|  |  |
| --- | --- |
| Template  -Uses HTML (template) with two way binding  -easier to use  -automatically tracks form and input element state. Use the state information to determine if the form is valid | Reactive  -Component handles the code  -more flexible  -since no data binding, immutable data model  -Easier to add input elements dynamically and unit test  -bind to the form model not the data model |
| Value Changed (Pristine/Dirty)  Validity (Valid/errors)  Visited (Touched/untouched) | -These states define whether the user has changed a value in an input element  -If one is dirty, then entire form is dirty  -All input elements must be valid  -key of each array element is the name of the  Validation rule  -Form is touched when any of the input element has been touched |
| Form Control  -tracks the value and state of an individual input elements  Form Model (not a data model)  -data structure that represents HTML form  -retains form state, user’s entries , FormControl  -same for Template&Reactive | Form Group  -tracks the value and state of a group of FormControls  -form itself is a FormGroup (what is a form but a group of input elements) |
| Directives | Template-driven (FormsModule)  -ng Form (to access the Form Model), ngModel, ModelGroup  Reactive (ReactiveFormsModule)  -form Group/Control, ControlName, GroupName, ArrayName |
| Template  -Angular generates the FormModel by creating FormGroup and FormControl instances  -HTML validation | Reactive  -Create the FormModel by creating FormGroup and FormControl ourselves in our component class  A screenshot of a computer code  Description automatically generated-Then bind the template to the form model, this means that our form is not directly modifying our data model  -validation in class |
| Template  -requires name attribute to associate the FormControl with FormGroup  - | -template reference variable  #firstNameVar=”ngModel” to access the FormControl instance  -use that TFV to check state properties  [ngClass]=”{‘is-invalid’: firtNameVar.touched}” |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |