

오토 레이아웃

- 뷰의 제약 사항을 바탕으로 뷰 체계 내의 모든 뷰의 크기와 위치를 동적으로 계산합니다.
- 뷰의 위치, 크기를 동적으로 지정해줍니다.

애플리케이션을 사용할 때 발생하는 외부 변경과 내부 변경에 동적으로 반응하는 사용자 인터페이스를 가능하게 합니다.

외부 변경

슈퍼뷰(부모뷰)의 크기나 모양이 변경될 때 발생
각각의 변화와 함께, 사용 가능한 공간을 가장 잘 사용할 수 있도록 뷰 체계의 레이아웃을 업데이트 필요

외부 변경 발생의 경우

- 사용자가 아이패드의 분할뷰를 사용하거나 사용하지 않는 경우
- 장치를 회전하는 경우
- 활성화콜과 오디오 녹음 바가 보여지거나 사라지는 경우
- 다른 크기의 클래스를 지원하기 원하는 경우
- 다른 크기의 스크린을 지원하기 원하는 경우

내부 변경

사용자 인터페이스의 뷰의 크기 또는 설정이 변경되었을 때 발생

내부 변경 발생의 경우

- 애플리케이션 변경에 의해 콘텐츠가 보여지는 경우
- 애플리케이션이 국제화를 지원하는 경우
- 애플리케이션이 동적 타입을 지원하는 경우

안전 영역(safeArea)

콘텐츠가 상태바, 내비게이션바, 톨바, 탭바를 가리는 것을 방지하는 영역



오토레이아웃 속성

Left, Right라는 개념이 왼쪽 오른쪽 개념이 있지만 leading, trailing은 text가 시작하는 지점, text가 끝나는 지점의 개념을 씁니다.

한국은 글을 쓸때 왼쪽 -> 오른쪽으로 쓰지만 아랍권은 오른쪽 -> 왼쪽으로 쓰기 때문입니다.

Top : 위

Bottom : 아래

Leading : text가 시작하는 지점(왼쪽)

Trailing : text가 끝나는 지점(오른쪽)

제약 우선도

- 콘텐츠 허깅 우선도(Content hugging priority) : 콘텐츠 고유 사이즈보다 뷰가 커지지 않도록 제한합니다. 다른 제약사항보다 우선도가 높으면 뷰가 콘텐츠 사이즈보다 커지지 않습니다.
- 콘텐츠 축소 방지 우선도(Content compression resistance priority) : 콘텐츠 고유 사이즈보다 뷰가 작아지지 않도록 제한합니다. 다른 제약사항보다 우선도가 높으면 뷰가 콘텐츠 사이즈보다 작아지지 않습니다.

<https://babbab2.tistory.com/135> 필수 참고!!!

코드로 오토레이아웃 설정

`translatesAutoResizingMaskIntoConstraints = false` 반드시 먼저 해줘야 됩니다.

기본값이 true로 되어있으며

true일시 AutoLayout을 따르지 않고 frame을 따르겠다(Frame-Based Layout)

false일시 AutoLayout을 따르겠다(AutoLayout)

Trailing, bottom 제약 조건시 중요한점

Ex) button의 trailing, bottom의 제약을 80으로부터 떨어지도록 걸고 싶을때

```
aButton.trailingAnchor.constraint(equalTo :  
self.view.safeAreaLayoutGuide.trailingAnchor, constant : -80).isActive = true  
aButton.bottomAnchor.constraint(equalTo :  
self.view.safeAreaLayoutGuide.bottomAnchor, constant : -80).isActive = true
```