Angular form

Form là 1 phầần không thể thiếu trong 1 trang web, giúp chúng ta có thể thu thập dữ liệu người dung

FormsModule **được** sử dụng để thực hiện tất cả các lần nhập cần thiết để triển khai biểu mẫu.

**Cú pháp:**import { FormsModule } from '@angular/forms';

**Cách tiếp cận:**

* Tạo một ứng dụng Angular để sử dụng.
* Trong app.component.ts nhập biểu mẫu từ FormsModules.
* Trong app.component.html, tạo biểu mẫu bằng FormsModule.
* Cung cấp ứng dụng góc bằng cách sử dụng ng serve để xem kết quả.

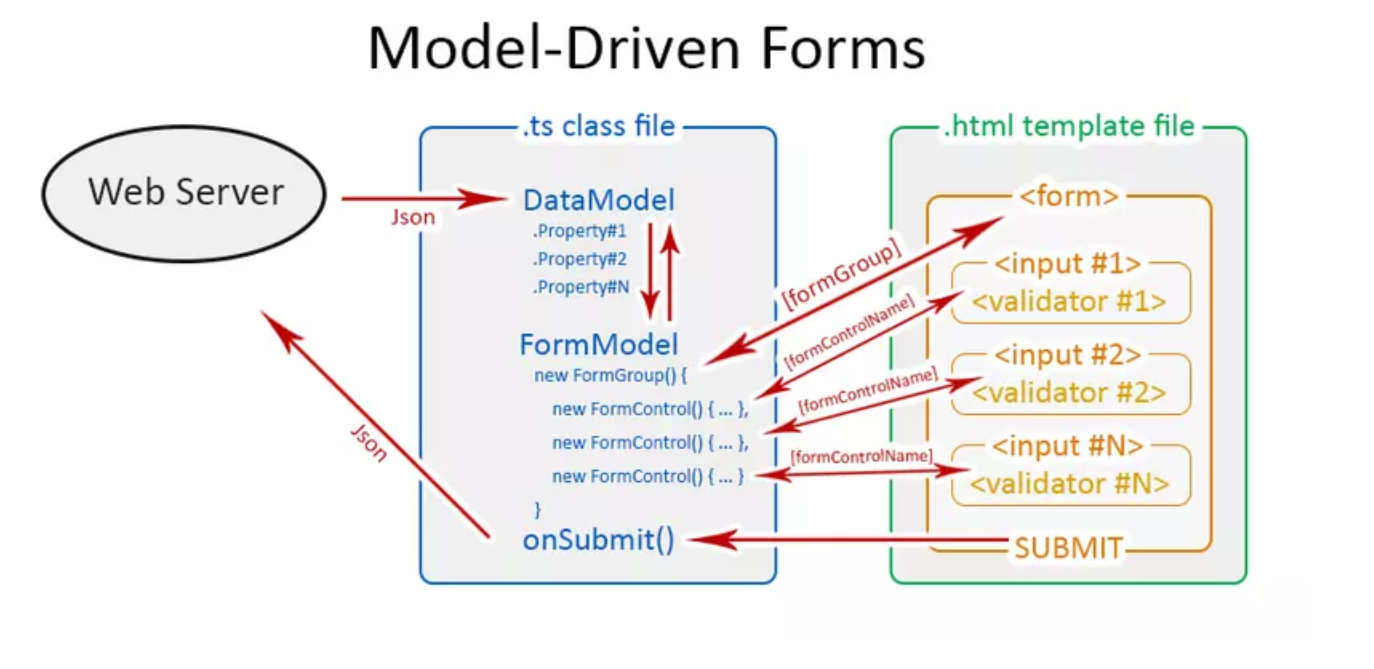
Có 2 loại form là Template driven form và model Dirven form (hay còn gọi là reactive form)

**Model-driven forms**

**Model Driven Forms** hay còn được gọi là **Reactive Forms**

Phương pháp này tránh việc sử dụng các directive như là **ngModel**, **required**... thay vào đó tạo các Object Model ở trong Component, rồi tạo ra form từ chúng.

Cấu trúc cơ bản của 1 **Reactive Form**:



Khi chúng ta nhập các data vào các ô input khi ấn submit nó sẽ chuyển vào các formcontrol tương ứng và sẽ chạy vào onSubmit và gửi data lên web Server

Khi chúng ta cần gọi data thì web Server sẽ gửi data Model rồi sẽ xuống các Form Control tương ứng và sẽ trả về các input tương ứng vói các form control

Như mọi người có thể thấy trong sơ đồ trên, ban đầu chúng ta tạo toàn bộ form control tree ở trong code bằng cách tạo mới các instance như là **FormGroup**, **FormControl**..., sau đó để liên kết các instance này với các element ở bên template thì chúng ta tiếp tục sử dụng các directive **formGroup**, **formControlName**...

Để có thể sử dụng các APIs mà Angular cung cấp cho việc thao tác với **Reactive forms**, chúng ta cần import ReactiveFormsModule từ package [@angular](https://viblo.asia/u/angular)/forms

### Form Builder

Ngoài cách sử dụng phương thức new để khởi tạo 1 instance cho formControl thì Angular form còn cung cấp cho chúng ta 1 cách khác để tạo ra Form model 1 cách nhanh chóng và tiện lợi hơn, đó là **Form Builder**.

Bằng cách inject **FormBuilder** vào class mà chúng ta muốn tạo form ta có thể sử dụng các API như **group**, **array**, **control** để tạo form.

**Validators**

Trong Reactive Form tồn tại 2 loại validator đó là: **Sync Validator** và **Async Validator**:

* **Sync Validator**: Giống như tên gọi của nó thì đây là loại validator đồng bộ, bao gồm các validator trực tiếp với form như là: required, minLength, maxLength...
* **Async Validator**: Validator bất đồng bộ tức là nó sẽ phải chờ 1 tác vụ bất đồng bộ nào đó xử lý xong thì mới tiến hành validate được. Ví dụ như để kiểm tra email trong form đăng ký xem đã tồn tại trong database hay chưa thì chúng ta bắt buộc phải gọi 1 API lên server để kiểm tra, sau khi có kết quả trả về thì lúc này ta mới validate được cho form đăng ký đó.