Briefly explain the two types of Angular Forms.

Angular provides two different approaches to handling user input through forms: reactive and template-driven. Both capture user input events from the view, validate the user input, create a form model and data model to update, and provide a way to track changes.

- Reactive forms provide direct, explicit access to the underlying forms object model. Compared to template-driven forms, they are more robust: they're more scalable, reusable, and testable. If forms are a key part of the application, or there are already other reactive patterns used, reactive forms should be used.
- Template-driven forms rely on directives in the template to create and manipulate the underlying object model. They are useful for adding a simple form to an app, such as an email list signup form. They're straightforward to add to an app, but they don't scale as well as reactive forms. If the form requirements are very basic and their logic can be managed solely in the template, template-driven forms could be a good fit.

What does data-binding in Angular allow you to do?

Data binding gives the components in the application a way to share data (in Angular it can be one- or two-way). Binding allows to listen for events and update values simultaneously between parent and child components.

What is interpolation?

Text interpolation allows to incorporate dynamic string values into the HTML templates and dynamically change what appears in an application view, such as displaying a custom greeting that includes the user's name. Interpolation refers to embedding expressions into marked up text. By default, interpolation uses the double curly braces as delimiters.