

Best Practice: ControlValueAccessor

- Può contenere un form indipendente dal principale
- Incapsulato: non accede al padre direttamente
- Basso rischio di bug
- **Riutilizzabile sia con Reactive che Template-driven forms**
- Rimane un componente: possiamo passargli **[input]** e ricevere **(output)**
- Dà soddisfazione personale allo sviluppatore!

Gestione degli errori

- Best practice: il validatore si applica dall'esterno (componente del form principale)
 - Se ci pensate, un input non si valida da solo, è solamente un contenitore
 - Potete usare sia funzioni validatrici (reactive) che direttive validatrici (template-driven)
 - Se il componente deve visualizzare errori internamente, possiamo passarli come `[input]` oppure AL LIMITE iniettare `NgControl` (attenzione)
- Il controllo può auto-validarsi implementando l'interfaccia **Validator**
 - Massima attenzione: guardando il componente padre, non ve ne accorgete