1. XAML, which stands for Extensible Application Markup Language, is a markup language primarily used for designing user interfaces in applications developed on the Microsoft platform, particularly in frameworks like WPF (Windows Presentation Foundation), UWP (Universal Windows Platform), and Xamarin.

2. XAML is considered a declarative language because it describes what should be done rather than how it should be done. Instead of writing procedural code to create a user interface element by element, developers use XAML to declare the structure and properties of UI elements in a markup format. This separation of concerns between UI structure and behavior allows for clearer, more maintainable code.

3. **Features of XAML:**
   - **Declarative Syntax:** As mentioned, XAML allows developers to define UI elements and their properties in a markup format, promoting a clear separation between design and logic.
   - **Data Binding:** XAML supports powerful data binding capabilities, enabling UI elements to be dynamically updated based on changes to underlying data sources.
   - **Styling and Templating:** XAML provides mechanisms for defining styles and templates, allowing developers to customize the appearance and behavior of UI controls.
   - **Animation and Transitions:** XAML supports animations and transitions, enabling the creation of rich and interactive user experiences.
   - **Layout Controls:** XAML includes a variety of layout controls such as grids, stacks, and panels, facilitating the creation of complex and responsive UI layouts.
   - **Resource Management:** XAML supports the definition and management of reusable resources such as styles, brushes, and templates, promoting consistency and maintainability in UI design.

4. **Properties and Elements of XAML:**
   - **Properties:** In XAML, properties define the characteristics of UI elements, such as size, color, and visibility. Examples include Width, Height, Background, and Visibility.
   - **Elements:** Elements represent the various UI controls and containers used to build the user interface. Examples include Button, TextBox, Grid, StackPanel, and UserControl.

5. **Importance of XAML in Application Programming:**
   XAML plays a crucial role in modern application development by providing a powerful and expressive language for designing user interfaces. Its declarative nature fosters a clear separation between UI design and application logic, leading to more maintainable and scalable codebases. By leveraging features like data binding, styling, and layout controls, XAML enables developers to create rich, responsive, and visually appealing user experiences across different platforms and form factors. Moreover, its integration with the Microsoft ecosystem, including frameworks like WPF and UWP, makes it a preferred choice for building Windows desktop, mobile, and web applications. Overall, XAML empowers developers to focus on creating compelling user interfaces while promoting code reusability, productivity, and collaboration in application development projects.