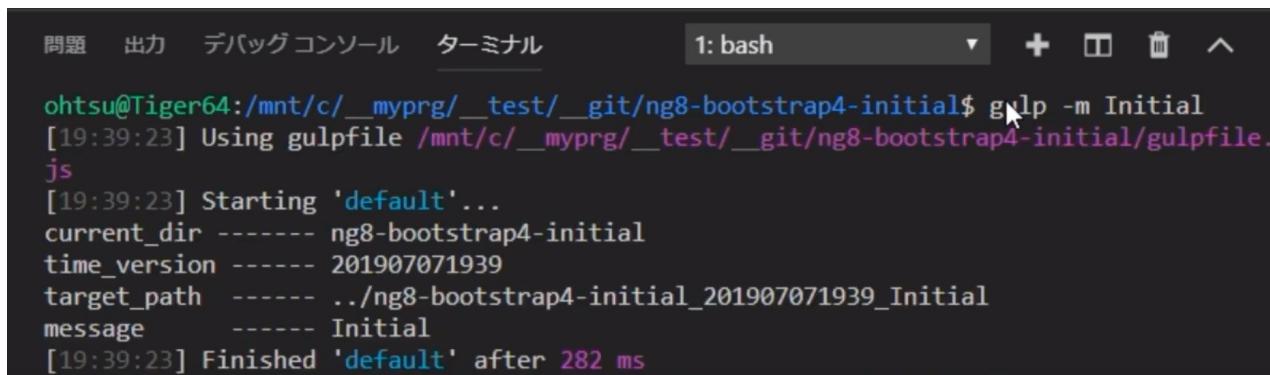
Gulpによるバックアップの作成

この段階で、変更前ソースのバックアップを取っておきます。但し、関連ライブラリまでバックアップをする必要はないので、`node_modules`以下のファイルは除外することにします。

このようなバックアップを行う`gulpfile.js`をすでにプロジェクトに含めてありますので、それを実行します。

```
gulp -m Initial
```

と入力します。`-m`オプション以下の文字は、作成したプロジェクト名に追加する文字列です。



```
問題 出力 デバッグコンソール ターミナル 1: bash ▾ + ■ ▲
```

```
ohtsu@Tiger64:/mnt/c/_myprg/_test/_git/ng8-bootstrap4-initial$ gulp -m Initial
```

```
[19:39:23] Using gulpfile /mnt/c/_myprg/_test/_git/ng8-bootstrap4-initial/gulpfile.js
```

```
[19:39:23] Starting 'default'...
```

```
current_dir ----- ng8-bootstrap4-initial
```

```
time_version ----- 201907071939
```

```
target_path ----- ../ng8-bootstrap4-initial_201907071939_Initial
```

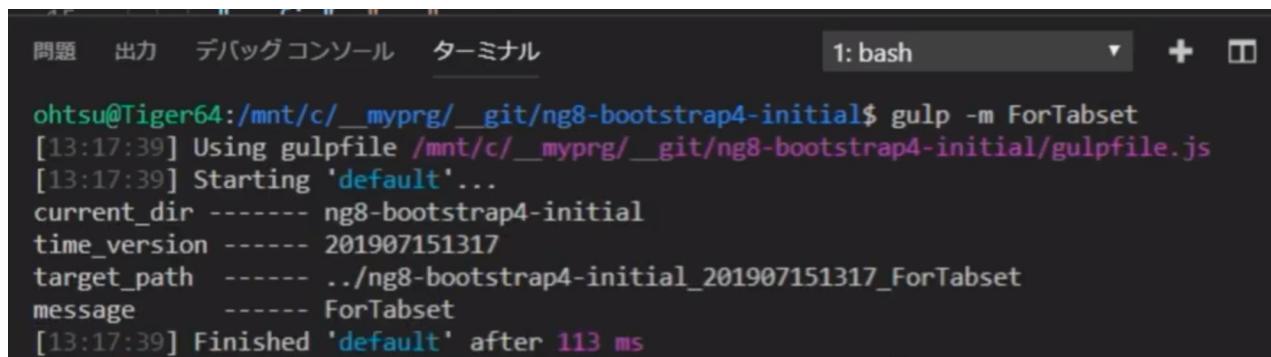
```
message ----- Initial
```

```
[19:39:23] Finished 'default' after 282 ms
```

次に、Tabsetテスト用のプロジェクトも作成しておきます。

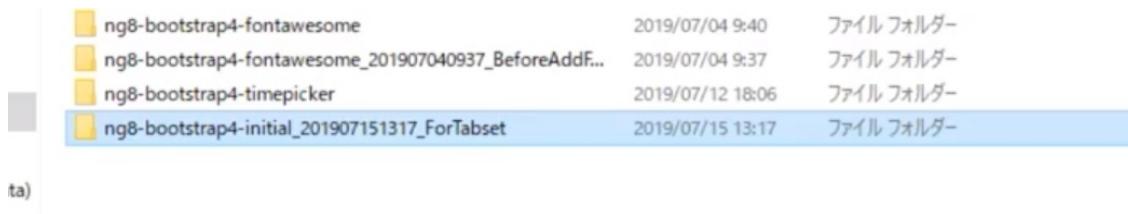
```
gulp -m ForTabset
```

と入力します。

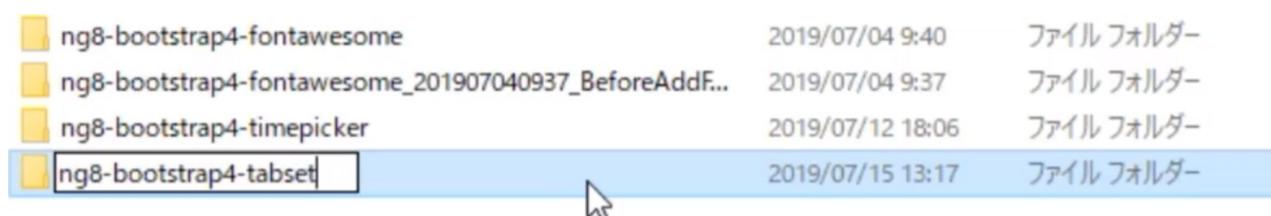


```
ohotsu@Tiger64:/mnt/c/_myprg/_git/ng8-bootstrap4-initial$ gulp -m ForTabset
[13:17:39] Using gulpfile /mnt/c/_myprg/_git/ng8-bootstrap4-initial/gulpfile.js
[13:17:39] Starting 'default'...
current_dir ----- ng8-bootstrap4-initial
time_version ----- 201907151317
target_path ----- ./ng8-bootstrap4-initial_201907151317_ForTabset
message ----- ForTabset
[13:17:39] Finished 'default' after 113 ms
```

次にファイルエクスプローラーを開き、作成したTabsetテスト用のプロジェクトをディレクトリごとコピーし、適当なディレクトリ内にペーストします。



そして、ディレクトリ名をng8-bootstrap4-tabsetに変更しておきます。

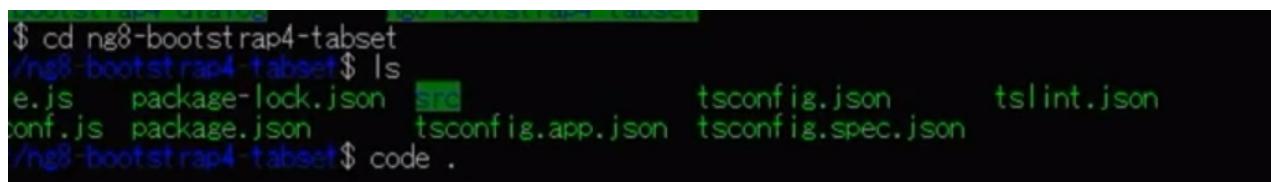


次に、コマンド・プロンプトを開き、このディレクトリに移動します。

そして

```
code .
```

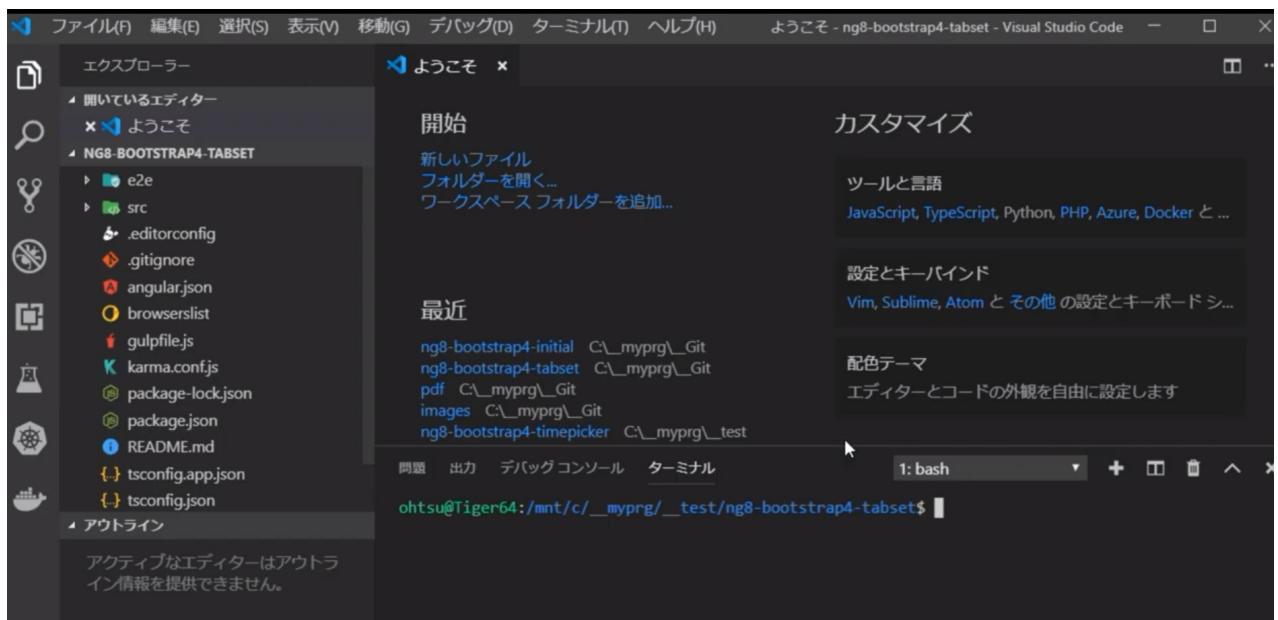
と入力して、VSCodeを開きます。



```
$ cd ng8-bootstrap4-tabset
$ ls
e.js      package-lock.json   tsconfig.json    tslint.json
conf.js   package.json       tsconfig.app.json  tsconfig.spec.json
$ code .
```

VSCodeが起動しましたら、

ctrl+@キーを押し、ターミナル・ウィンドウを開きます。



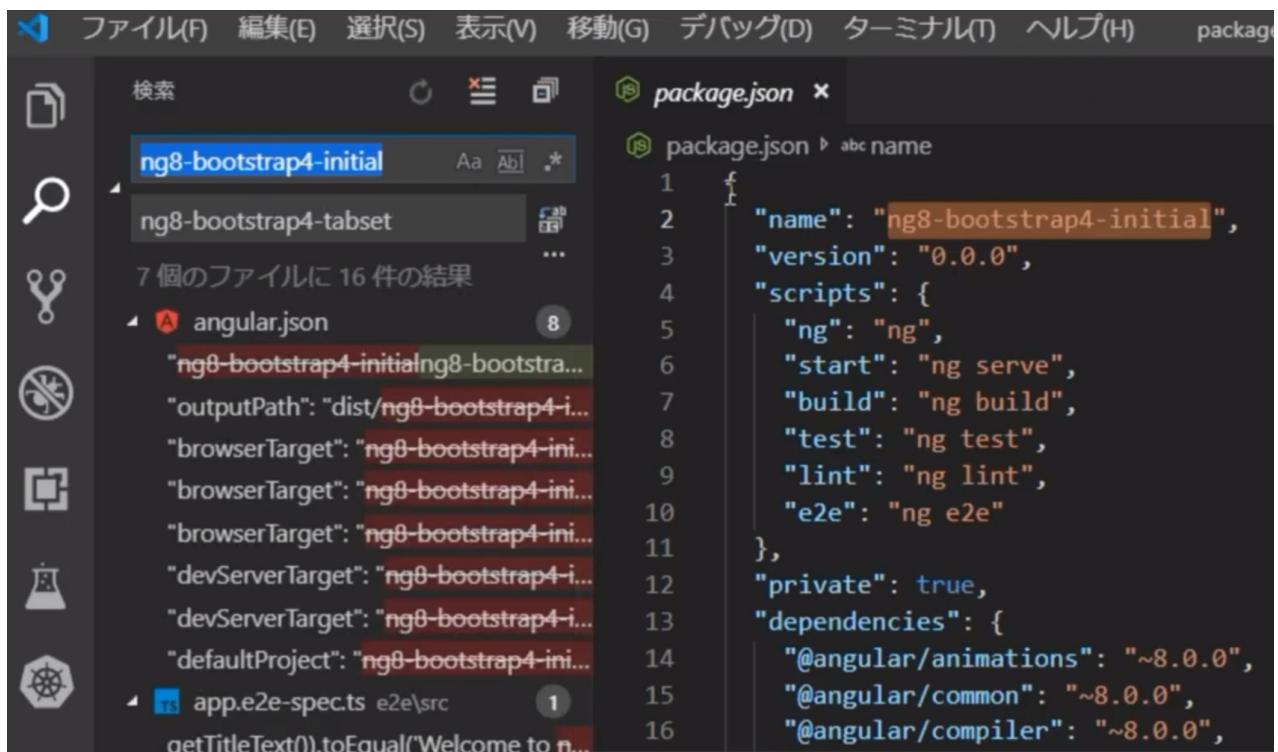
プロジェクト名の変更

次に、プロジェクトの内容を変更していきます。

まず、プロジェクト名を変更します。

`package.json` ファイルを開くと、プロジェクト名が `ng8-bootstrap4-initial` になっているのがわかります。これを `ng8-bootstrap4-tabset` に変更します。

左の検索アイコンをクリックし、プロジェクト全体のソースを対象として置換を行います。



次に、パスカルケース、すなわち-ではなく大文字で区切っているものも存在します

で、Ng8Bootstrap4InitialをNg8Bootstrap4Tabsetに変更しておきます。

```
package.json
1 {  
2   "name": "ng8-bootstrap4-tabset",  
3   "version": "0.0.0",  
4   "scripts": {  
5     "ng": "ng",  
6     "start": "ng serve",  
7     "build": "ng build",  
8     "test": "ng test",  
9     "lint": "ng lint",  
10    "e2e": "ng e2e"  
11  }  
12}
```

次に

```
npm i
```

と入力し、再度ライブラリをインストールします。

```
ohtsu@Tiger64:/mnt/c/_myprg/_test/ng8-bootstrap4-tabset$ npm i  
[.....] \ loadDep:to-regex-range: sill install loadAllDepsIntoIdealTree
```

このプロセスには若干時間をおこします。

npm audit fix の実行

ここでパッケージの脆弱性に関するウォーニングが表示されましたので、パッケージの修正を行います。

```
invalid argument, scandir '/mnt/c/_myprg/_test/ng8-bootstrap4-tabset/node_modules/fsevents/no de_modules'  
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.9 (node_modules/fsevents):  
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.9: wanted {  
"os":"darwin", "arch":"any"} (current: {"os":"linux", "arch":"x64"})  
  
added 1145 packages from 1092 contributors and audited 25405 packages in 361.108s  
found 291 high severity vulnerabilities  
  run `npm audit fix` to fix them, or `npm audit` for details  
ohtsu@Tiger64:/mnt/c/_myprg/_test/ng8-bootstrap4-tabset$
```

```
npm audit fix
```

と入力します。

```
問題 18 出力 デバッグコンソール ターミナル 1: bash + □ ×  
ohtsu@Tiger64:/mnt/c/_myprg/_test/ng8-bootstrap4-tabset$ npm audit fix  
[████.....] \ loadDep:yargs-parser: sill install loadAllDepsIntoIdealTree
```

次に再度

npm i

と入力し、もう一度ライブラリをインストールします。

```
問題 18 出力 デバッグコンソール ターミナル 1: bash +  
ohtsu@Tiger64:/mnt/c/_myprg/_test/ng8-bootstrap4-tabset$ npm i  
[████.....] | diffTrees: sill install generateActionsToTake
```

パッケージの脆弱性に関するウォーニングが消えました。

```
問題 18 出力 デバッグコンソール ターミナル 1: bash + □ ×  
npm WARN bootstrap@4.3.1 requires a peer of popper.js@^1.14.7 but none is installed. You must install peer dependencies yourself.  
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.9 (node_modules/fsevents):  
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.9: wanted {"os": "darwin", "arch": "any"} (current: {"os": "linux", "arch": "x64"})  
audited 25696 packages in 8.794s  
found 0 vulnerabilities  
ohtsu@Tiger64:/mnt/c/_myprg/_test/ng8-bootstrap4-tabset$
```

OKです。

プロジェクト起動のチェック

また、プロジェクトが起動できるかもチェックします。

ng s -o

と入力します。

```
問題 18 出力 デバッグコンソール ターミナル 1: bash +  
ohtsu@Tiger64:/mnt/c/_myprg/_test/ng8-bootstrap4-tabset$ ng s -o  
10% building 3/3 modules 0 active
```

ブラウザが自動的に開き、ドロップダウン及びドロップアップが表示されます。

localhost:4200

アプリ Angular powered B... Google Google 翻訳 Free Online XLIFF E... FAX.PLUS Facebook Google Cloud Platf... Angular Ohtsu (Shuichi Ohts... 一時停止中 0

Welcome to ng8-bootstrap4-tabset!

Toggle dropdown ▾

Action - 1

Another Action

Something else is here

Toggle dropdown ▾

OKです。

Tabsetを追加

次に`ng-bootstrap`のサイトを開き、Tabsetのサンプル・プログラムを参照します。

そのURLは、

<https://ng-bootstrap.github.io/#/home>

です。

https://ng-bootstrap.github.io/#/home

アプリ Angular powered B... Google Google 翻訳 Free Online XLIFF E... FAX.PLUS Facebook Google Cloud Platf... Angular Ohtsu (Shuichi Ohts... 一時停止中 0

Bootstrap widgets

The angular way

Angular widgets built from the ground up using only Bootstrap 4 CSS with APIs designed for the Angular ecosystem.

No dependencies on 3rd party JavaScript.

Demo Installation

Currently at v5.0.0

ここで、Demoボタンをクリックし、左側のメニューからTabsetをクリックします。

そしてGlobal configuration of tabsを表示します。
この設定を取り込むことにします。

The screenshot shows the Angular Material design system documentation. On the left, there's a sidebar with a list of components: buttons, Carousel, Collapse, Datepicker, Dropdown, Modal, Pagination, Popover, Progressbar, Rating, Table, Tabset, Timepicker, Toast, Tooltip, and Typeahead. The main area displays examples for 'Simple' and 'Fancy title' tab sets. The 'Simple' example shows two tabs labeled 'One' and 'Two' with content: 'These tabs are displayed as pills...'. The 'Fancy title' example shows three tabs with titles: 'Raw denim you probably haven't heard of them jean shorts Austin. Nesciunt tofu stumptown aliqua. retro synth master cleanse. Mustache cliche tempor, williamsburg carles vegan helvetica. Reprehenderit butcher retro keffiyeh dreamcatcher synth. Cosby sweater eu banh mi, qui irure terry richardson ex squid. Aliquip placeat salvia cillum iphone. Seitan aliquip quis cardigan american apparel, butcher voluptate nisi qui.' and 'Horizontal' / 'Vertical' buttons. On the right, there are links for 'Tabset', 'Pills', 'Select an active tab by id', 'Prevent tab change', 'Nav justification', 'Nav orientation', and 'Global configuration of tabs'.

まず、右上のCodeボタンをクリックし、そのソースを表示します。

まずHTML文をコピーします。

The screenshot shows the StackBlitz code editor for the 'tabset-config' example. It has tabs for 'tabset-config.html' and 'tabset-config.ts'. The 'tabset-config.html' tab contains the following HTML code:

```
<ngb-tabset>
  <ngb-tab title="One">
    <ng-template ngbTabContent>
      <p>These tabs are displayed as pills...</p>
    </ng-template>
  </ngb-tab>
  <ngb-tab title="Two">
    <ng-template ngbTabContent>
      <p>Because default values have been customized.</p>
    </ng-template>
  </ngb-tab>
</ngb-tabset>
```

Below the code, there's a preview area showing two tabs labeled 'One' and 'Two' with the text 'These tabs are displayed as pills...'.

VSCODEに戻り、src/app内のapp.component.htmlファイルを開きます。

コピーしたHTMLをここに貼り付けます。

```
34
35  <ngb-tabset>
36    <ngb-tab title="One">
37      <ng-template ngbTabContent>
38        <p>These tabs are displayed as pills...</p>
39      </ng-template>
40    </ngb-tab>
41    <ngb-tab title="Two">
42      <ng-template ngbTabContent> I
43        <p>Because default values have been customized.</p>
44      </ng-template>
45    </ngb-tab>
46  </ngb-tabset>
47  <router-outlet></router-outlet>
```

また、プルダウン及びプルアップ関連の部分をコメントアウトします。

```
src > app > app.component.html > div > ngb-tabset
21  </div>
22
23  <div class="col text-right">
24    <div ngbDropdown placement="top-right" class="d-inline-block">
25      <button class="btn btn-outline-primary" id="dropdownBasic2" ngbDropdownToggle>
26        <div ngbDropdownMenu aria-labelledby="dropdownBasic2">
27          <button ngbDropdownItem>Action - 1</button>
28          <button ngbDropdownItem>Another Action</button>
29          <button ngbDropdownItem>Something else is here</button>
30        </div>
31      </div> I
32    </div>
33  </div>
34  -->
35  <ngb-tabset>
36    <ngb-tab title="One">
```

ブラウザに戻り、TypeScriptのファイルを開きます。

まず追加が必要なimport部分をピーします。

Global configuration of tabs

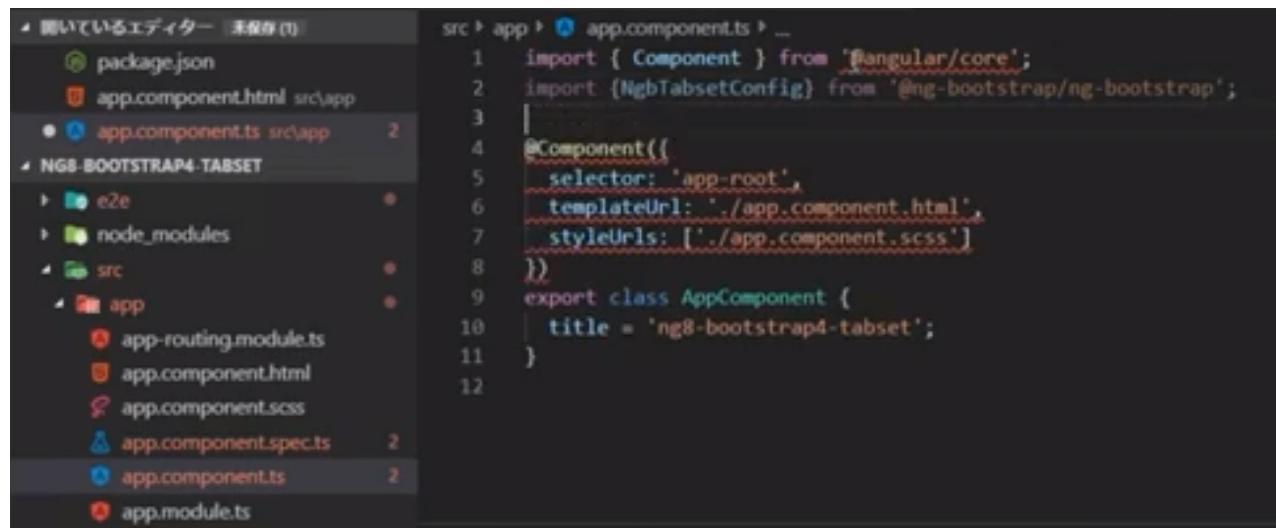
[Code](#)[StackBlitz](#)

```
tabset-config.html tabset-config.ts

import {Component} from '@angular/core';
import {NgbTabsetConfig} from '@ng-bootstrap/ng-bootstrap';

@Component({
  selector: 'ngbd-tabset-config',
  templateUrl: './tabset-config.html',
  providers: [NgbTabsetConfig] // add NgbTabsetConfig to the component provider
})
export class NgbdTabsetConfig {
  constructor(config: NgbTabsetConfig) {
    // customize default values of tabssets used by this component tree
    config.justify = 'center';
    config.type = 'pills';
  }
}
```

VSCodeに戻り、src/app内のapp.component.tsファイルを開きます。
コピーしたimport部分をここに貼り付けます。



ブラウザに戻り、providers部分をコピーします。

Global configuration of tabs

[Code](#)[StackBlitz](#)[tabset-config.html](#)[tabset-config.ts](#)

```
import {Component} from '@angular/core';
import {NgbTabsetConfig} from '@ng-bootstrap/ng-bootstrap';

@Component({
  selector: 'ngbd-tabset-config',
  templateUrl: './tabset-config.html',
  providers: [NgbTabsetConfig] // add NgbTabsetConfig to the component provider
})
```

VSCODEに戻り、*app.component.ts*ファイルを開き、providers部分をここに貼り付けます。

```
4 @Component({
5   selector: 'app-root',
6   templateUrl: './app.component.html',
7   styleUrls: ['./app.component.scss'],
8   providers: [NgbTabsetConfig]
9 })
10 export class AppComponent {
11   title = 'ng8-bootstrap4-tabset';
12 }
13
```

ブラウザに戻り、クラス内の主要部分をコピーします。

```
@Component({
  selector: 'ngbd-tabset-config',
  templateUrl: './tabset-config.html',
  providers: [NgbTabsetConfig] // add NgbTabsetConfig to the component provider
})
export class NgbdTabsetConfig {
  constructor(config: NgbTabsetConfig) {
    // customize default values of tabs used by this component tree.
    config.justify = 'center';
    config.type = 'pills';
  }
}
```

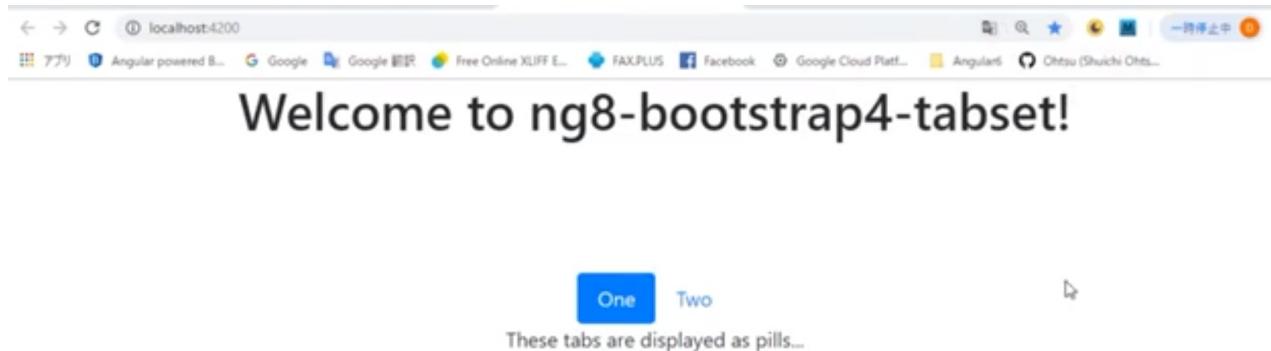
VSCODEに戻り、*app.component.ts*ファイルを開き、ここに貼り付けます。

```
4 @Component({
5   selector: 'app-root',
6   templateUrl: './app.component.html',
7   styleUrls: ['./app.component.scss'],
8   providers: [NgbTabsetConfig]
9 })
10 export class AppComponent {
11   title = 'ng8-bootstrap4-tabset';
12   constructor(config: NgbTabsetConfig) {
13     // customize default values of tabsets used by this component tree
14     config.justify = 'center';
15     config.type = 'pills';
16   }
17 }
18 ]
```

Tabsetの表示

ここで、ブラウザに戻り、<http://localhost:4200>を開きます。

Tabsetが表示されました。



OKです。

Tabのスタイルの変更

この表示ですと、タブが通常のボタンのように表示されていますので、標準のタブのスタイルに変更することにします。

VSCodeに戻り、*app.component.ts*ファイルを開き、

```
config.type = 'tabs'
```

と変更します。

```
10  export class AppComponent {  
11    title = 'ng8-bootstrap4-tabset';  
12    constructor(config: NgbTabsetConfig) {  
13      // customize default values of tabsets used  
14      config.justify = 'center';  
15      config.type = 'tabs';  
16    }  
17  }
```

ブラウザに戻り、表示を確認すると、標準のタブのスタイルになりました。

Welcome to ng8-bootstrap4-tabset!



OKです。

左寄せ

次に、タブが中央に表示されていますので、左寄せで表示することにします。

VSCodeに戻り、*app.component.ts*ファイルを開き、

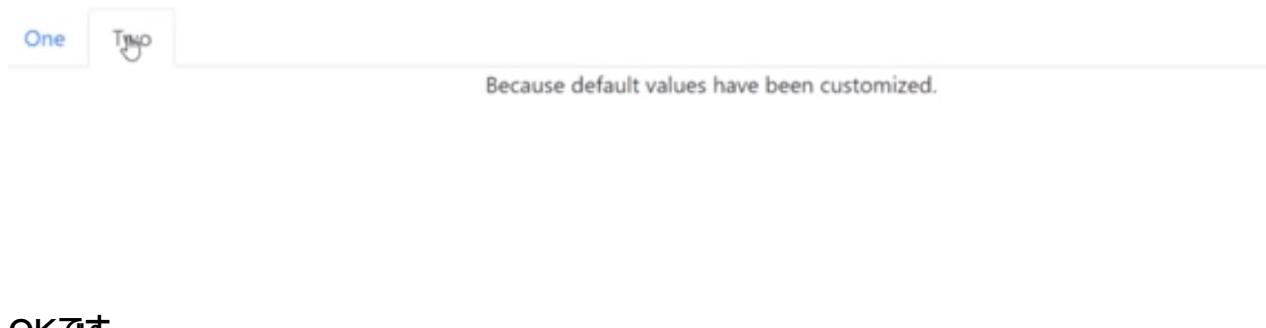
config.justify = 'start'

に変更します。

```
10  export class AppComponent {  
11      title = 'ng8-bootstrap4-tabset';  
12      constructor(config: NgbTabsetConfig) {  
13          // customize default values of tabssets used  
14          config.justify = 'start';  
15          config.type = 'tabs';  
16      }  
17  }
```

ブラウザに戻り、表示を確認すると、タブが左寄せで表示されるようになりました。

Welcome to ng8-bootstrap4-tabset!



OKです。

タブの垂直表示

これまでタブが水平に連なるように表示されていましたが、これを垂直方向に変更にしてみます。

VSCODEに戻り、*app.component.ts*ファイルを開き、

```
config.orientation = 'vertical'
```

を追加します。

```
10  export class AppComponent {
11    title = 'ng8-bootstrap4-tabset';
12    constructor(config: NgbTabsetConfig) {
13      // customize default values of tabssets
14      config.orientation = 'vertical';
15      config.justify = 'start';
16      config.type = 'tabs';
17    }
18  }
```

ブラウザに戻り、表示を確認すると、タイトルが垂直に連なるようになりました。

Welcome to ng8-bootstrap4-tabset!

One

Two

Because default values have been customized.

ボタンへの変更

しかし、このままでいるとタイトルが分かりづらいので、ボタン形式に変更します。

VSCODEに戻り、*app.component.ts*ファイルを開き、

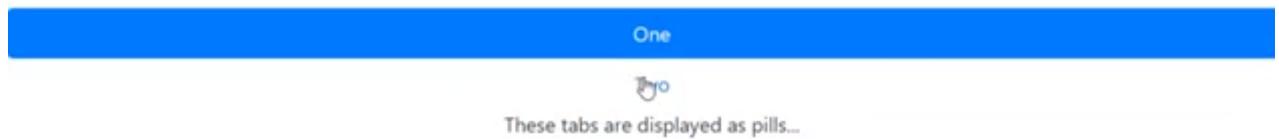
config.type = 'pills'

に変更します。

```
10  export class AppComponent {
11    title = 'ng8-bootstrap4-tabset';
12    constructor(config: NgbTabsetConfig) {
13      // customize default values of tabsets
14      config.orientation = 'vertical';
15      config.justify = 'start';
16      config.type = 'pills';
17    }
18  }
```

ブラウザに戻り、表示を確認すると、ボタン形式に変更になりました。

Welcome to ng8-bootstrap4-tabset!



OKです。

Reference

- "Angular8におけるBootstrap4環境の設定方法(Youtube)",
<https://youtu.be/gFJuSi4IZjM>
- "How to set up Bootstrap 4 environment in Angular 8",
<https://youtu.be/NoSkIVv5o8>
- "Angular8におけるBootstrap4環境の設定方法(Qiita)",
<https://qiita.com/SOhtsu/items/62253bb2309c9a4809fd>
- "Angular 7 Bootstrap 4 UIs: ng-bootstrap and ngx-bootstrap [Part 1]",
<https://www.techiediaries.com/angular-bootstrap-ui/>
- "Angular5, Angular6, Angular7 Custom Library: Step-by-step guide",
<https://www.udemy.com/angular5-custom-library-the-definitive-step-by-step-guide/>
- "Angular5, Angular6, Angular7用 カスタムライブラリの作成: 完全ステップ・バイ・ステップ・ガイド",
[Angular5, Angular6, Angular7用 カスタムライブラリの作成: 完全ステップ・バイ・ステップ・ガイド](#)

<https://www.udemy.com/angular5-1/>