**一、 ARC环境**

* ***单个viewController的生命周期***
  + initWithCoder:(NSCoder \*)aDecoder：（如果使用storyboard或者xib）
  + loadView：加载view
  + viewDidLoad：view加载完毕. 在创建VC的内容视图（其视图层次结构的顶部）并从storyboard中加载时调用。保证在调用此方法时，VC的出口具有有效值。使用此方法可以执行视图控制器所需的任何其他设置。通常，viewDidLoad()首次创建其内容视图时，iOS 仅调用一次。但是，在第一次实例化控制器时不一定创建内容视图。相反，它是在系统或任何代码第一次访问控制器view属性时被惰性创建的。
  + viewWillAppear：控制器的view将要显示. 在将视图控制器的内容视图添加到应用程序的视图层次结构之前调用。使用此方法可以触发在屏幕上显示内容视图之前需要进行的所有操作。尽管有名称，只是因为系统调用了此方法，所以不能保证内容视图将变为可见。该视图可能被其他视图遮挡或隐藏。此方法仅指示内容视图将被添加到应用程序的视图层次结构中
  + viewWillLayoutSubviews：控制器的view将要布局子控件
  + viewDidLayoutSubviews：控制器的view布局子控件完成  
    这期间系统可能会多次调用viewWillLayoutSubviews 、 viewDidLayoutSubviews 俩个方法
  + viewDidAppear:控制器的view完全显示. 在将视图控制器的内容视图添加到应用程序的视图层次结构之后立即调用。使用此方法可以触发在屏幕上显示视图后立即进行的所有操作，例如获取数据或显示动画。尽管有名称，只是因为系统调用了此方法，所以不能保证内容视图可见。该视图可能被其他视图遮挡或隐藏。此方法仅指示内容视图已添加到应用程序的视图层次结构中。
  + viewWillDisappear：控制器的view即将消失的时候. 在将视图控制器的内容视图从应用程序的视图层次结构中删除之前调用。使用此方法可以执行清除任务，例如提交更改或退出第一响应者状态。尽管有名称，但系统不会仅由于内容视图将被隐藏或遮挡而调用此方法。仅在要从应用程序的视图层次结构中删除内容视图时才调用此方法。  
    这期间系统也会调用viewWillLayoutSubviews 、viewDidLayoutSubviews 两个方法
  + viewDidDisappear：控制器的view完全消失的时候. 在从应用程序的视图层次结构中删除视图控制器的内容视图之后立即调用。使用此方法执行其他拆卸活动。尽管有名称，但系统不会仅因为内容视图已隐藏或模糊而调用此方法。仅当内容视图已从应用程序的视图层次结构中删除时，才调用此方法。
* ***多个viewControllers跳转***
  + 当我们点击push的时候首先会加载下一个界面然后才会调用界面的消失方法
  + initWithCoder:(NSCoder \*)aDecoder：ViewController2 (如果用xib创建的情况下）
  + loadView：ViewController2
  + viewDidLoad：ViewController2
  + viewWillDisappear：**ViewController1** 将要消失
  + viewWillAppear：ViewController2 将要出现
  + viewWillLayoutSubviews ViewController2
  + viewDidLayoutSubviews ViewController2
  + viewWillLayoutSubviews:**ViewController1**
  + viewDidLayoutSubviews:**ViewController1**
  + viewDidDisappear:**ViewController1** 完全消失
  + viewDidAppear:ViewController2 完全出现
* 小结：  
  -整个控制器声明周期： viewDidLoad -> viewWillAppear -> viewWillLayoutSubviews -> viewDidLayoutSubviews -> viewDidAppear -> viewWillDisappear -> viewDidDisappear