



# UI Guide

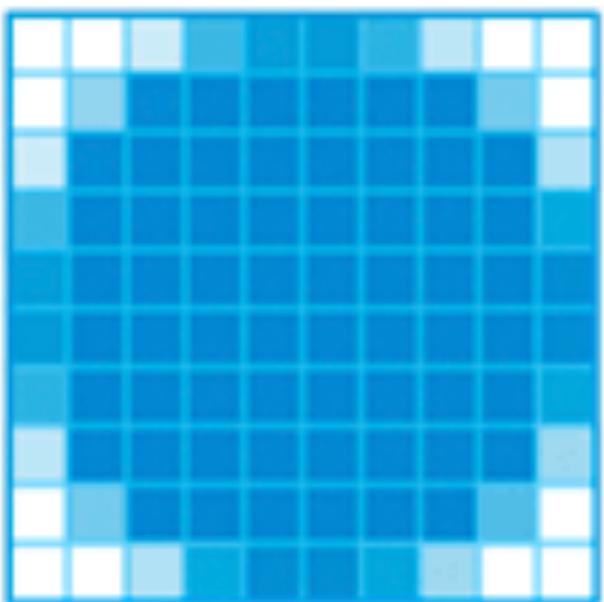
**Giftbot**

# iPhone X Resolution

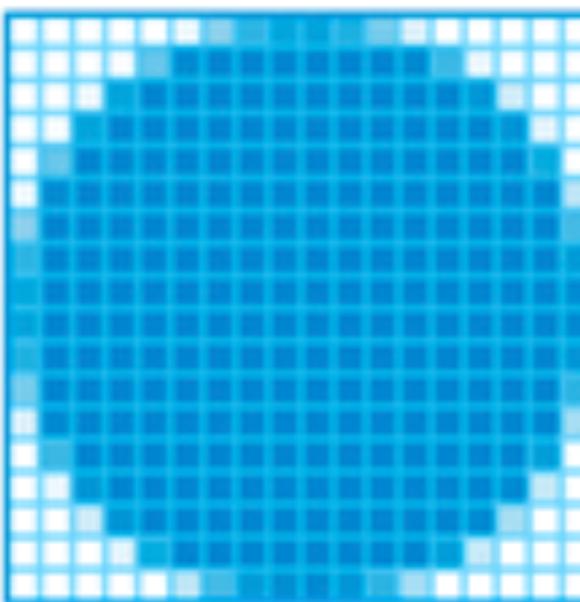


Device	Retina	Portrait (px)	Landscape (px)
iPhone XS Max	🕒	1242 x 2688	2688 x 1242
iPhone XR	🕒	828 x 1792	1792 x 828
iPhone X, XS	🕒	1125 x 2436	2436 x 1125
iPhone 6+, 6S+, 7+, 8+	🕒	1080 x 1920	1920 x 1080
iPhone 6, 6S, 7, 8	🕒	750 x 1334	1334 x 750
iPhone 5, 6SE 5, 5S, 5C, 6SE	🕒	640 x 1136	1136 x 640
iPhone 4 4, 4S	🕒	640 x 960	960 x 640
iPhone 1st, 2nd & 3rd Generation	🕒	320 x 480	480 x 320
iPad Air / Retina iPad 1st & 2nd Generation / 3rd & 4th	🕒	1536 x 2048	2048 x 1536
iPad Pro	🕒	2048 x 2732	2732 x 2048
iPad Mini 2nd, 3rd & 4th Generation	🕒	1536 x 2048	2048 x 1536
iPad Mini, 1st & 2nd Generation	🕒	768 x 1024	1024 x 768

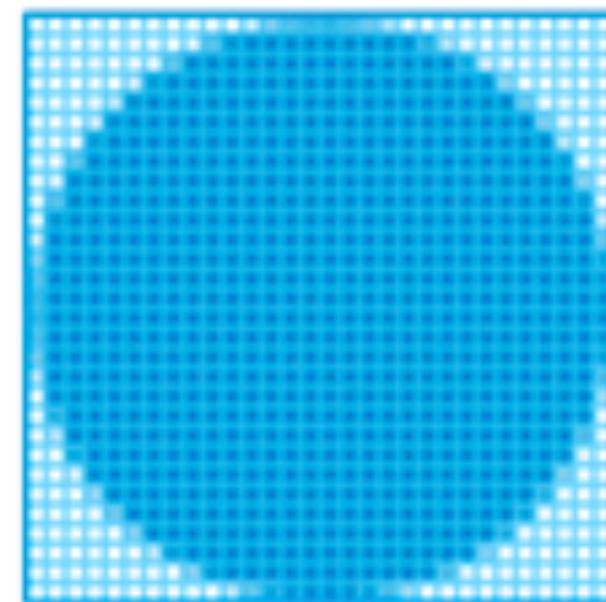
# High Resolution



10 physical pixels

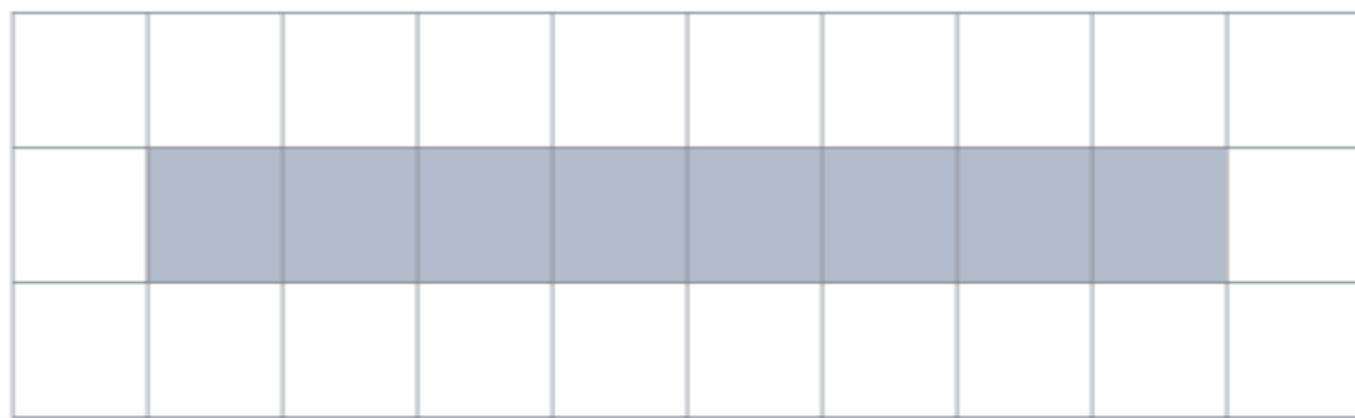


20 physical pixels

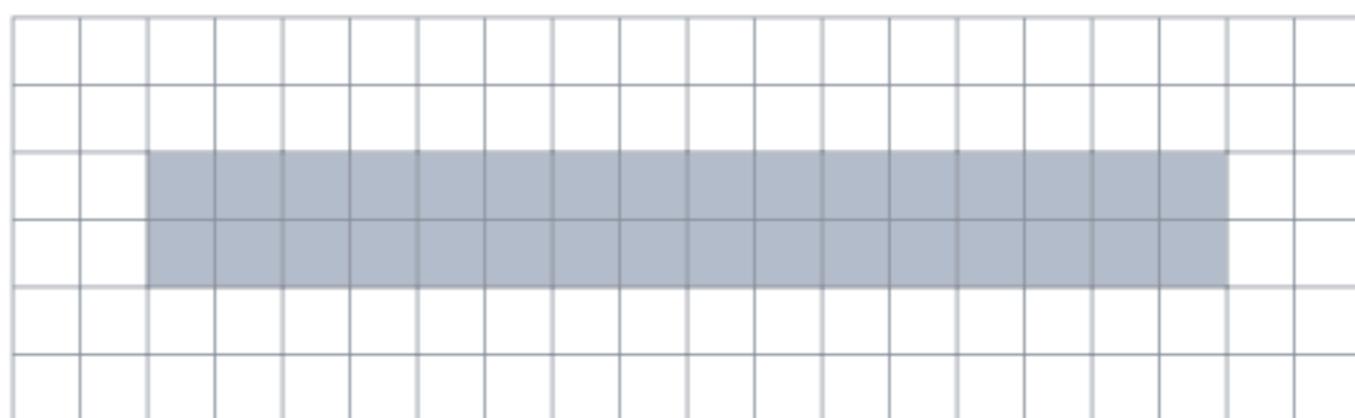


40 physical pixels

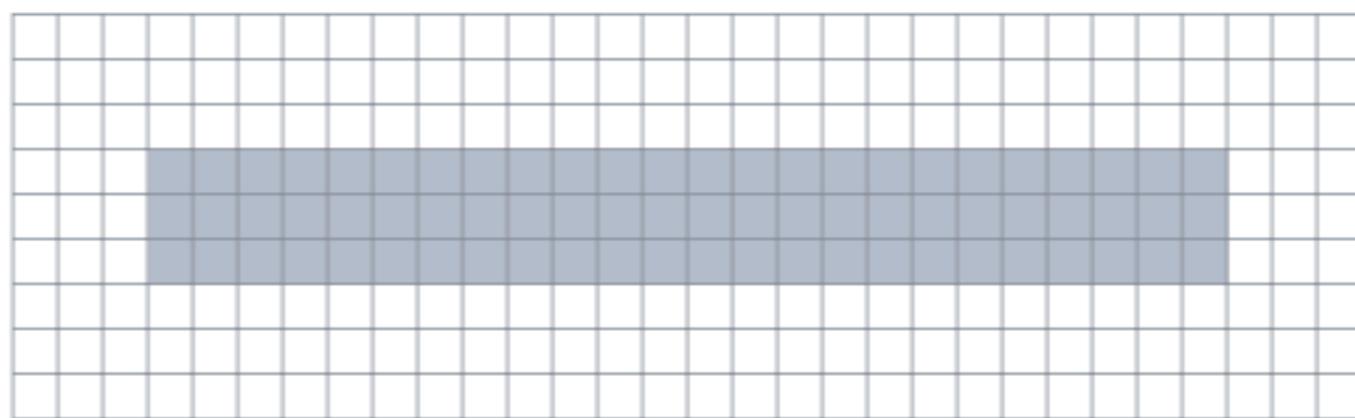
Original iPhone



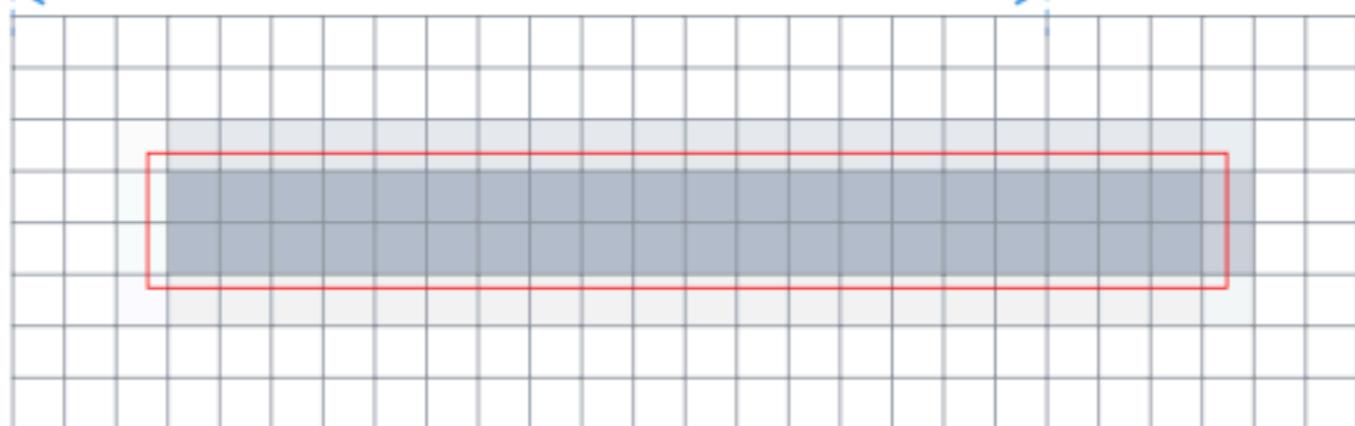
iPhone 5



Hypothetical  
Perfect 3x Display



iPhone 6 Plus



Original iPhone  
320 x 480 points

iPhone 4  
320 x 480 points

iPhone 5  
320 x 568 points

iPhone 8  
375 x 667 points

iPhone 8 Plus  
414 x 736 points

iPhone X  
375 x 812 points

@1x



320 x 480 pixels

@2x



640 x 960 pixels

@2x



640 x 1136 pixels

@2x



750 x 1334 pixels

@3x



Downsampled (87%)  
1080 x 1920 pixels

1242 x 2208 pixels

@3x



1125 x 2436 pixels

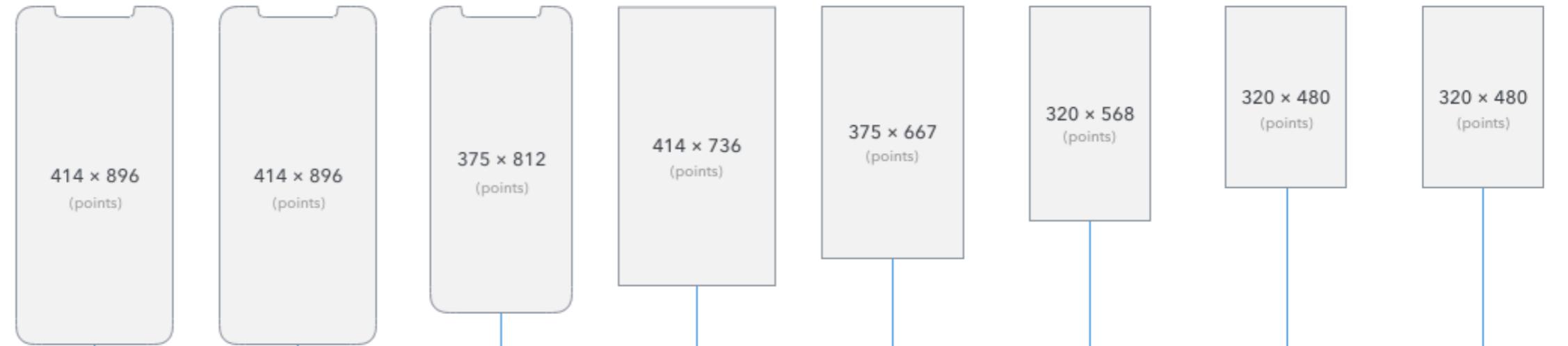
# iPhone Resolutions



## Points

At the beginning, coordinates of all drawings are specified in **points**.

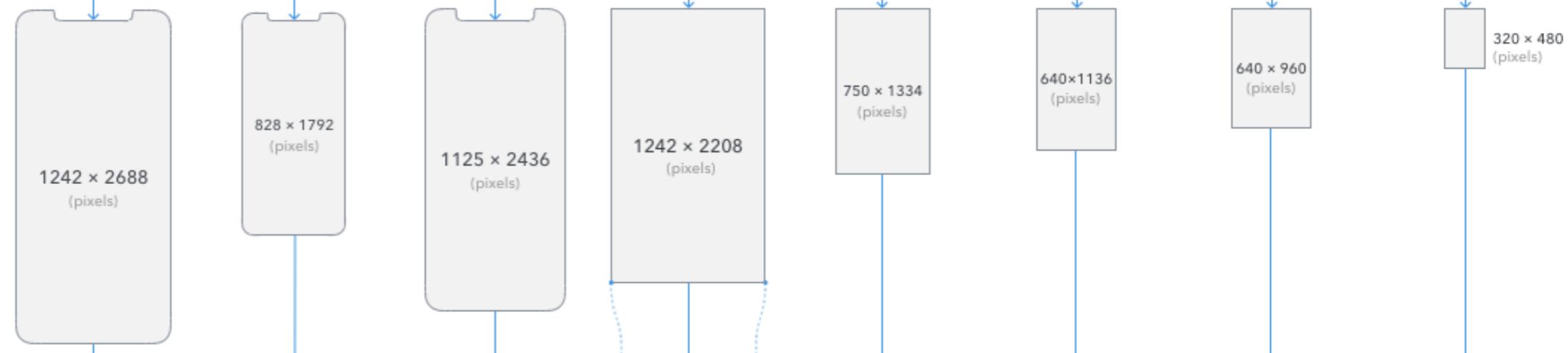
Points are abstract units, they only make sense in this mathematical coordinate space.

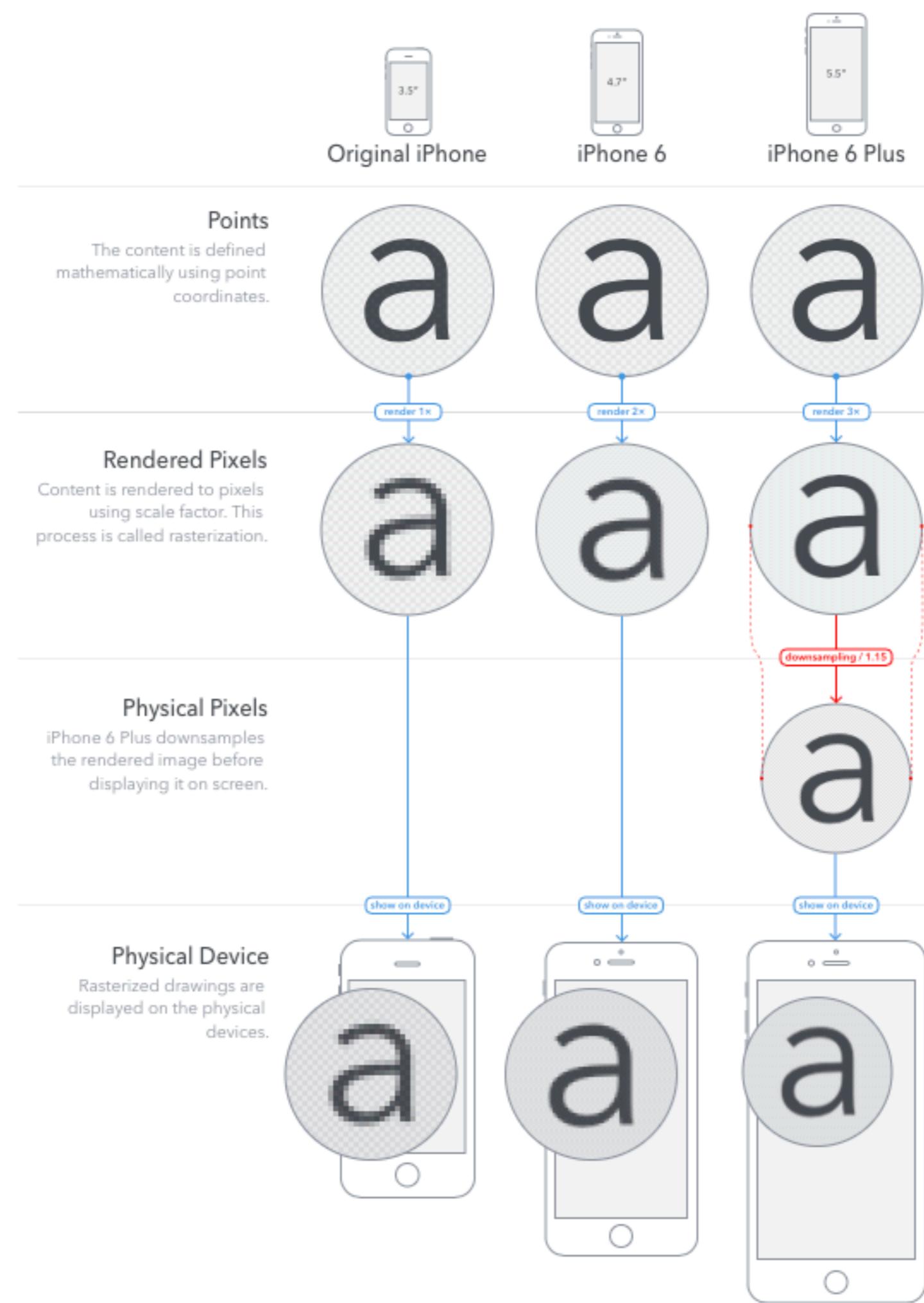


## Rendered Pixels

Point-based drawings are rendered into pixels. This process is known as **rasterization**.

Point coordinates are multiplied by scale factor to get pixel coordinates. Higher scale factors result in higher level of detail.





# UIViewContentMode

---

```
public enum UIViewContentMode : Int {  
    case scaleToFill  
    case scaleAspectFit  
    case scaleAspectFill  
  
    case redraw  
  
    case center  
    case top  
    case bottom  
    case left  
    case right  
    case topLeft  
    case topRight  
    case bottomLeft  
    case bottomRight  
}
```

———— **Scailing**

———— **Redrawing**

———— **Positioning**

# Scaling

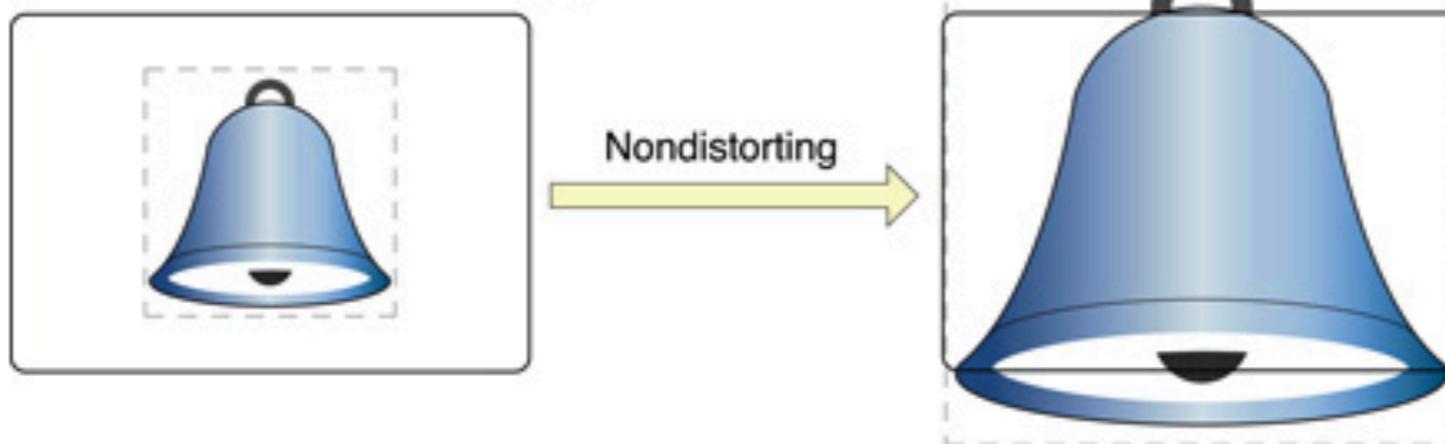
**UIViewContentModeScaleToFill**



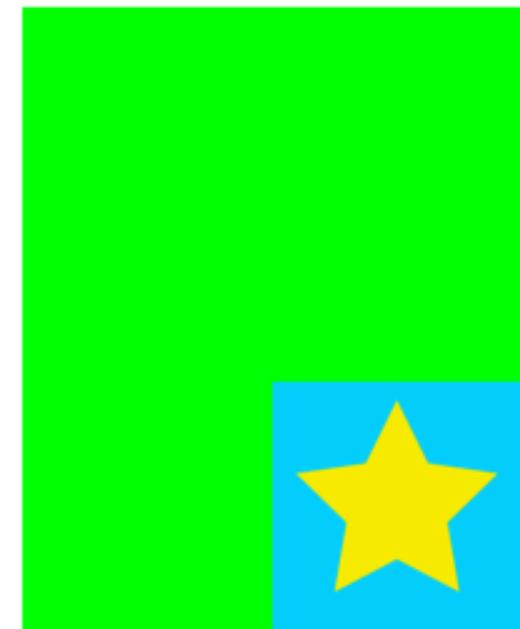
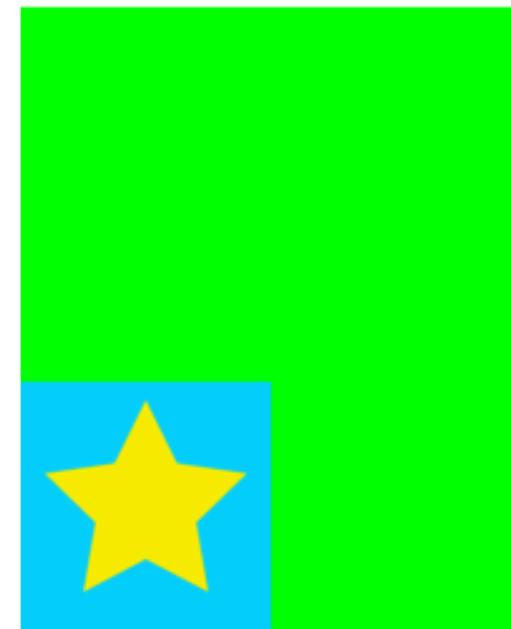
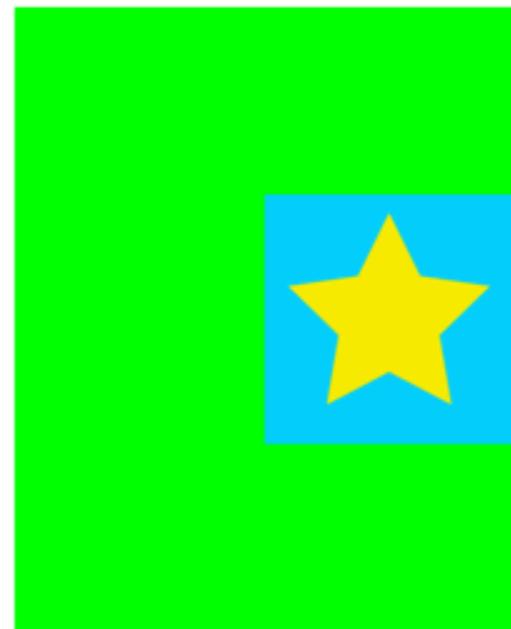
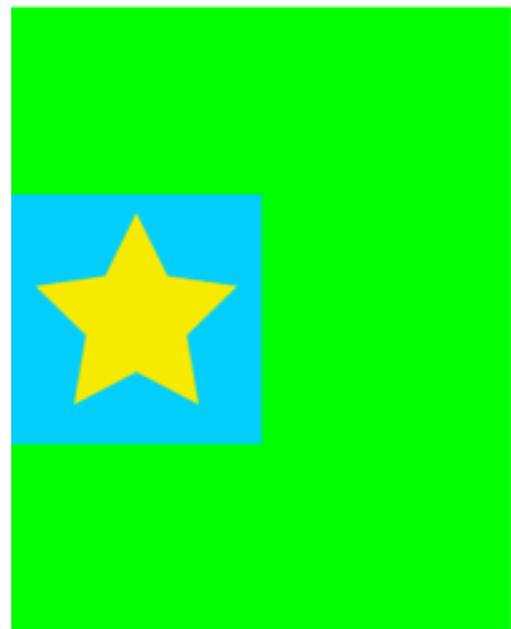
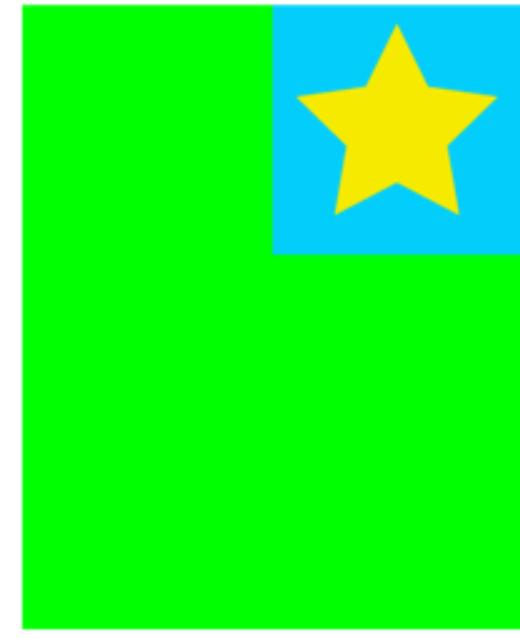
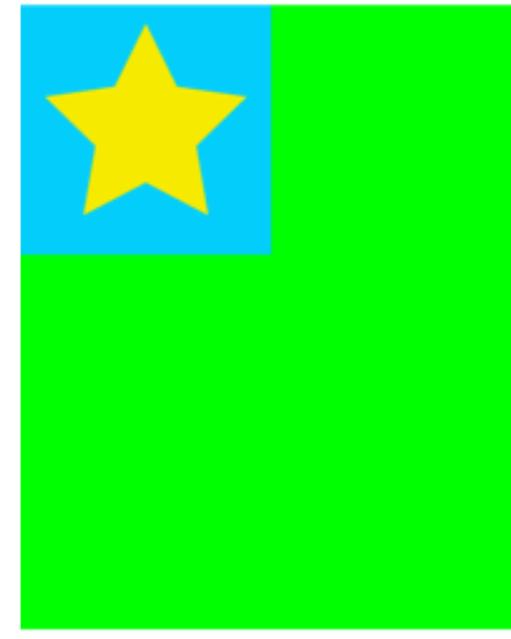
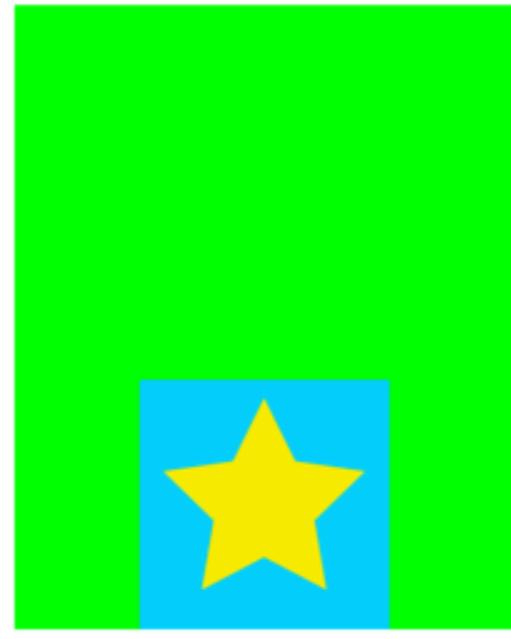
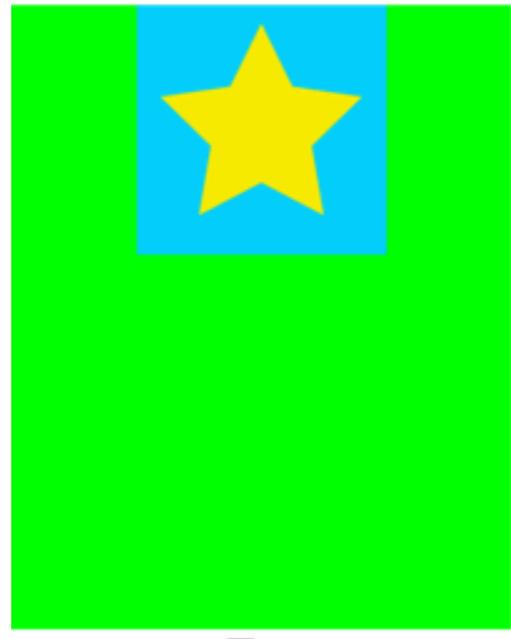
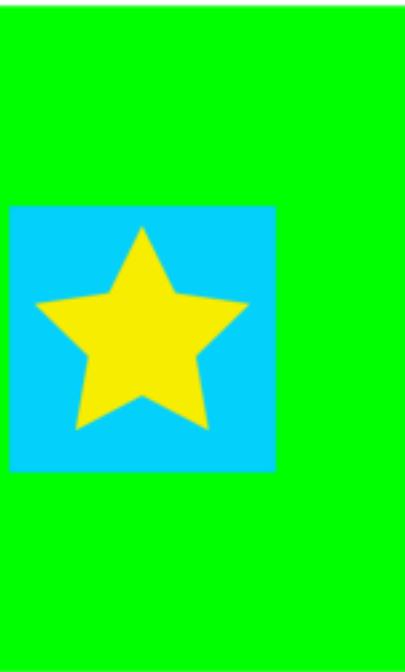
**UIViewContentModeScaleAspectFit**



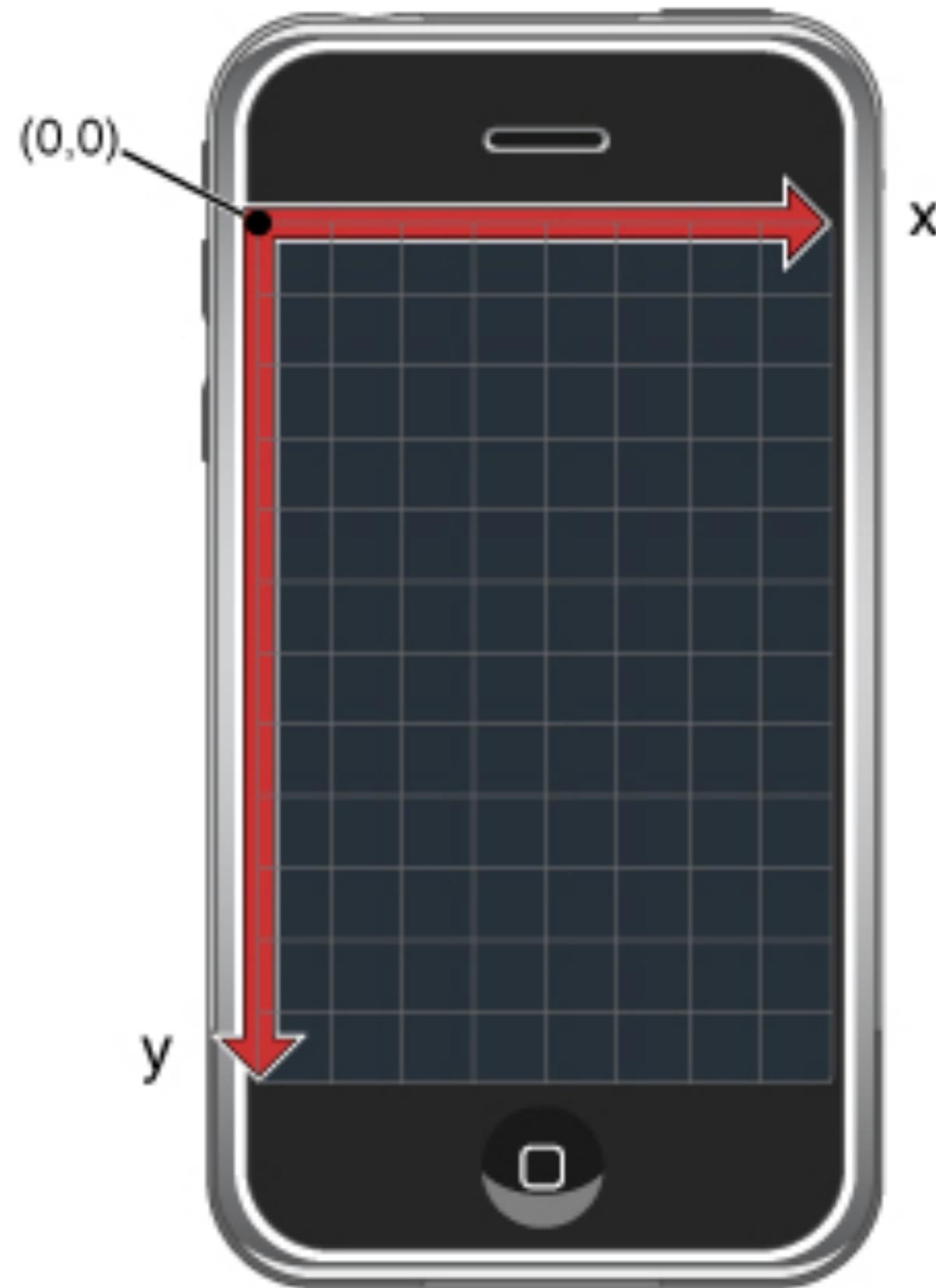
**UIViewContentModeScaleAspectFill**



# Positioning

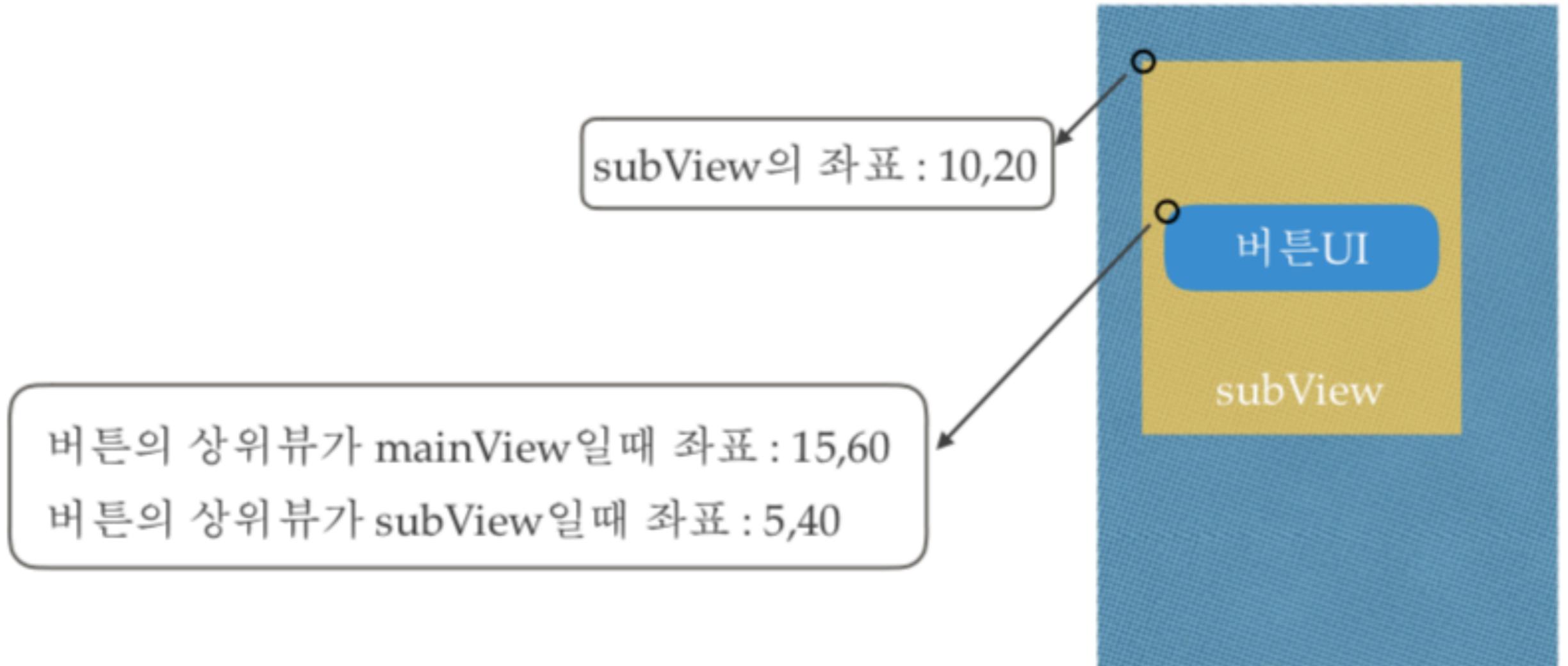


# Coordinate system orientation



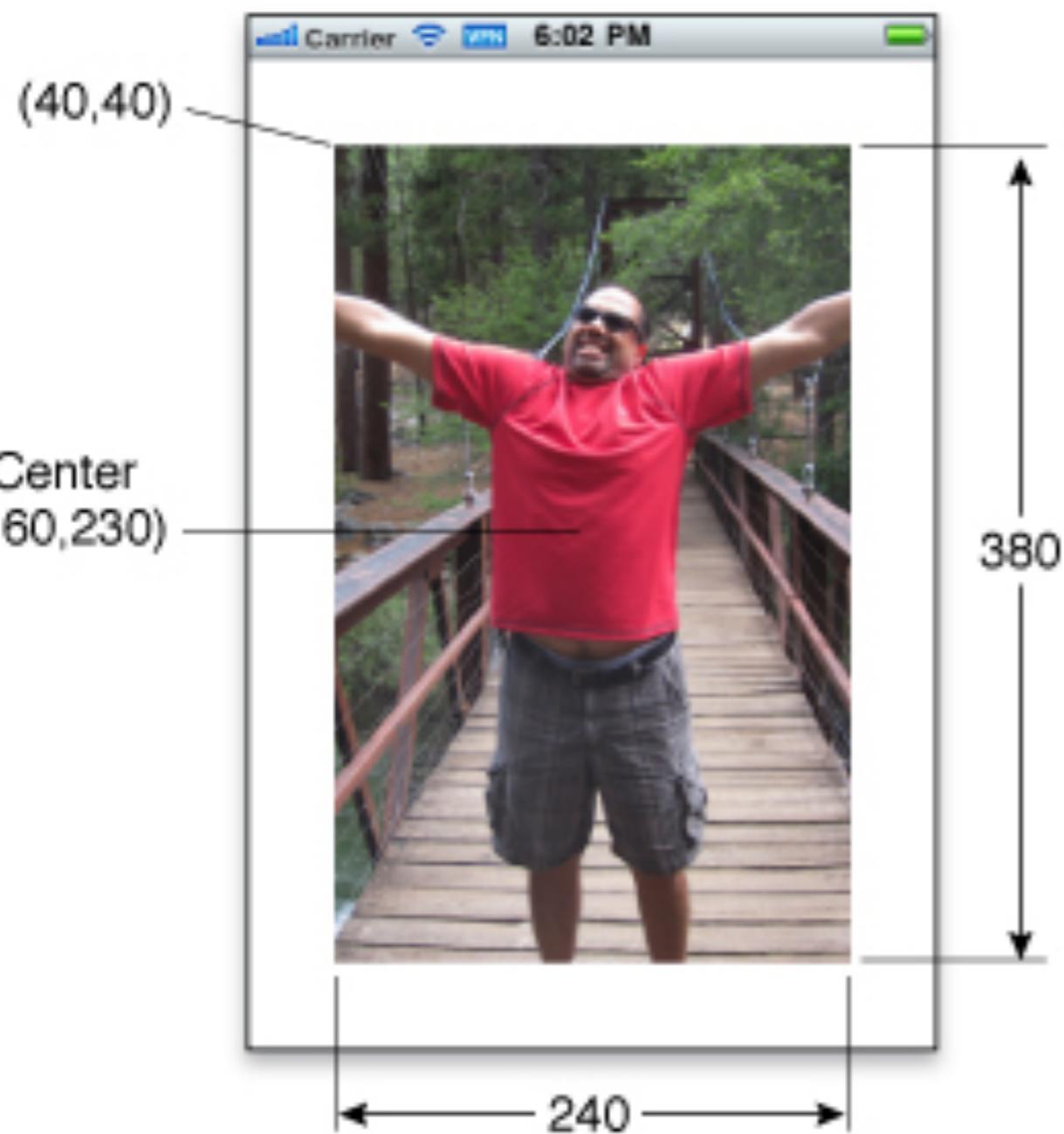
# View Frame

View Frame 의 좌표는 상위뷰를 기준으로 결정

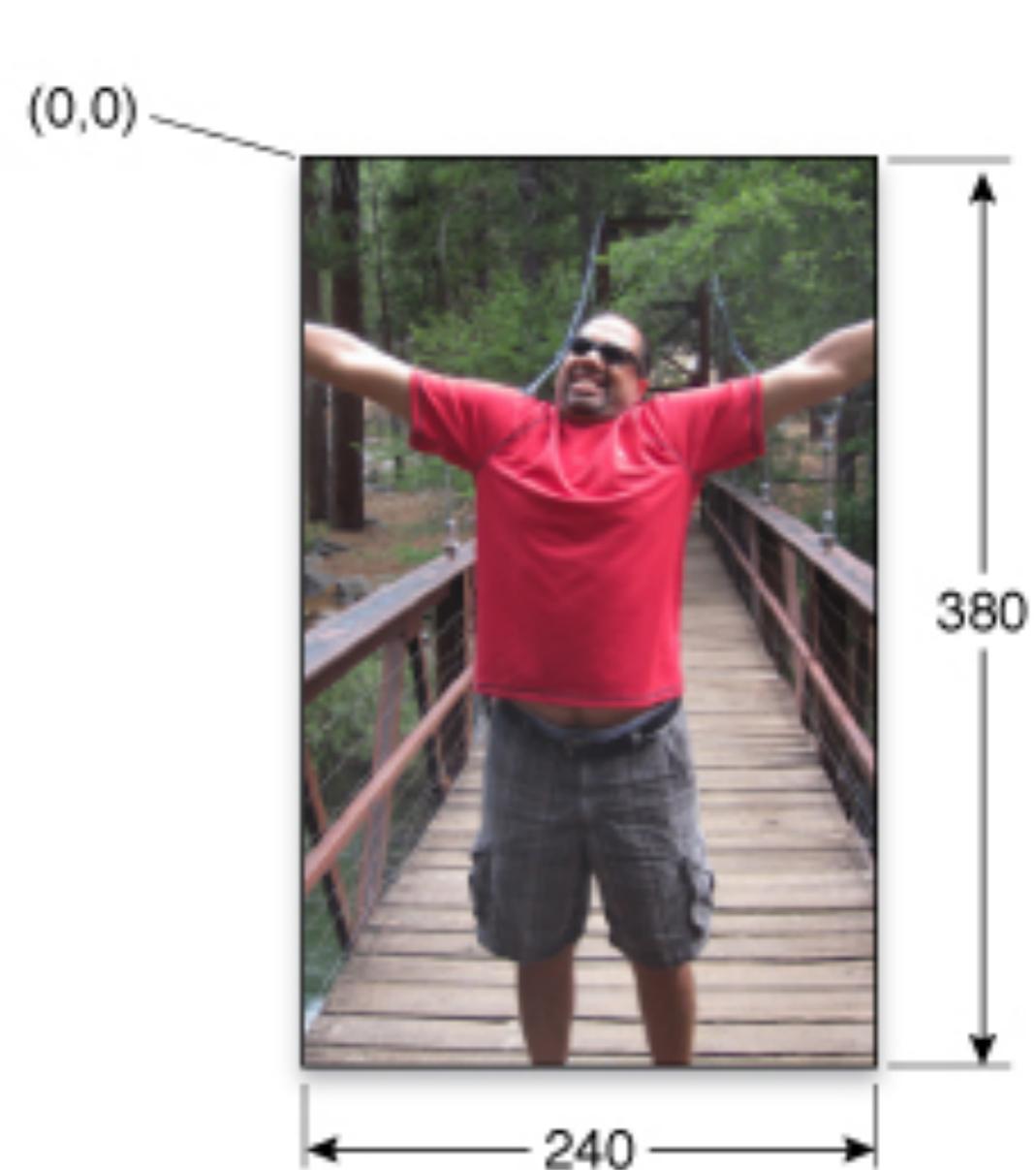


# Frame

Frame rectangle



Bounds rectangle



Super View

frame (0, 0, 375, 667)  
bounds (0, 0, 375, 667)

(0, 0)

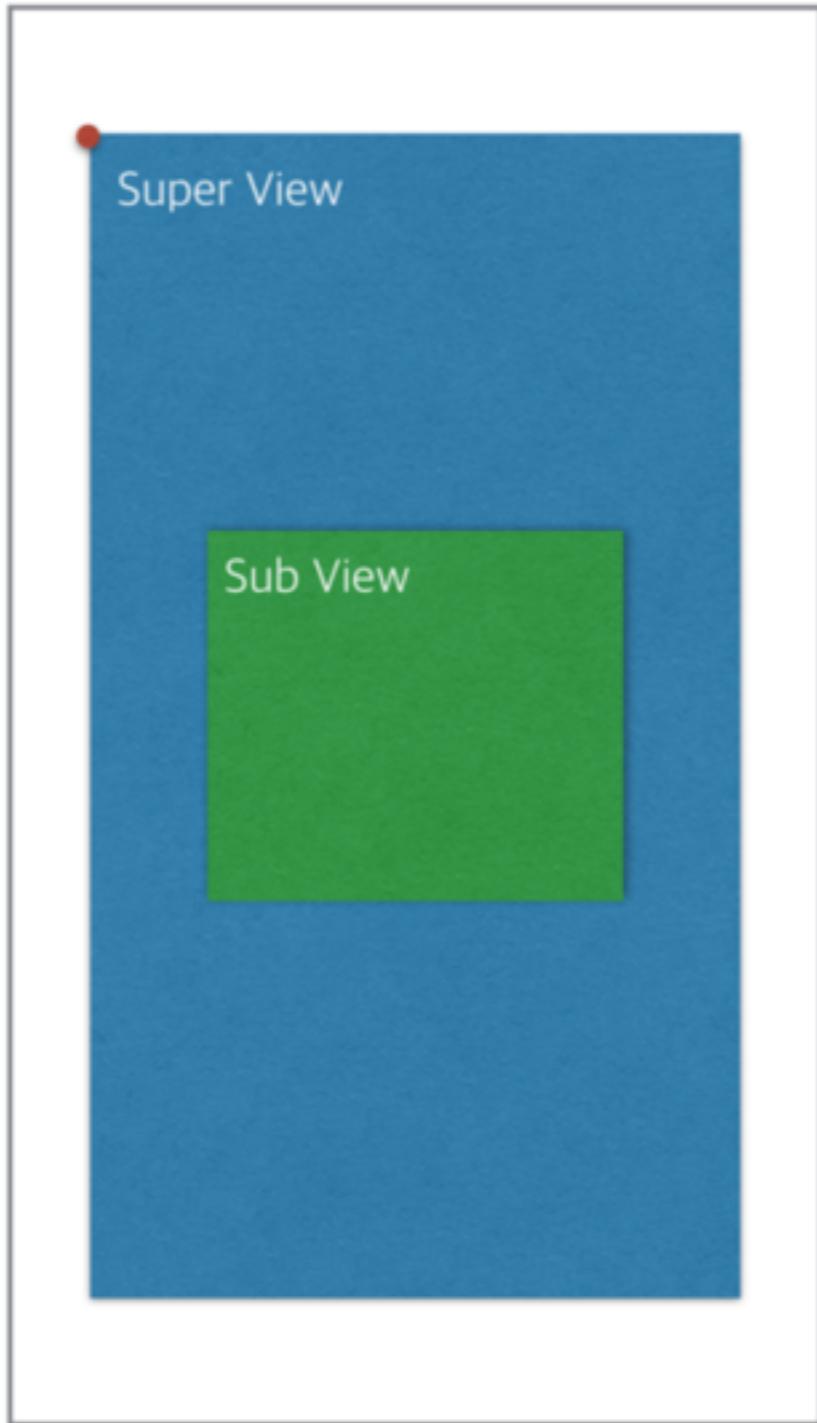
Sub View

frame (119, 217, 137, 127)  
bounds (0, 0, 137, 127)

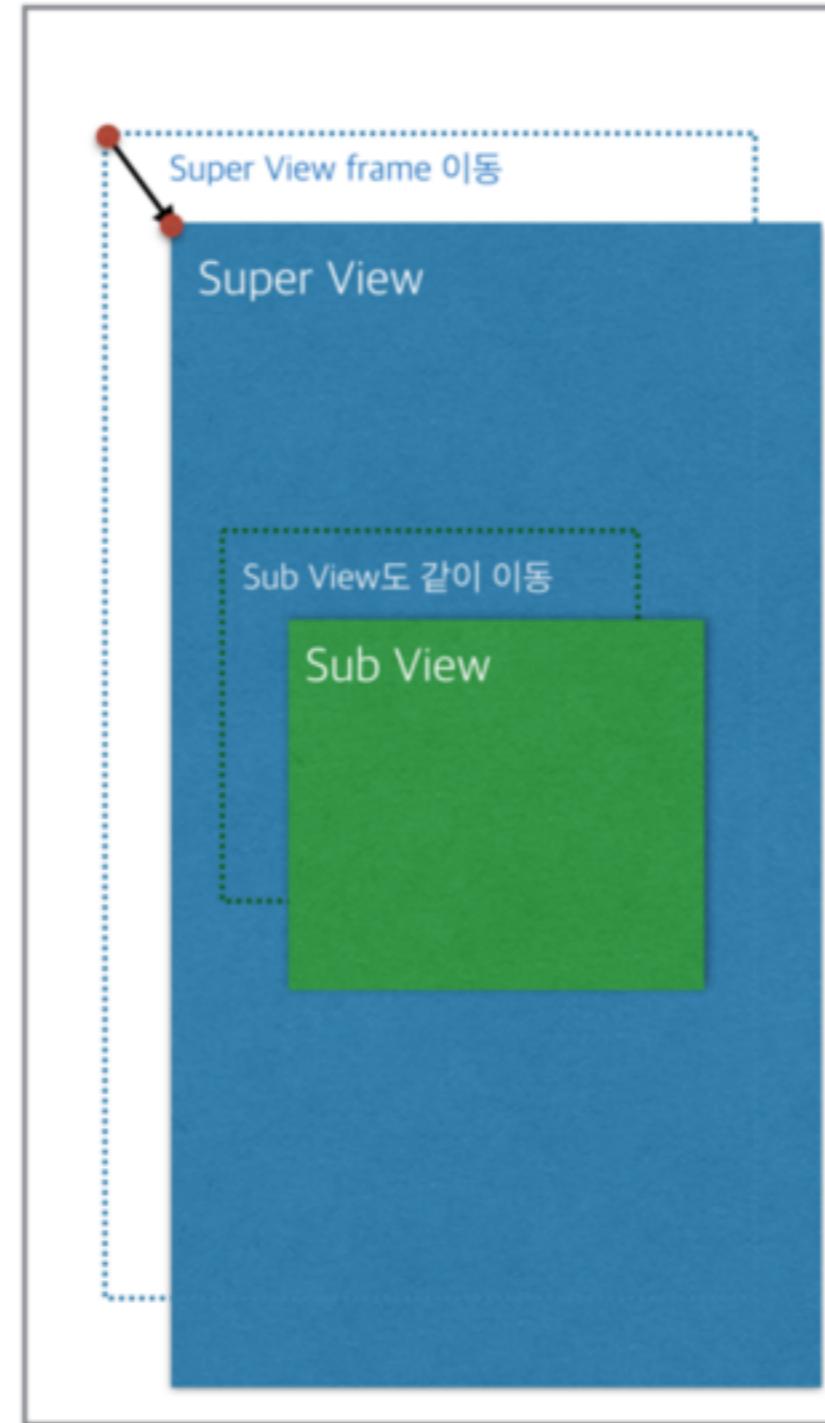
(119, 217)

# Frame

Super Super View

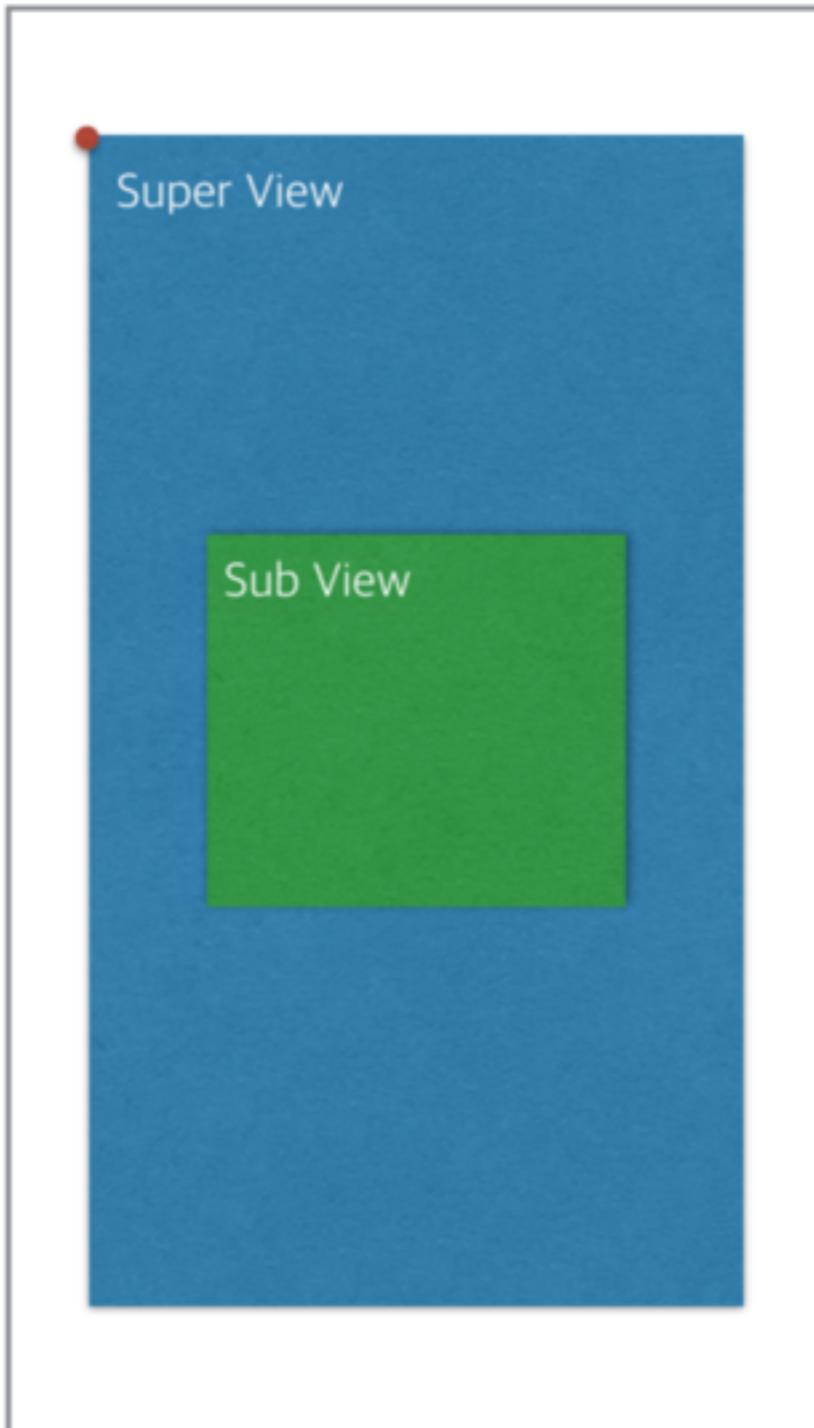


Super Super View

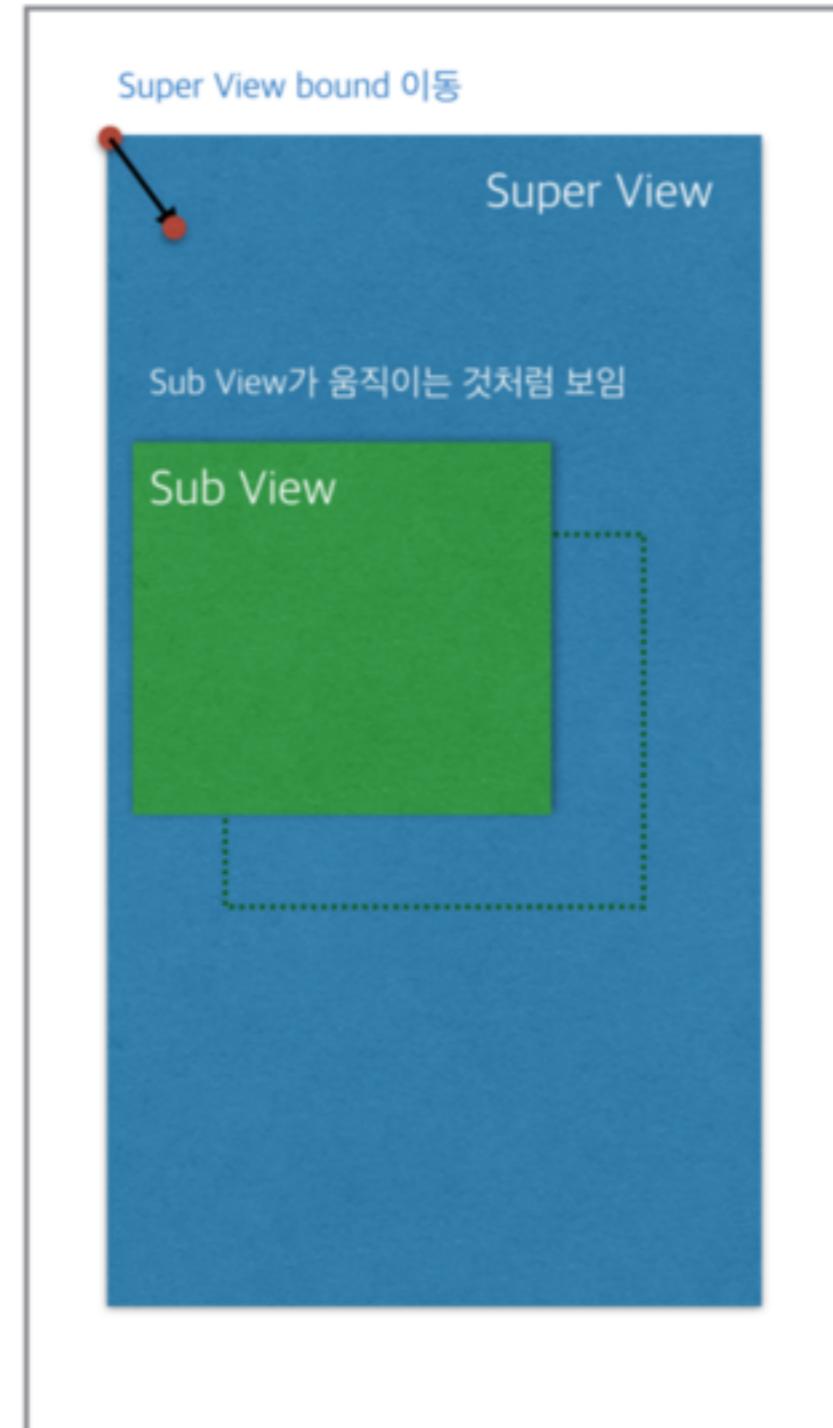


# Bounds

Super Super View



Super Super View



# Bounds

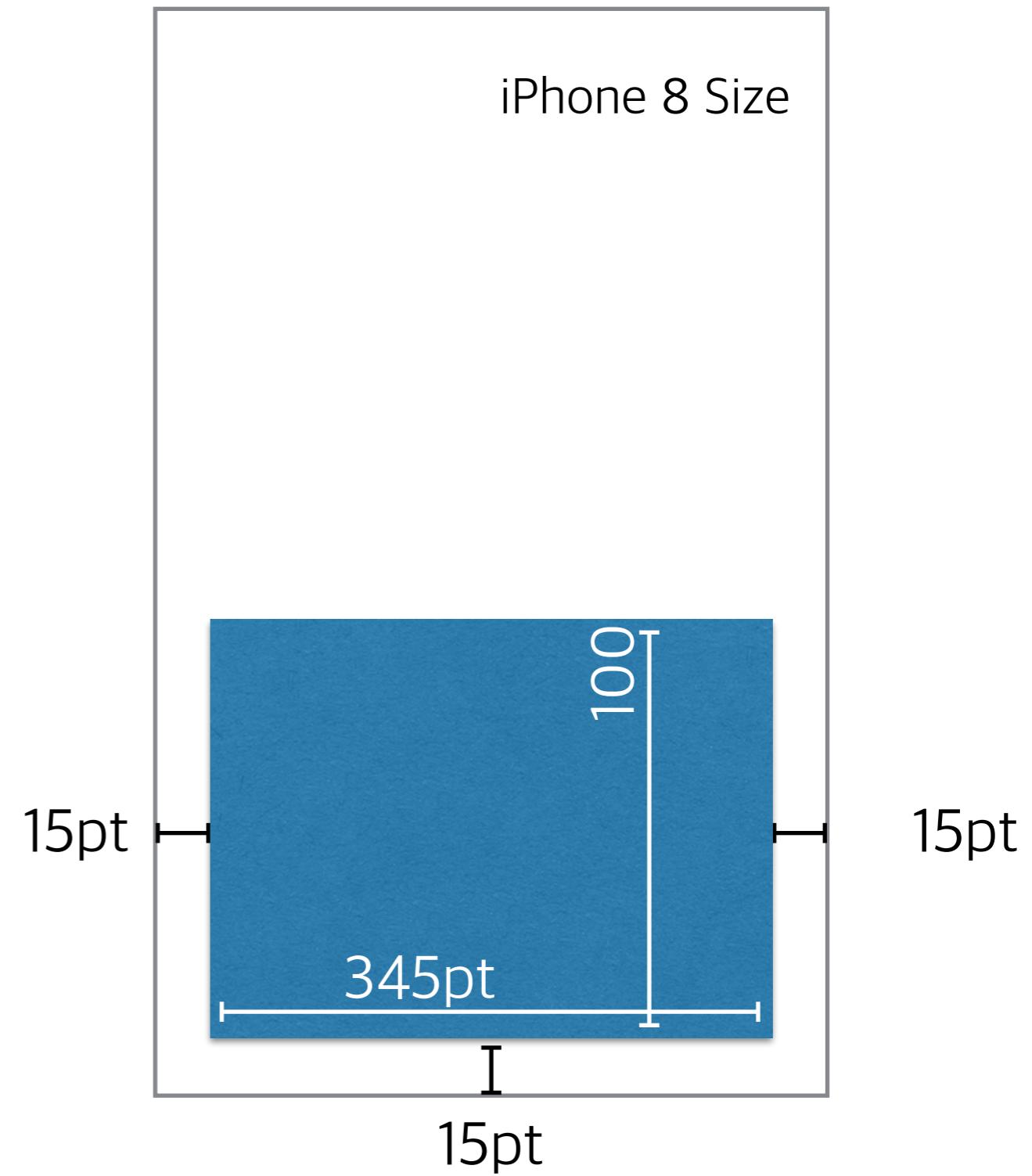
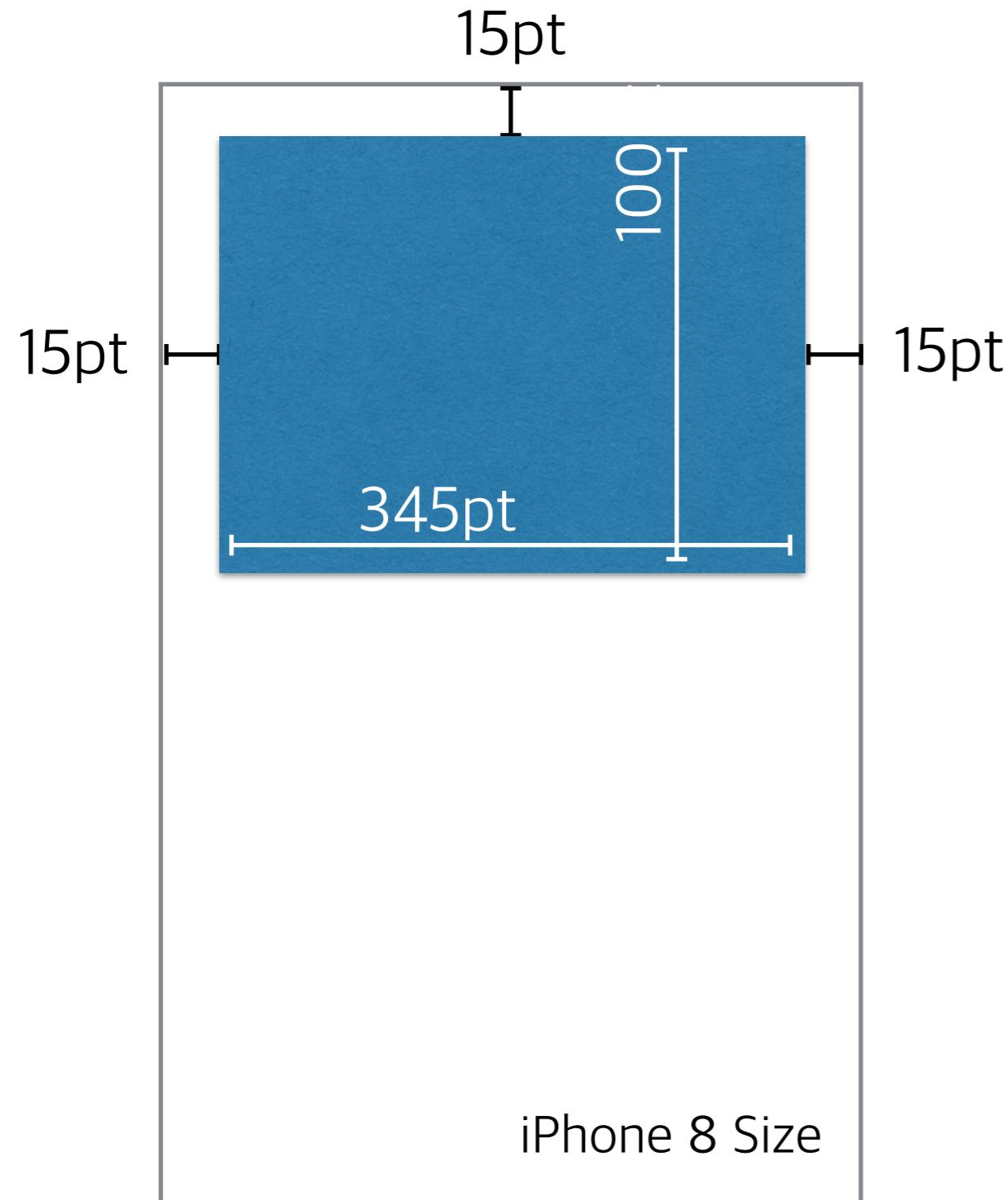


실습

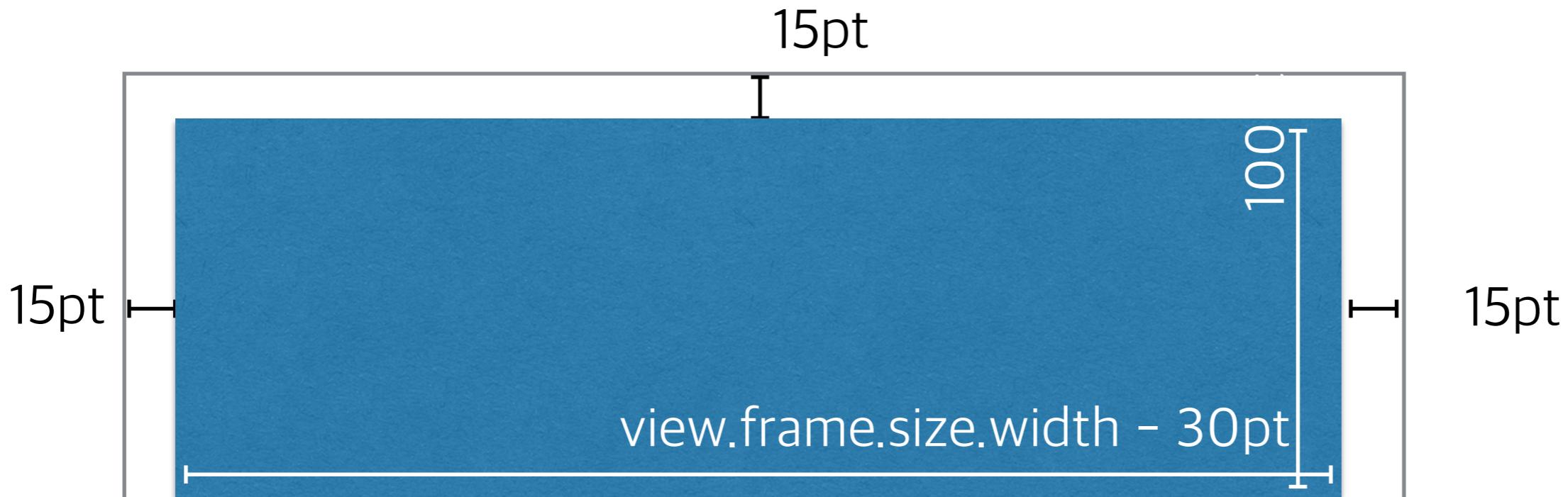
Giftbot

# Practice

---



# Practice



- SuperView의 size를 참조해서 view의 size를 정한다.
- view의 가로 사이즈가 기기 사이즈에 따라 유동적으로 변한다.

# Practice

- 서로 다른 배경색을 가지는 View 3개 생성
- 각 View의 상하좌우 여백은 30
- Frame, Bounds 의 차이에 대해 확인해보기

