Animations

Why do we need them?

- Beautiful transitions
- Way to catch eye

Animation

- class method UIView

```
class func animate(withDuration duration: TimeInterval, delay:
TimeInterval, options: UIViewAnimationOptions = [], animations:
@escaping (Void) -> Void, completion: ((Bool) -> Void)? = nil)
```

- convenient method

```
class func animate(withDuration duration: TimeInterval,
animations: @escaping (Void) -> Void)
```

UIViewAnimationOptions

- the way how animation will happen
- linear, increasing the spead
- can repeat, return to the final state
- can allow user interation

Transition between 2 views

```
class func transition(from fromView: UIView, to toView:
UIView, duration: TimeInterval, options:
UIViewAnimationOptions = [], completion: ((Bool) -> Void)? =
nil)
```

Animating constraints

- Change a constant

- call **layoutIfNeeded** on the view which has invalid layout

UIDynamicAnimator

- for example please see iPhone lock screen
- let animator = UIDynamicAnimator(referenceView: view)
- can simulate physics
- per node basis

!!! not a precise physics simulation!!!

UIDynamicBehaviour

- UIGravityBehaviour

```
var angle: CGFloat
var magnitude: CGFloat
```

- UICollisionBehaviour

```
var collisionMode: UICollisionBehaviorMode
var translatesReferenceBoundsIntoBoundary: Bool
```

- UIAttachmentBehaviour
- UIPushBehaviour
- UISnapBehaviour
- UIDynamicItemBehavior

```
allowsRotation, friction, elasticity
let density: CGFloat
```