**General Information**

Tailwind CSS is a popular utility-first CSS framework that allows developers to rapidly build modern user interfaces. It provides a set of pre-designed CSS classes that can be applied directly to HTML elements to style them without writing custom CSS. Tailwind CSS takes a different approach compared to traditional CSS frameworks like Bootstrap or Foundation.

Here are some key features and concepts of Tailwind CSS:

**Utility-first approach**: Tailwind CSS is built around the concept of utility classes. These classes are small, single-purpose CSS classes that can be combined to create complex styles. Instead of creating custom CSS rules for each element, you apply pre-defined classes directly in your HTML markup.

**Atomic CSS**: Each utility class in Tailwind CSS corresponds to a single CSS property, such as margin, padding, font size, color, etc. By combining these utility classes, you can easily create complex layouts and styles without writing custom CSS code.

**Customizable**: Tailwind CSS is highly customizable. You can modify and extend the default set of utility classes, or even create your own, to match your project's specific requirements. It provides configuration files where you can define color palettes, breakpoints, spacing scales, and more.

**Responsive design**: Tailwind CSS makes it easy to build responsive designs. It provides responsive utility classes that allow you to define different styles for various screen sizes or breakpoints. You can show/hide elements, change their position, adjust margins, and more, based on the device screen width.

**Efficient workflow**: With Tailwind CSS, you can rapidly prototype and iterate on designs. Its utility classes provide a consistent and predictable way of styling elements, eliminating the need to write and maintain a lot of custom CSS code. It also offers features like JIT (Just-in-Time) mode, which generates CSS on-the-fly, reducing the overall file size and improving performance.

Overall, Tailwind CSS aims to provide a highly flexible and efficient way of building modern UIs by leveraging a utility-first approach. It has gained popularity due to its simplicity, customizability, and developer-friendly workflow.

**Note:**

* Tailwind precisely design for mobile. So in order to build application with tailwind, it is recommended to design it for mobile first.  
  For ex:   
  bg-white px-10 md:text-center  
  This style gonna apply this mobo upawards.  
  Here, bg-white px-10 will apply by default (means, if we start designing our app with mobo screen, so this style will apply there)  
  md:text-center this style will apply for medium screen devices. So this will not apply for mobo devices.