The following examples are based on a scaffolded application named todo-app. The minimal project and app can be scaffolded using the following commands:

ng new todo-project --collection=@bb-cli/schematics

cd todo-project

npm run ng -- generate app --name=todo-app

Scaffold ToDo Widget

In the root directory of your project execute

npm run ng -- generate widget --name=todo-widget

After that structure of your project should be:



Generate Data Modules:

ng generate @bb-cli/schematics:data-module --name TodoData --ramlPath ./raml/api.raml

Response structure:

{

"status": 200,

"TodoItems": [

{

"id": "1",

"value": {

"title": "first todo",

"done": true,

"tags": "important, fun",

"dueDate": "2018-12-05"

}

},

{

"id": "2",

"value": {

"title": "second todo",

"done": false,

"tags": "important, boring",

"dueDate": "2019-06-01"

}

},

{

"id": "3",

"value": {

"title": "third todo",

"done": false,

"tags": "boring",

"dueDate": "2020-02-29"

}

}

]

}

Create -> todo.service.ts

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

import { HttpResponse } from '@angular/common/http';

import { Observable, ReplaySubject } from 'rxjs';

import { map, switchMap } from 'rxjs/operators';

import {TodoDataService} from '../../todo-data/src/todo-data.service';

import {TodoItemsResponse} from '../../todo-data/src/todo-data.interfaces'

export interface TodoItem {

  id: string;

  title: string;

  dueDate: Date;

}

@Injectable()

export class TodoService {

  private readonly id = new ReplaySubject<string>();

  constructor(private readonly data: TodoDataService) {}

  readonly items: Observable<Array<TodoItem>> = this.data.getTodos()

    .pipe(map((response: HttpResponse<TodoItemsResponse>): Array<TodoItem> => {

      return response.body

        ? response.body.TodoItems.map(itemFromData)

        : [];

    }));

}

function itemFromData(record: any): TodoItem {

  return {

    id: record.id,

    title: record.value.title,

    dueDate: new Date(record.value.dueDate),

  };

}

Add -> todo-widget.modue.ts

import { TodoDataModule } from '../../todo-data/src/todo-data.module';

import { TodoService } from './todo.service';

@NgModule({

  declarations: [TodoWidgetComponent],

  imports: [

    ------

    TodoDataModule,

providers: [TodoService],

-------

})

export class TodoWidgetModule { }

Replace: todo-widget.component.ts

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

import { Observable } from 'rxjs';

import { TodoService, TodoItem } from './todo.service';

@Component({

  selector: 'bb-todo-widget',

  template: `

  <ul>

    <li \*ngFor="let todo of todos | async">

      {{ todo.title }}

    </li>

  </ul>

  `,

})

export class TodoWidgetComponent {

  todos: Observable<Array<TodoItem>>;

  constructor(private todoService: TodoService) {

    this.todos = todoService.items;

  }

}

Now environment.ts:

import { TodoDataMocksProvider } from '../../../../libs/todo-data/src/todo-data-mocks.service';

mockProviders: [createMocksInterceptor(),TodoDataMocksProvider,] as Array<Provider>,

validate on <http://localhost:4200/#/>

run – npm start

all good then :

npm run package:apps

and finally import :

bb-import package dist/provisioning-packages/cx6/todo-app.zip --portal-port=8080 --portal-host=localhost --portal-username admin --portal-password admin --portal-context gateway/api --portal-auth-path gateway/api/auth/login --portal-version 6.1