In Reactive Forms Approach,

1. Create formGroupName, like for e.g. [formGroupName]=”form”,
2. Each form controls, should be defined using ‘formControlName’
3. On OnInit lifecycle hook, create the validators for all the form controls, for e.g., this.form = new FormGroup ({

name: new FormControl(‘’, ..optional validators array);

email: new FormControl(‘’, ..optional validators array);

});

1. FormGroup, FormControl, Validators are need to be imported from ‘@angular/forms’
2. Validators can be verified using the existing methods/members like required, minLength, maxLength, pattern, email etc. for e.g.,

name: new FormControl('', [Validators.required, Validators.minLength(4)]),

email: new FormControl('',[Validators.required, Validators.email] )

1. For displaying interactive messages, in the mark up can use <formName>.get(‘formControlName’).hasError(‘<validator>’), for e.g.

<div \*ngIf="form.get('name').hasError('minlength')">

Should be atleast 4 characters!

</div>

<div \*ngIf="form.get('name').hasError('required')">

Name is required!

</div>

1. Note that, ‘minlength’ here is all small letters, unlike in component (which is camelCase), same applies for ‘maxlength’

For multiple custom validations: <https://stackoverflow.com/questions/48322092/angular-4-two-or-more-custom-validation>