

Angular8におけるBootstrap4のDialogの利用方法

2019.7.8

by *Shuichi Ohtsu*



フルソースコード

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

ビデオ解説(日本語)

<https://youtu.be/ZwuDrmy5QGA>

ビデオ解説(英語)

<https://youtu.be/tbSiZ8Qs9gl>

プロジェクトの作成方法について

ここでは、Angular8においてBootstrap4のDialogを利用する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/gFxsSi4IZjM>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Dialog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。

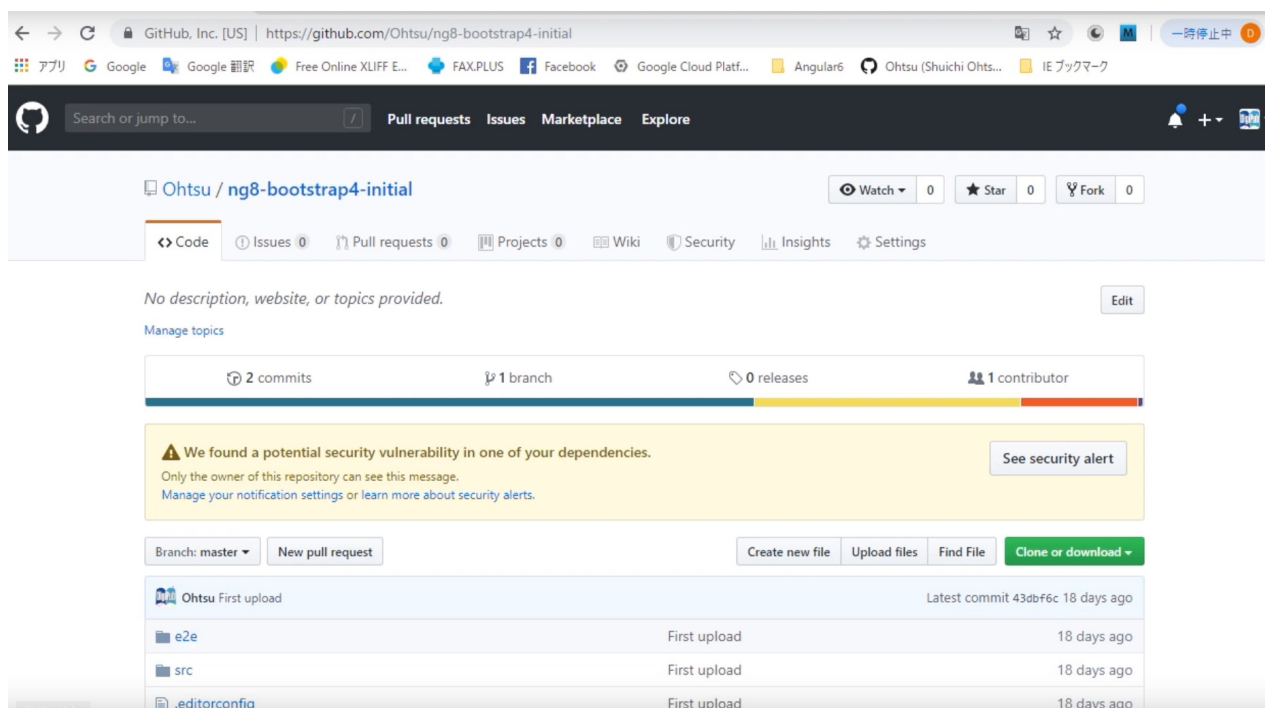
Bootstrap4を組み込んだプロジェクト・ソースのGitHubからのダウンロード

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、

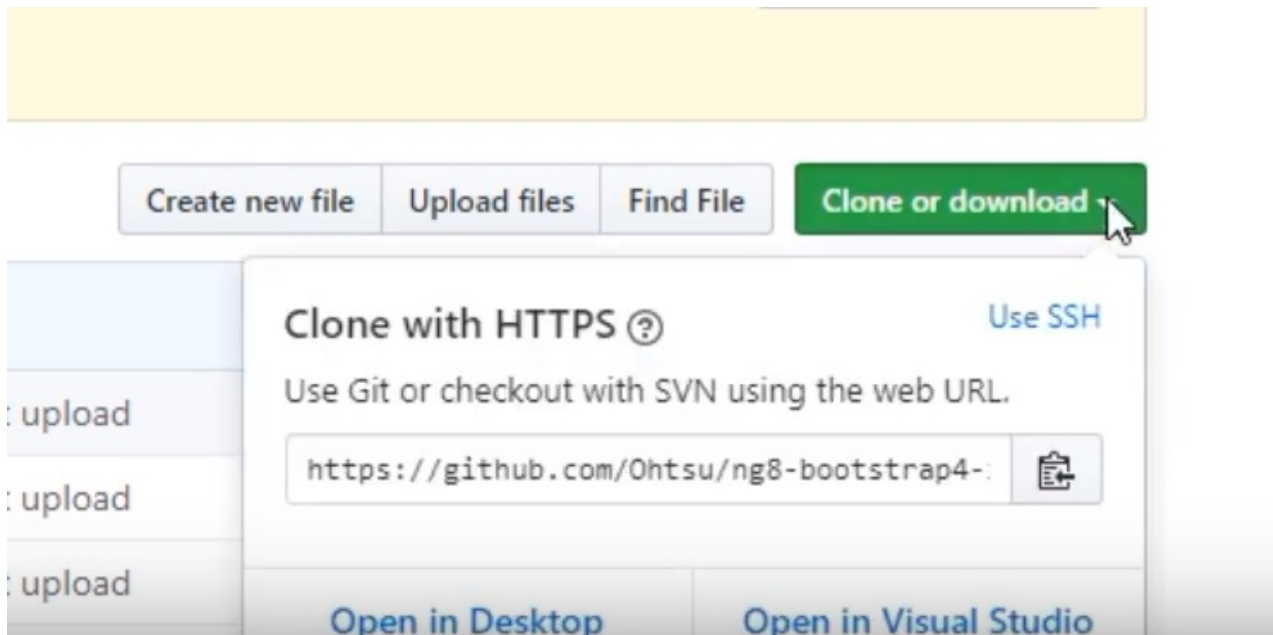
`https://github.com/Ohtsu/ng8-bootstrap4-initial`

です。



ページが表示されましたら、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:/ng8-bootstrap4-initial$ 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。

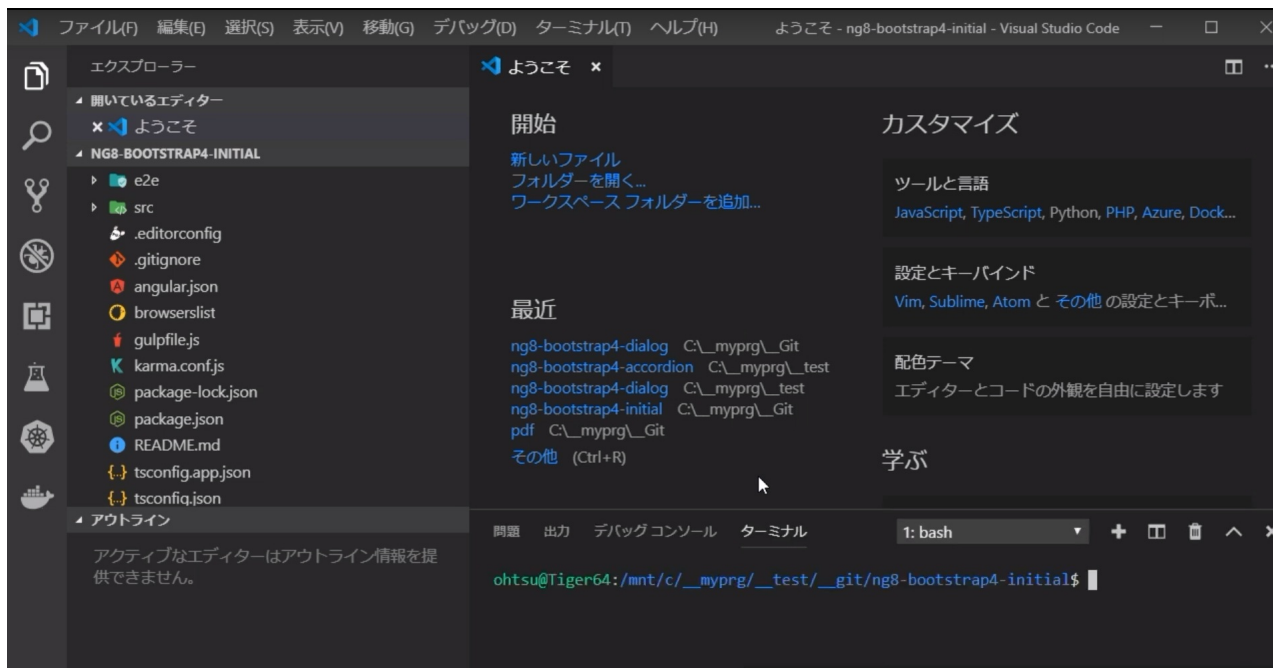
```
code .
```

と入力します。

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

VSCodeが起動したら、

ctrl+@キーを押し、ターミナル・ウィンドウを開きます。

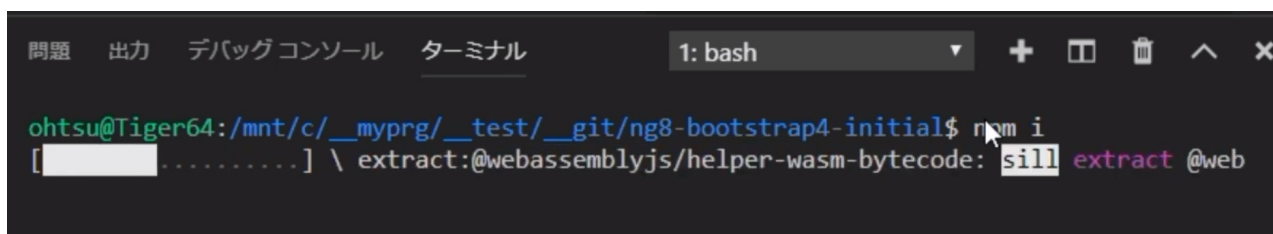


そして、

```
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。



プロジェクトの起動の確認

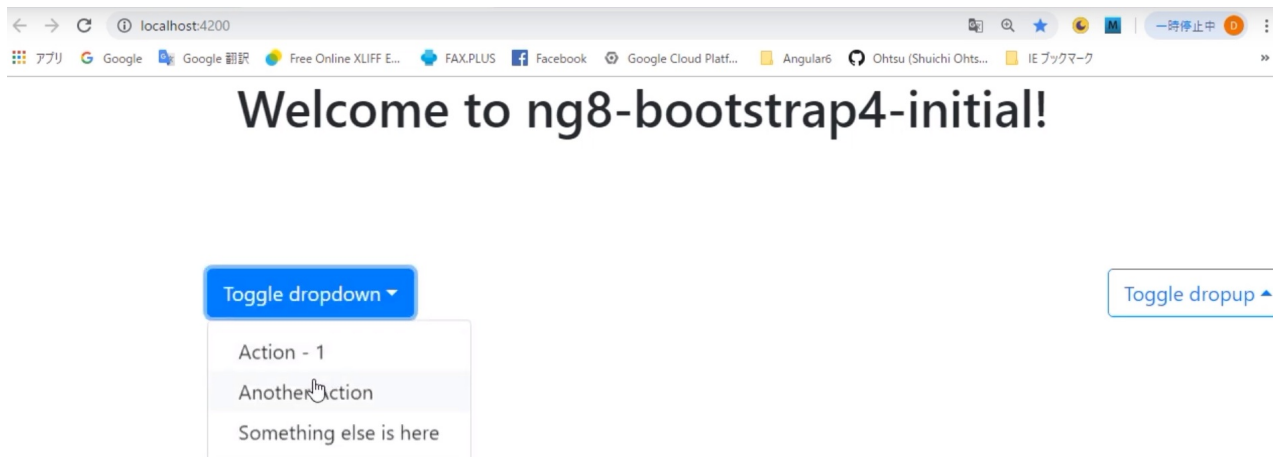
次に、プロジェクトが起動できるかどうかを確認します。

```
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なります。

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
```

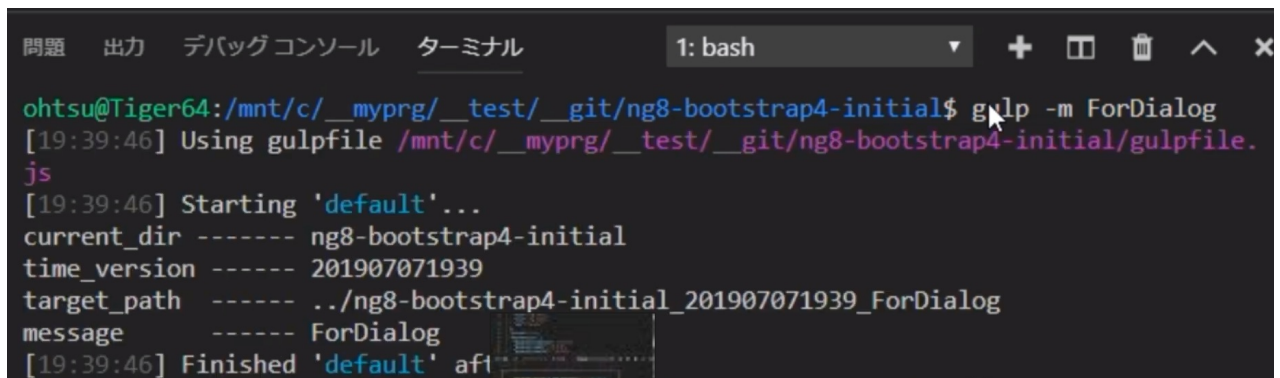
と入力します。`-m` オプション以下の文字は、作成したプロジェクト名に追加する文字列です。

```
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、Dialogテスト用のプロジェクトも作成しておきます。

```
gulp -m ForDialog
```

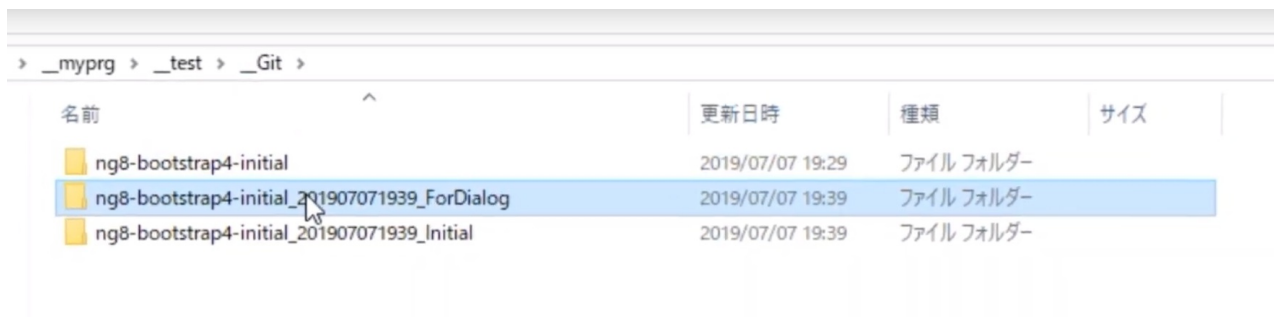
と入力します。



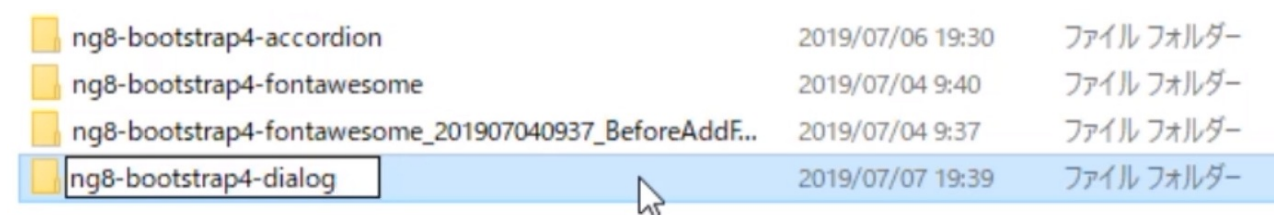
A terminal window titled '1: bash' showing the execution of the command `gulp -m ForDialog`. The output includes the gulpfile path, the task name 'default', and various configuration details like current directory, time version, target path, and message. The task finished successfully.

```
ohtsu@Tiger64:/mnt/c/_myprg/_test/_git/ng8-bootstrap4-initial$ gulp -m ForDialog
[19:39:46] Using gulpfile /mnt/c/_myprg/_test/_git/ng8-bootstrap4-initial/gulpfile.js
[19:39:46] Starting 'default'...
current_dir ----- ng8-bootstrap4-initial
time_version ----- 201907071939
target_path ----- ../ng8-bootstrap4-initial_201907071939_ForDialog
message ----- ForDialog
[19:39:46] Finished 'default' after 0 ms
```

次にファイルエクスプローラーを開き、作成したDialog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-dialog`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

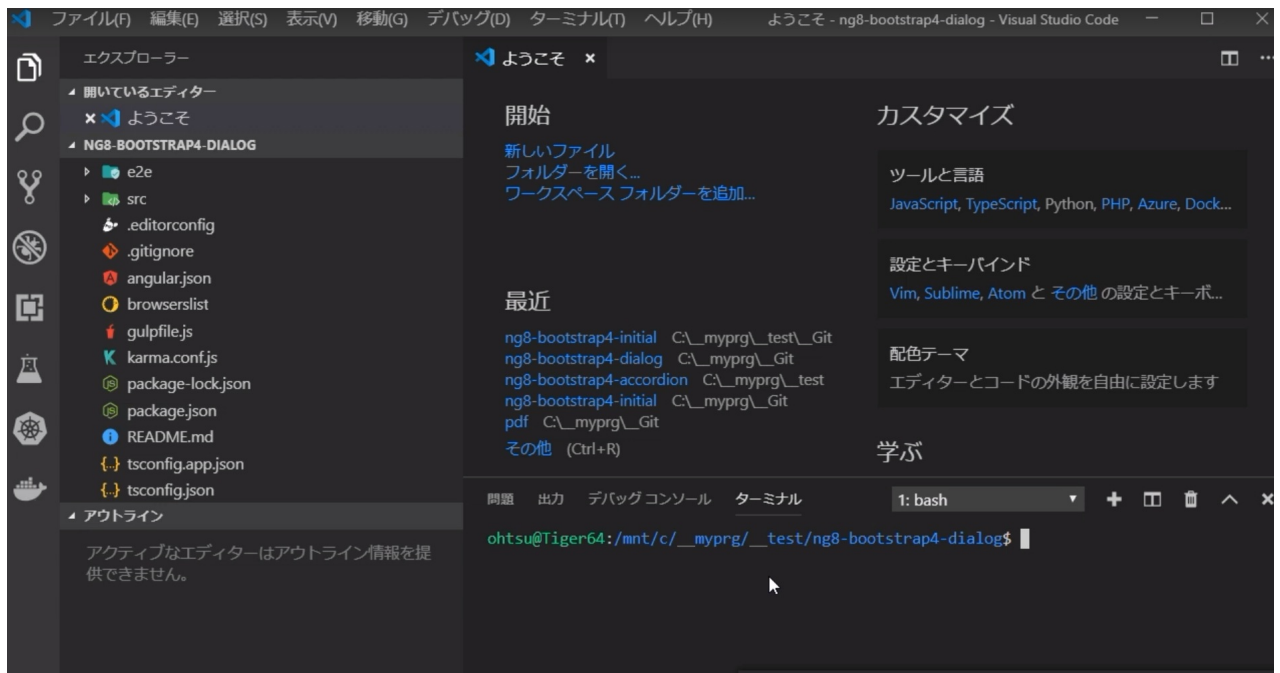
```
code .
```

と入力して、VSCodeを開きます。



VSCodeが起動したら、

ctrl+@キーを押し、ターミナル・ウィンドウを開きます。



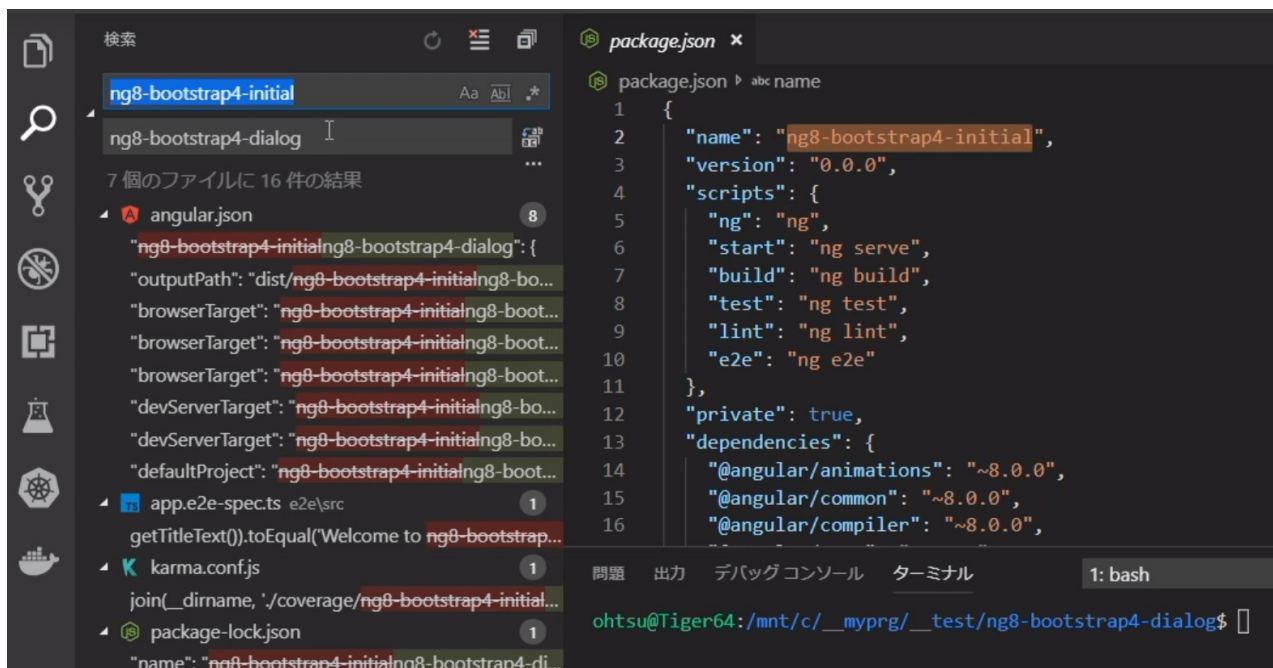
プロジェクト名の変更

次に、プロジェクトの内容を変更していきます。

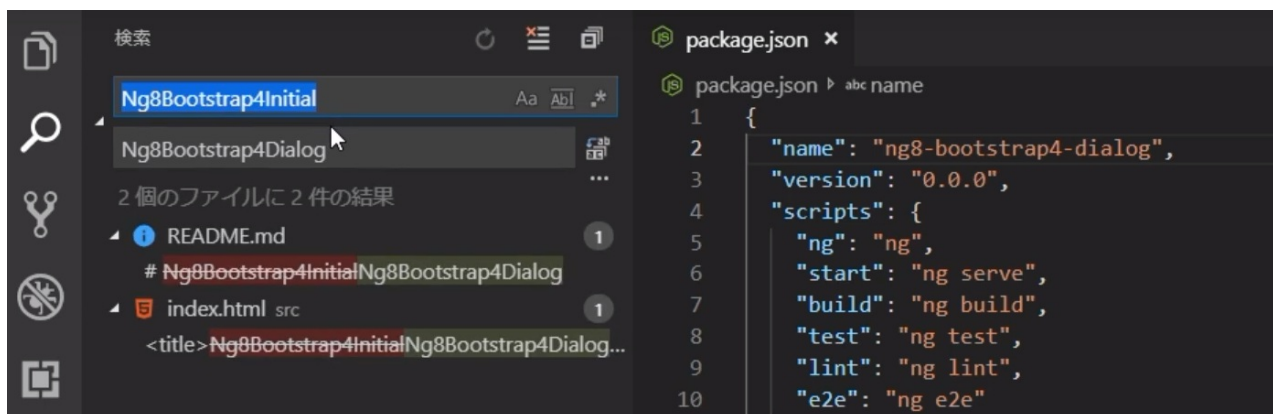
まず、プロジェクト名を変更します。

`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-dialog`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。



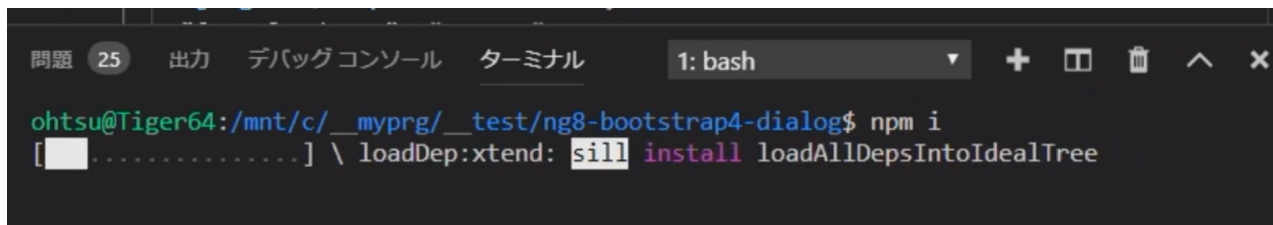
次に、パスカルケース、すなわち-ではなく大文字で区切っているものも存在しますので、Ng8Bootstrap4InitialをNg8Bootstrap4Dialogに変更しておきます。



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。



このプロセスには若干時間を要します。

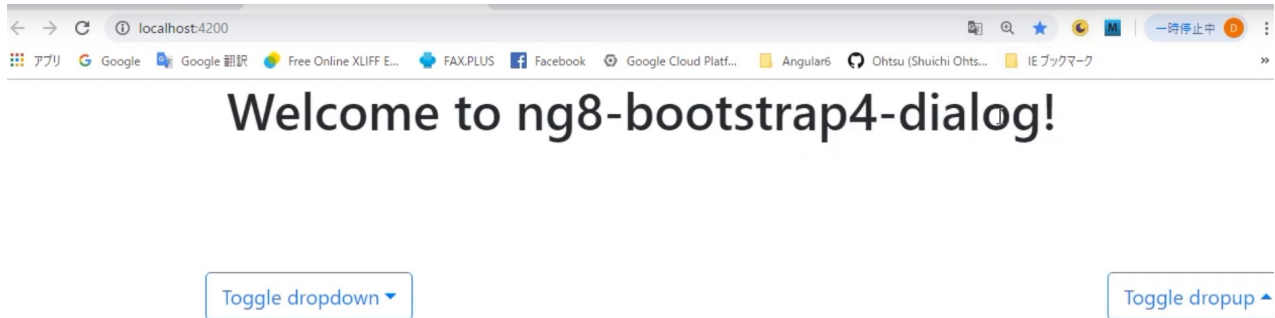
また、プロジェクトが起動できるかもチェックします。

`ng s -o`

と入力します。

```
問題 10 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-dialog$ ng s -o
11% building 15/16 modules 1 active ...bpack/hot sync nonrecursive /^\.\/log$/
```

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。

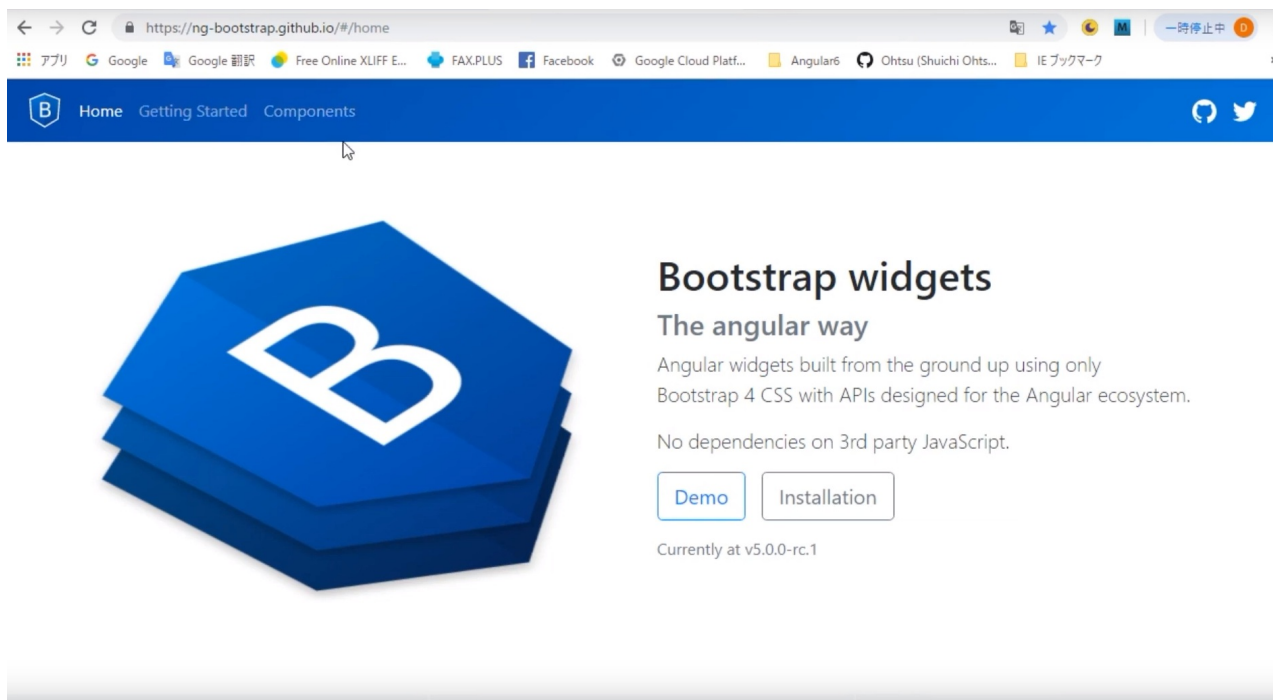
ダイアログを追加

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

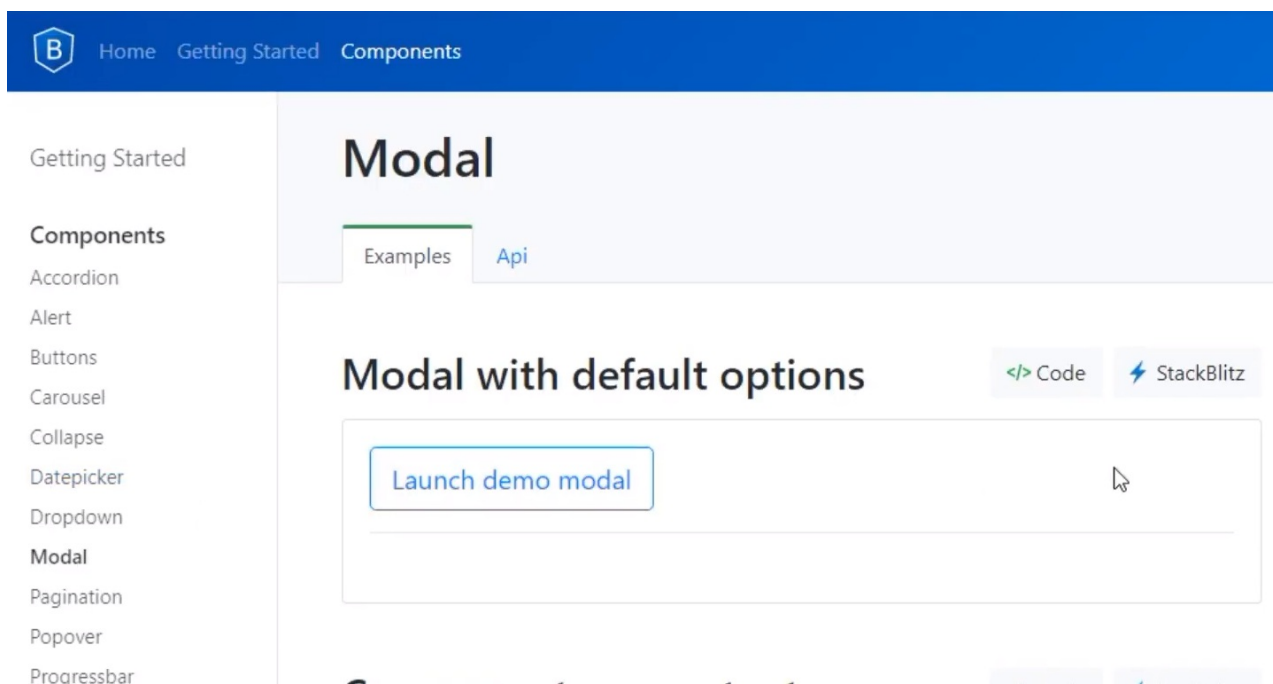
そのURLは、

`https://ng-bootstrap.gitub.io/#/home`

です。



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



Modalのサンプルが表示されていますので、まずこの中のModal with default optionsのサンプルプログラムをAngular8のプロジェクトに取り込むことにします。

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。

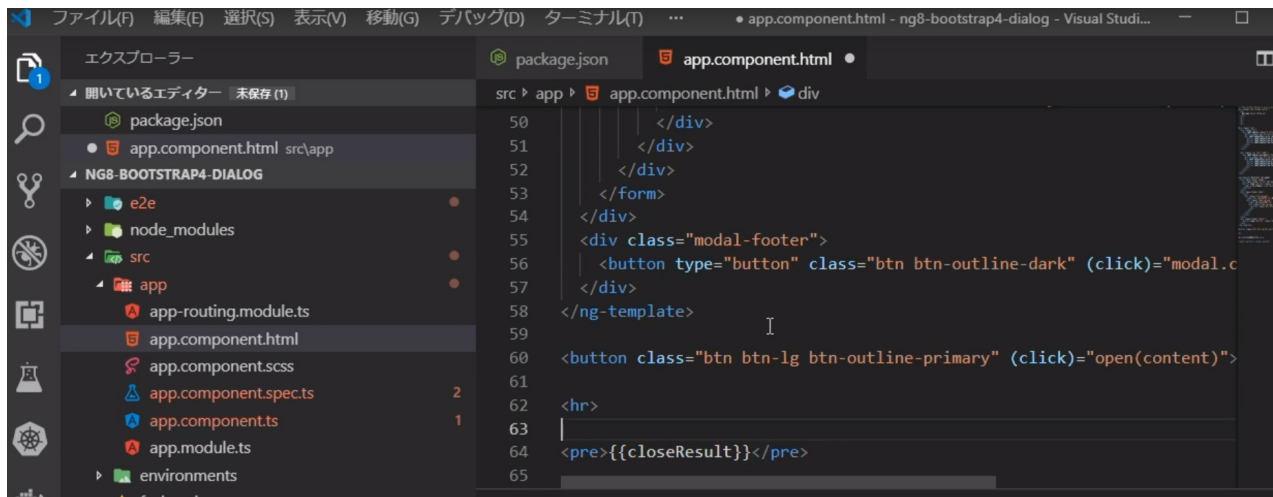
Modal with default options

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

```
modal-basic.html modal-basic.ts

<ng-template #content let-modal>
  <div class="modal-header">
    <h4 class="modal-title" id="modal-basic-title">Profile update</h4>
    <button type="button" class="close" aria-label="Close" (click)="modal.d
      <span aria-hidden="true">&times;</span>
    </button>
  </div>
  <div class="modal-body">
    <form>
      <div class="form-group">
        <label for="dateOfBirth">Date of birth</label>
        <div class="input-group">
          <input id="dateOfBirth" class="form-control" placeholder="yyyy-mm
          <div class="input-group-append">
            <button class="btn btn-outline-secondary calendar" (click)="dp.
```

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



```
package.json app.component.html
src > app > app.component.html > div
50 |
51 |   </div>
52 |   </div>
53 |   </form>
54 | </div>
55 | <div class="modal-footer">
56 |   <button type="button" class="btn btn-outline-dark" (click)="modal.c
57 | </div>
58 | </ng-template>
59 |
60 | <button class="btn btn-lg btn-outline-primary" (click)="open(content)">
61 |
62 | <hr>
63 |
64 | <pre>{{closeResult}}</pre>
65 |
```

コピーしたHTMLをここに貼り付けます。

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。

ブラウザに戻り、TypeScriptのファイルを開きます。

Modal with default options

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

```
modal-basic.html modal-basic.ts

import {Component} from '@angular/core';

import {NgbModal, ModalDismissReasons} from '@ng-bootstrap/ng-bootstrap';

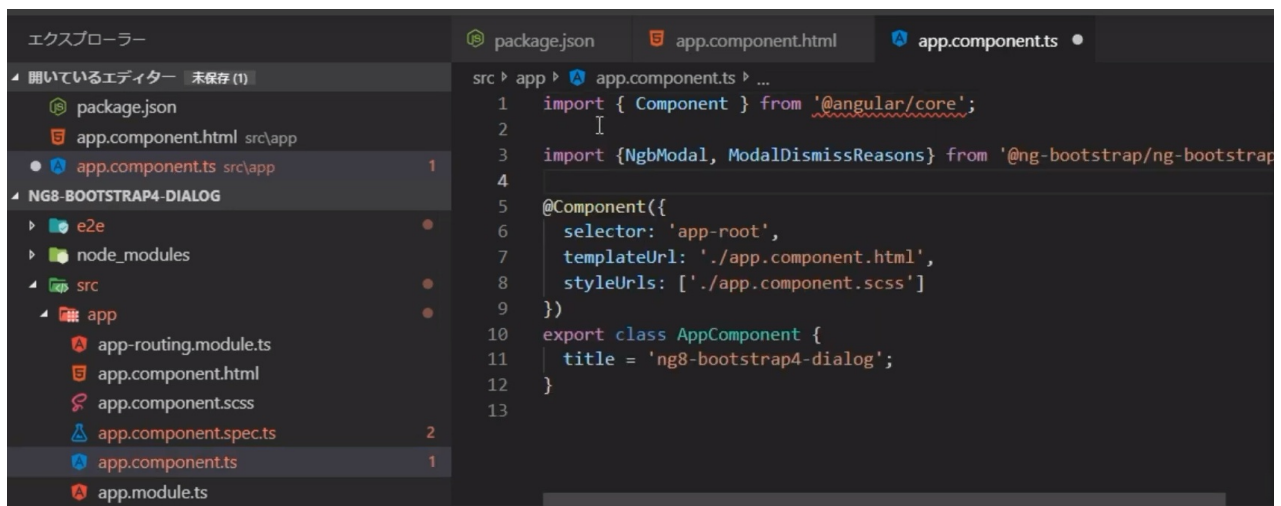
@Component({
  selector: 'ngbd-modal-basic',
  templateUrl: './modal-basic.html'
})
export class NgbdModalBasic {
  closeResult: string;

  constructor(private modalService: NgbModal) {}
}
```

まず追加が必要なimport部分をコピーします。

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

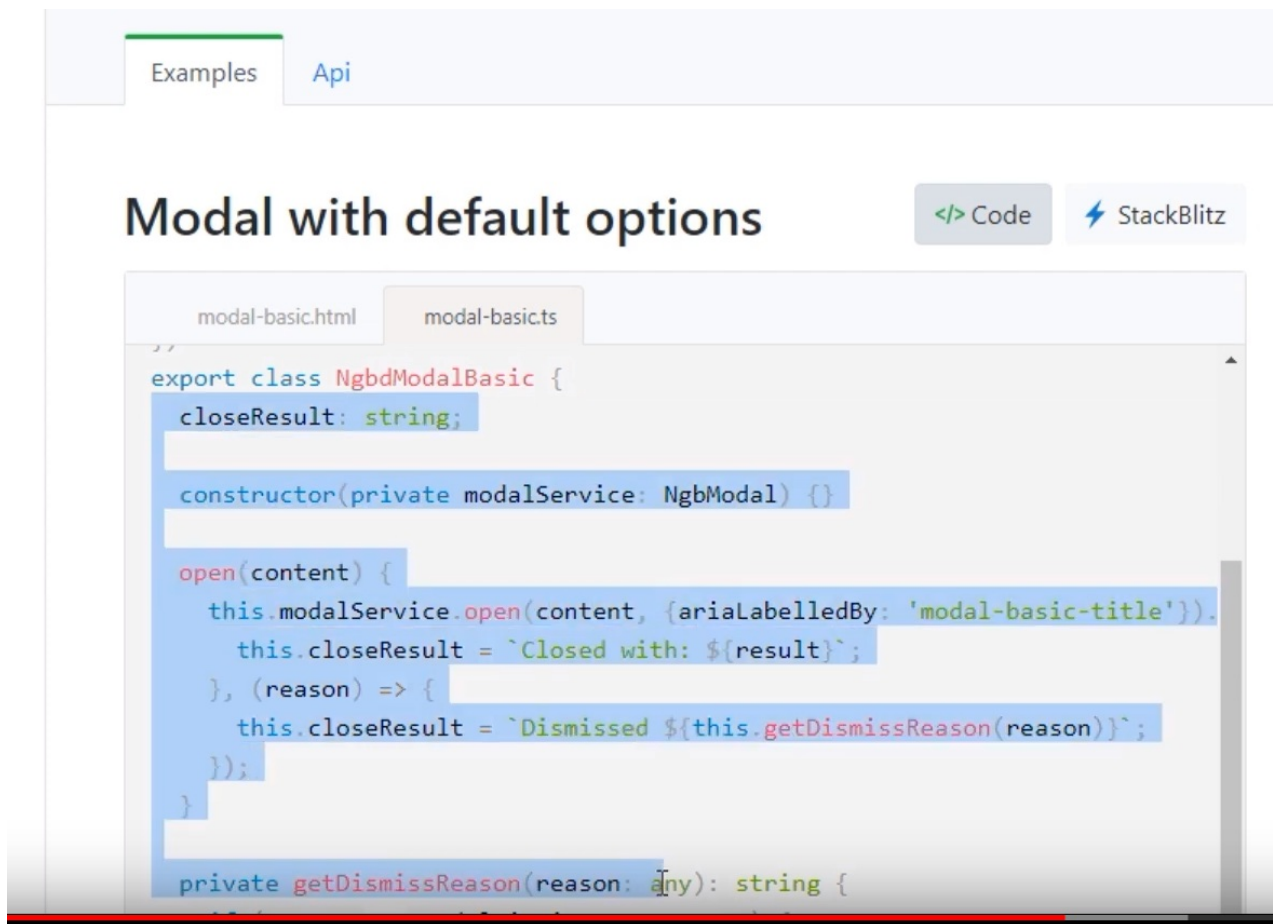
コピーしたimport部分をここに貼り付けます。



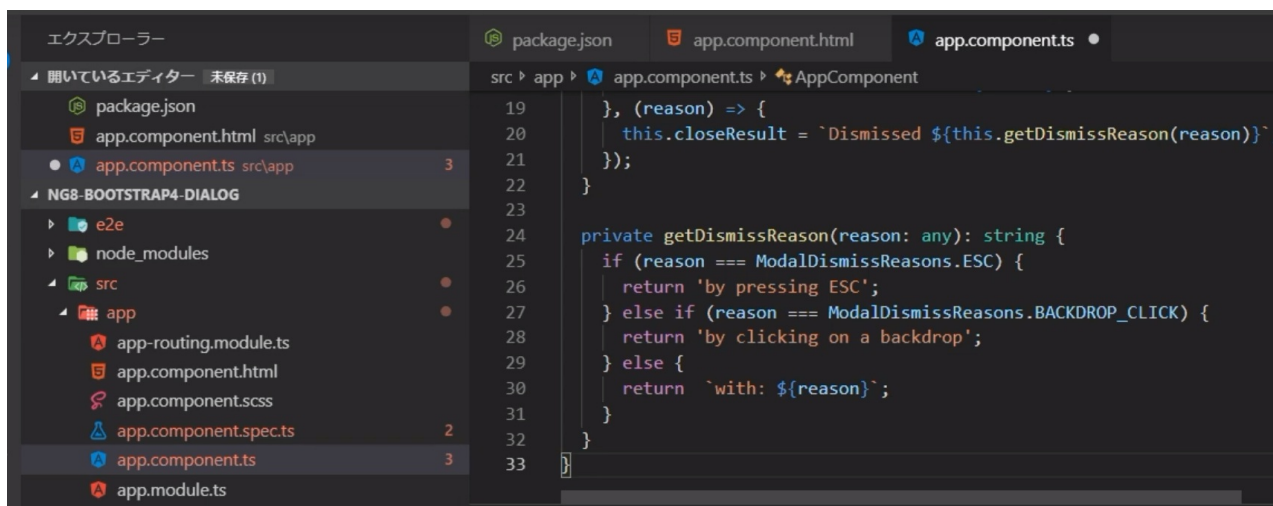
The screenshot shows the VS Code interface. On the left, the Explorer sidebar shows the project structure with 'app.component.ts' selected. The main editor area shows the 'app.component.ts' file with the following code:

```
1 import { Component } from '@angular/core';
2
3 import {NgbModal, ModalDismissReasons} from '@ng-bootstrap/ng-bootstrap';
4
5 @Component({
6   selector: 'app-root',
7   templateUrl: './app.component.html',
8   styleUrls: ['./app.component.scss']
9 })
10 export class AppComponent {
11   title = 'ng8-bootstrap4-dialog';
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。



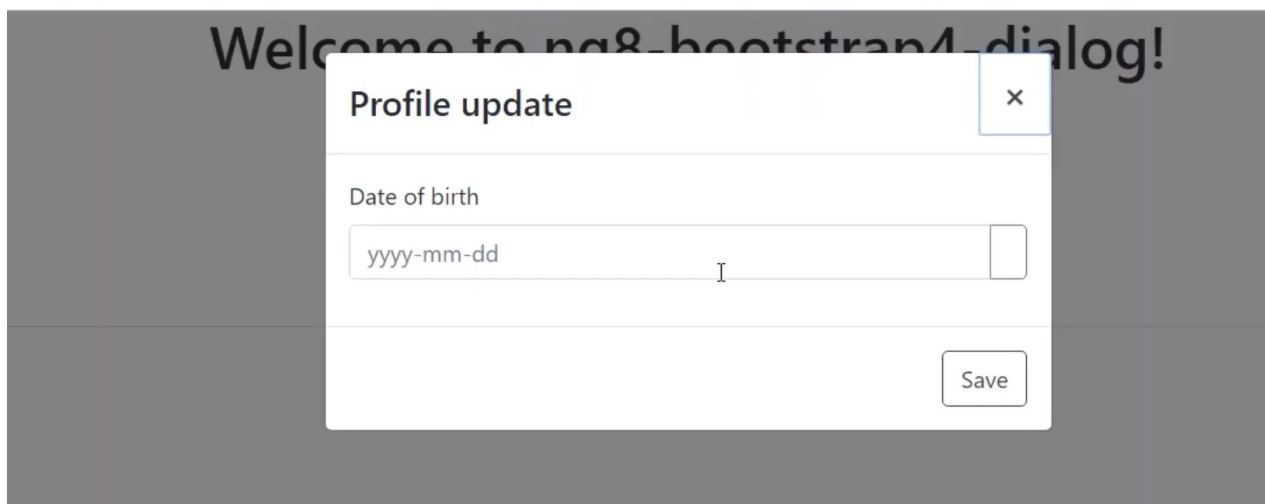
VSCodeに戻り、`app.component.ts`ファイルを開き、クラス内にコピーした部分をここに貼り付けます。



ダイアログの表示

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

`Launch demo modal` ボタンをクリックすると、ダイアログが表示されます。



*Date of birth*の入力欄では、デートピッカーが表示され、*Save*ボタンをクリックすると、ダイアログが閉じられ、画面上にClosed with: Save clickと表示されます。

Welcome to ng8-bootstrap4-dialog!

Launch demo modal

Closed with: Save click

OKです。

日付データの取得

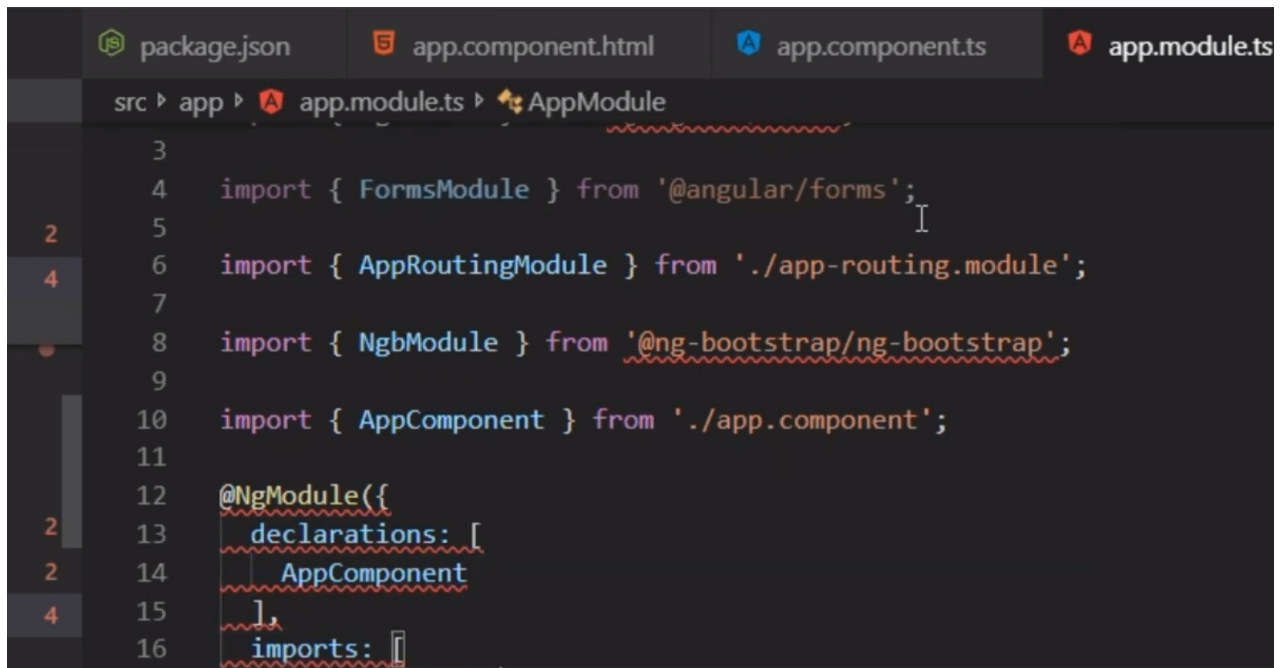
以上がサンプルプログラムと同じ内容のものです。せっかくデートピッカーで日付を指定しても、そのデータが取得できないのであれば意味がありません。

そこで、日付データを取得できるように修正することになります。

そこで、`ngModel`の双方向データバインディングを利用することになります。

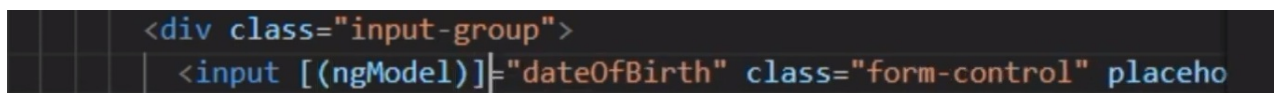
それにはまず、`FormsModule`を新たにインポートする必要があります。

VSCodeに戻り、`app.module.ts`ファイルを開き、`FormsModule`をインポートします。

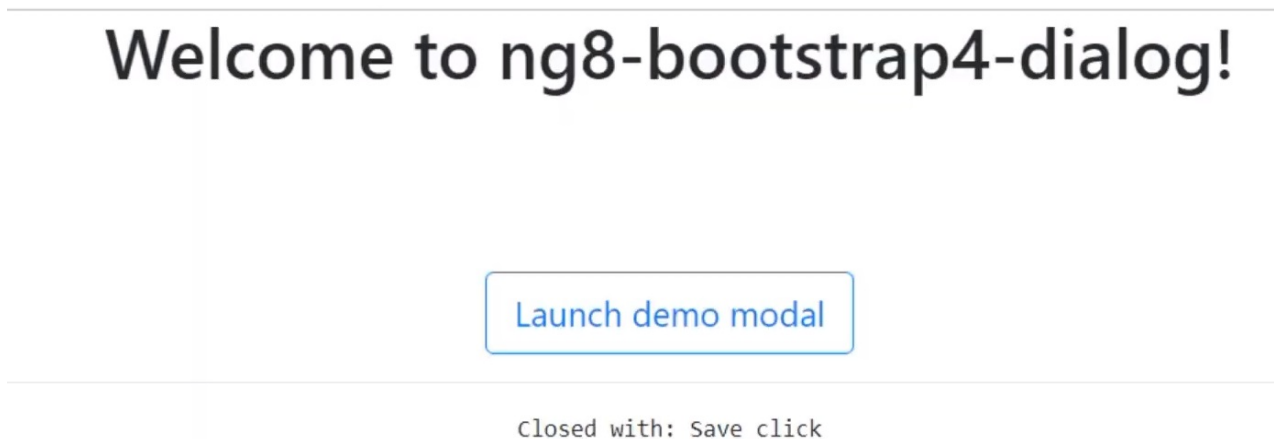


次に、app.module.html ファイルを開き、コメントアウト部分を削除します。

次に、日付を入力している部分に [(ngModel)] を指定して、入力データを取得できるようにします。



ブラウザに戻り、ダイアログを表示し、日付を指定します。



しかし、このままではデータが取得できているかわかりませんので、画面上にそのデータを表示するように修正します。

VSCodeに戻り、app.module.html ファイルを開き、

```
<pre>{{ dateOfBirth | json }}</pre>
```

を挿入します。これは取得したデータをjson形式で表示するという意味になります。

```

39
40 <pre>{{closeResult}}</pre> |
41 <pre>{{dateOfBirth | json}}</pre>

```

ブラウザに戻り、ダイアログを表示し、日付を指定します。指定した日付データが確かに取得できていることがわかります。

Launch demo modal

Closed with: Save click

```

{
  "year": 2019,
  "month": 7,
  "day": 20
}

```

OKです。

最後に、キャンセル・ボタンも追加します。

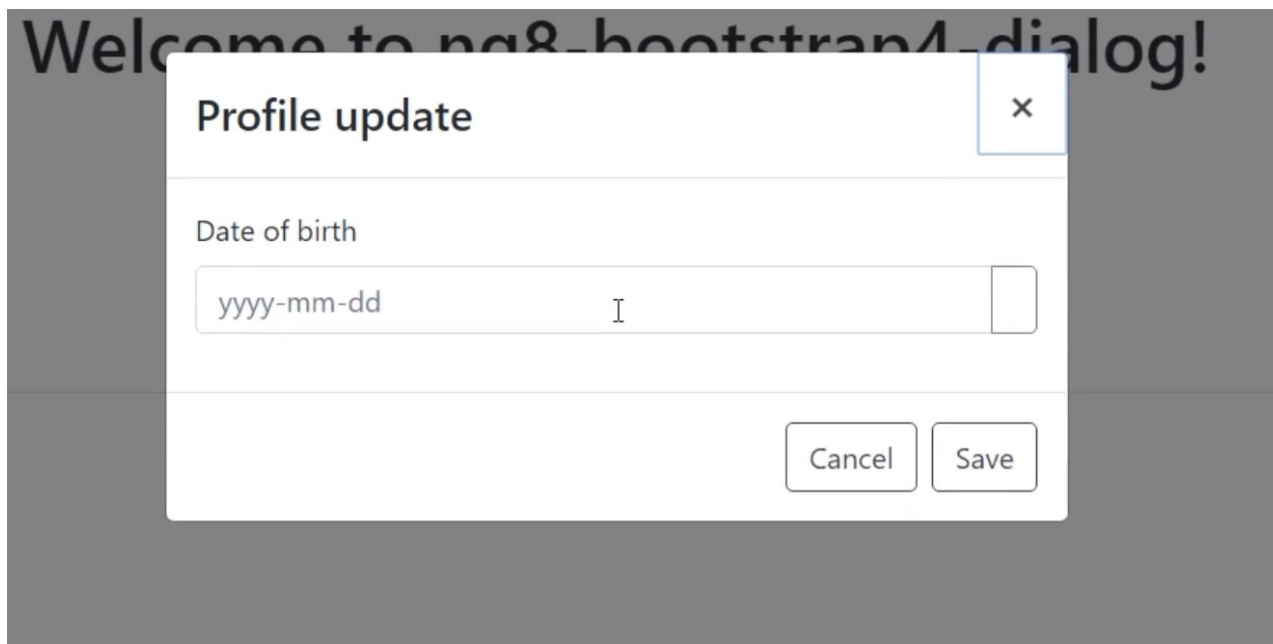
VSCodeに戻り、`modal.footer` 部分を書き換えます。

```

30
31
32 "btn btn-outline-dark" (click)="modal.dismiss('Cancel click')">Cancel
33 "btn btn-outline-dark" Cancel
34

```

ブラウザに戻り、確認します。キャンセル・ボタンが表示され、そのボタンをクリックした場合には、画面上に `Closed with: Cancel click` と表示されます。



Launch demo modal

Dismissed with: **cancel click**

OKです。

Reference

- "Angular8におけるBootstrap4環境の設定方法(Youtube)",
<https://youtu.be/gFxuSi4IZjM>
- "How to set up Bootstrap 4 environment in Angular 8",
<https://youtu.be/NoSkIVvf5o8>
- "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",
<https://www.udemy.com/angular5-1/>