Auto Layout

Expressive constraint based layout engine
Constraints are a mathematical relationship between two layout elements
Relate one attribute of an element to an attribute of a different element (\( y = m \cdot x + b \))
Many constraints make up a typical layout
Each element must have a defined x position, y position, width, and height

Constraints

Encapsulates a relationship between two elements
Installed on the closest common parent of the related elements
Relates attributes with equality, greater than or equal to, or less than or equal to
Priority is used to determine precedence of constraints that would otherwise be ambiguous (0 being least important, 1000 being required)
Attributes

- Left, Right, Top, Bottom, Leading, Trailing, Center X, Center Y, First Baseline, Last Baseline, Width, Height
- Margin based variants for all but baseline, width, and height
- “Not an Attribute” attribute used in some cases for the x in y = m*x + b when setting a value to a constant (i.e., height of a view = 100)

Intrinsic Content Size

- Default “ideal” size of views that have an intrinsic size based on content
- Typically this will be any view that displays content or renders a control provided by the SDK (UILabel, UIImageView, UIButton, UISlider, UISwitch, etc.)
- Two sets of priorities used to control when a view will not be equal to its intrinsic content size: content hugging and content compression priorities
- Specifying explicit height or width will override

Priorities

- Every constraint has a priority value between 0 and 1000
- Constraints with a priority of 1000 are considered required
- Content hugging and content compression priorities for views with intrinsic content size
Documentation

“Auto Layout Guide”
“NSLayoutConstraint”