**High-level Differences between Template-driven and Reactive Forms**

Below are some of the high-level differences between the two types:

1. Template-driven forms make use of the "FormsModule", while reactive forms are based on "ReactiveFormsModule".
2. Template-driven forms are asynchronous in nature, whereas Reactive forms are mostly synchronous.
3. In a template-driven approach, most of the logic is driven from the template, whereas in reactive-driven approach, the logic resides mainly in the component or typescript code.

**Template Driven Forms Features**

* Easy to use
* Suitable for simple scenarios and fails for complex scenarios
* Similar to AngularJS
* Two way data binding(using [(NgModel)] syntax)
* Minimal component code
* Automatic track of the form and its data(handled by Angular)
* Unit testing is another challenge

**Reactive Forms Features**

* More flexible, but needs a lot of practice
* Handles any complex scenarios
* No data binding is done (immutable data model preferred by most developers)
* More component code and less HTML markup
* Reactive transformations can be made possible such as
  + - * + Handling a event based on a debounce time
        + Handling events when the components are distinct until changed
        + Adding elements dynamically
* Easier unit testing