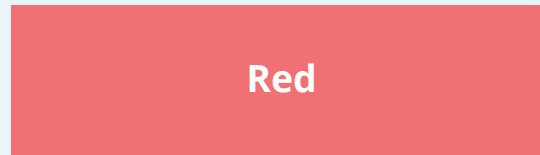


Cheatsheet - Auto-Layout

Anatomy of a constraint



Blue



Red

`RedView.Left = 1.0 * BlueView.Right + 8.0`
View Attribute Relation Multiplier View Attribute Constant



Functions

```
view.translatesAutoresizingMask  
    IntoConstraints = false
```

```
NSLayoutConstraint(item: Any,  
    attribute: NSLayoutConstraint.Attribute,  
    relatedBy: NSLayoutConstraint.Relation,  
     toItem: Any?,  
    attribute: NSLayoutConstraint.Attribute,  
    multiplier: CGFloat,  
    constant: CGFloat)
```

```
view.addConstraints  
    (constraints: [NSLayoutConstraint])
```


Types of constraints


 Top
 Bottom

 Left
 Right


Leading
Trailing


 Width

 Height

 Center X

 Center Y

 Ag First Baseline

 Last Baseline

 Top Margin
 Bottom Margin

 Left Margin
 Right Margin

Leading Margin
Trailing Margin

Checklist

Remember that you need to set

- Width
- Height
- X-Position
- Y-Position

in order for Auto-Layout to be able to resolve the constraints properly.



© The Swift Library

This document is part of the Auto-Layout module at <http://theswiftlibrary.com/course/auto-layout>.