JS

1. Использование форм

Для использования форм необходимо импортировать FormsModule и ReactiveFormsModule в модуль приложения.

Сначала необходимо создать модель программно в коде, а затем привязать элементы HTML к данной модели с помощью директив в шаблоне.

Модель формы — базовая структура данных, которая содержит данные формы. Она создается из специальных классов, определенных в модуле @angular/forms.

Класс **FormControl** — неделимый элемент формы. Обычно он соответствует одному элементу <input>.

**FormGroup** представляет собой часть формы и является набором из нескольких FormControl. Модель FormGroup также используется для представления формы целиком.

Модель **FormArray** обычно является коллекцией полей, способной увеличиваться в размерах.

Создадим форму добавления пользователя с возможностью указать несколько email.

* Создадим новый проект

ng new formExample

* Импортируем модуль форм

app.module.ts

import { BrowserModule } from '@angular/platform-browser';

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

import { FormsModule, ReactiveFormsModule } from '@angular/forms';

import { AppComponent } from './app.component';

@NgModule({

    declarations: [

        AppComponent

    ],

    imports: [

        BrowserModule,

        FormsModule,

        ReactiveFormsModule

    ],

    providers: [],

    bootstrap: [AppComponent]

})

export class AppModule { }

* Создадим объект формы с нужными полями и добавим обработчики событий для кнопок

app.component.ts

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

import { FormGroup, FormControl, Validators, FormArray } from '@angular/forms';

interface MyUser {

   name?: string;

   surname?: string;

   emails?: string[];

}

@Component({

   selector: 'app-root',

   templateUrl: './app.component.html',

   styleUrls: ['./app.component.css']

})

export class AppComponent {

   userForm: FormGroup;

   users: MyUser[] = [];

   constructor() {

      this.userForm = new FormGroup({ *// создание новой формы*

         name: new FormControl(null, [Validators.required]), *// поля формы*

         surname: new FormControl(null, [Validators.required]),

         emails: new FormArray([ *// может быть несколько email, Используем массив*

            new FormControl(null, [Validators.required])

         ])

      });

   }

   onUserFormSubmit() {

      this.users.push(this.userForm.value); *// данные из формы при сабмите добавляем в массив пользователей*

      this.userForm.reset(); *// сброс значений полей формы*

   }

   onAddEmail() {

      (<FormArray>this.userForm.controls['emails']).push(new FormControl(null, [Validators.required]));

*// приводим поле формы к типу FormsArray и добавляем*

   }

   onDelEmail(index) {

      (<FormArray>this.userForm.controls['emails']).removeAt(index); *// приводим поле формы к типу FormsArray и удаляем элемент по интедксу*

   }

}

* Сверстаем форму

app.component.html

<div class="container">

   <div class="row" \*ngFor="let user of users">

      <div class="col-4">

         {{user.name}}

      </div>

      <div class="col-4">

         {{user.surname}}

      </div>

      <div class="col-4">

         <div class="row" \*ngFor="let email of user.emails">

            <div class="col-12">

               {{email}}

            </div>

         </div>

      </div>

   </div>

   <hr>

   <form [formGroup]="userForm" (ngSubmit)="onUserFormSubmit()">

      <div class="row">

         <div class="col-6">

            <div class="form-group">

               <label for="name">Имя</label>

               <input type="text" class="form-control" placeholder="Имя" formControlName="name">

            </div>

         </div>

         <div class="col-6">

            <div class="form-group">

               <label for="surname">Фамилия</label>

               <input type="text" class="form-control" placeholder="Фамилия" formControlName="surname">

            </div>

         </div>

      </div>

      <ng-container formArrayName="emails">

         <div class="row" \*ngFor="let e of userForm.get('emails').controls; let i = index, let last = last">

            <div class="col-6">

               <div class="form-group">

                  <input type="text" class="form-control" placeholder="email" [formControlName]="i">

               </div>

            </div>

            <div class="col-2">

               <div class="form-group">

                  <button class="btn btn-primary" \*ngIf="last" (click)="onAddEmail()">Добавить email</button>

               </div>

            </div>

            <div class="col-2">

               <div class="form-group">

                  <button class="btn btn-danger" \*ngIf="last" (click)="onDelEmail(i)">Удалить email</button>

               </div>

            </div>

         </div>

      </ng-container>

      <button type="submit" class="btn btn-primary" [disabled]="!userForm.valid">Добавить пользователя</button>

   </form>

</div>

https://angular.io/guide/forms

<https://metanit.com/web/angular2/5.5.php>

1. Задание

В проект из lr12 добавить для пользователя поле телефон. Переписать проект с использованием formGroup и осуществить валидацию формы. В поле ввода телефона значение должно вводиться по маске, для этого используйте сторонний модуль text mask, который можно найти в [www.npmjs.com](http://www.npmjs.com). Например <https://www.npmjs.com/package/ngx-mask> или <https://www.npmjs.com/package/angular2-text-mask>