**Lesson15 Angular reactive form custom validator**

**Notes:-**

**1-Angular provides several built-in validator functions like required, pattern, minLength, maxLength , but we can make custom validators**

**Example:-**

**Let’s say we only want to allow an email address with pragimtech.com as the domain.**  
Angular reactive form custom validator  
**Any other email domain is invalid.**  


**Notes:-**

required(control: AbstractControl): ValidationErrors | null;

**1-If you look to the definition you take a look at the required built-in function, notice it takes AbstractControl as a parameter. Both FormControl and FormGroup inherits from AbstractControl class**

**2-The method returns null if the control passes validation otherwise Validation Errors object.**

**3-We pass the method as an object with a key and a value. Key is a string and value can be anything**

{ [key: string]: any }

**In the employee-comp.component.ts we apply the following code**

//we apply this function with the parameter we pass the key and value Paris and with the

function emailDomain(control: AbstractControl): { [key:string]: any } | null{

const email :string = control.value;

const domain = email.substring(email.lastIndexOf("@")+1);

//If the email text is null or the domain is 'pragitmatch.com then true

Else activate email Domain Validation

if(email === '' || domain.toLowerCase() === 'pragitmatch.com')

{return null;}

else{return {'emailDomain':true};}}

**2-In the validation Messages we insert the validation message**

validationMessages = {

'fullName': {

'required': 'Full Name is required.',

'minlength': 'Full Name must be greater than 2 characters',

'maxlength': 'Full Name must be less than 10 characters.',},

'email': {

'required': 'Email is required.',

//we use the validation emailDomain with validation value

'emailDomain': 'Email domian should be prgaimtech.com'},

'phone': {

'required': 'Phone is required.'},

'skillName': {

'required': 'Skill Name is required.',},

'experienceInYears': {

'required': 'Experience is required.',},

'proficiency': {

'required': 'Proficiency is required.'}};

**3-we apply the validation on the control email**

ngOnInit() {

// Include FormControls for contactPreference, email & phone

// contactPreference has email as the default value

this.employeeForm = this.fb.group({

fullName: ['', Validators.compose([Validators.required, Validators.minLength(2), Validators.maxLength(10)])],

contactPreference: ['email'],

email: ['', Validators.compose([Validators.required,emailDomain])],

phone: [''],

skills: this.fb.group({

skillName: ['', Validators.required],

experienceInYears: ['', Validators.required],

proficiency: ['', Validators.required]})});}