**Forms-**

Forms are mainly building block component that uses to capture and process the data which is entered by the user called as forms. There are different types of forms are as follows as-

**Template Driven forms-**

Template driven forms are emphasizes on developing form that are programmed in the components template using directive.

It uses the heavy use of ngModel directive.

What is use of directive here?

The entire form which is converted into angular.

The "ng-submit" directive is used to process the input entered by the user

**Example-**

Contact.component.html

<div class="container">

  <div class="row">

    <form #loginForm ="ngForm"(ngSubmit)="onFormSubmit(loginForm.value)">

        <div class="col-md-12">

            username<input type="text" ngModel name="username">

        </div>

        <div class="col-md-12">

            password<input type="text" ngModel name="password">

        </div>

        <div class="col-md-12">

            <button class="btn btn-primary">Login</button>

        </div>

    </form>

  </div>

</div>

Contact.component.ts

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

@Component({

  selector: 'app-contact',

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

  styleUrls: ['./contact.component.scss']

})

export class ContactComponent implements OnInit {

  constructor() { }

  ngOnInit(): void {

  }

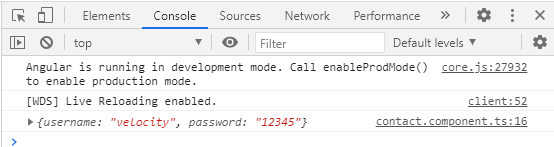
  onFormSubmit(formData: any){

    console.log(formData);

  }

}

Will see the output in the console as



Note- app.module.ts file, need to add FormsModule

**Reactive forms-**

This form will create the explicitly model objects in the type scripts.

Reactive form works with type script component code and template driven form works with HTML.

We construct the form model object explicitly using the FormControl, FormGroup and formArray classes.

Note- FormControl means any HTML attributes it means input, textarea, checkbox and radio buttons.

Note- first thing is to validate the data which is entered by the user at client side then process that data to the server.

When?

If the requirement is complex then go for reactive form otherwise use template driven form for login and registration purpose.

FormControl- it tracks the value and validation status of individual form control.

It means any html attributes that is input, textarea, etc

FormGroup- it tracks the value and status for collection of form control.

Why?

Lets say I have

firstname,

lastname

city

password

confirm password

Address

City

State

Country

Pincode

If you want to group fields you can use FormGroup or will use this as one form

FormArray- it tracks the value and status for array of form control.

Forms directives ngModelGroup- it represents part of form fields together. The ngModelGroup created the nested objects inside ngForm.

Note- for reactive form, you need to declare it in app.module.ts file and import

ReactiveFormsModule

**Example-**

Registration.component.html

<form [formGroup]='angForm' (ngSubmit)="onFormSubmit()">

    username<input type="text" formControlName="username" placeholder="Enter username"><br>

    password<input type="password" formControlName="password" placeholder="Enter password"><br>

    <button type="submit" >Submit</button>

</form>

Registration.component.ts

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

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

import { LoginService } from '../login.service';

@Component({

  selector: 'app-registration',

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

  styleUrls: ['./registration.component.scss']

})

export class RegistrationComponent implements OnInit {

  angForm : any;

  constructor(private fb: FormBuilder) {

  }

  ngOnInit(): void {

    this.createForm();

  }

  createForm(){

      this.angForm=this.fb.group({

        username: new FormControl('', Validators.required),

        password: new FormControl('', Validators.required)

      });

  }

  onFormSubmit(){

    console.log("data using form is>>",this.angForm.value);

  }

}

Output-

Check the console- {username: "velocity", password: "pune"}