

# Angular - Formulaire + Observables

SIMPLON  
.CO

# Formulaire : template driven

## Important

Import **FormsModule**

( '@angular/forms' )

dans app.Module

```
<form (ngSubmit)="onSubmit(myForm)" #myForm="ngForm">

  <input type="text"
        name="firstName"
        ngModel
        required>

  <input type="text"
        name="lastName"
        ngModel
        required>

  <button type="submit"
        [disabled]="myForm.invalid">Envie</button>

</form>
```

# Formulaire : reactive form;

```
userForm : FormGroup
```

```
constructor(  
  private FormBuilder: FormBuilder,  
) {  
  this.userForm = new FormGroup({});  
}
```

```
ngOnInit(): void {  
  this.initForm();  
}
```

```
initForm() {  
  this.userForm = new FormGroup({  
    firstName: new FormControl('', Validators.required),  
    lastName: new FormControl('', Validators.required),  
    email: new FormControl('', [Validators.required, Validators.email])  
  });  
}
```

```
onSubmitForm() {  
  const formValue = this.userForm.value;  
  const newUser = new User(  
    formValue['firstName'],  
    formValue['lastName'],  
    formValue['email']  
  )  
}
```

```
<form (ngSubmit)="onSubmitForm()" [formGroup]="userForm">  
  <input formControlName="firstName">  
  <input formControlName="lastName">  
  <input formControlName="email">  
  
  <button type="submit"  
    [disabled]="userForm.invalid">Envoyer</button>  
</form>
```

**Important**

Import

**ReactiveFormsModule**

('@angular/forms')

dans app.Module

# Formulaires Angular

---

Deux méthodes :

- **template driven** : facile à mettre en place, logique déportée dans le template, adapté aux formulaires simple sans règle de gestion complexe
- **reactive form** : plus élaborée, logique déportée dans votre ts, meilleure gestion des validations de formulaire

**MERCI À  
TOUTES ET  
TOUS**

