1. QQ空间可拉伸头部控件
2. 被拉伸控件（顶部图片）位于tableView的下方，然后通过设置透明的表头。
3. 根据tableView的contentOffset（上滚大于0，下滚小于0）触发navigationBar改变背景颜色透明度。
4. 关键点
5. navigationBar颜色渐变  navBar.backgroundColor = [[UIColor whiteColor] colorWithAlphaComponent:yOffset/headHeight];  这个方法提供了颜色的渐变，很好的避免了修改navBar.alpha引起其子视图透明度的改变这一问题。
6. 控件放大时定点左标修改  frame.origin.x -= (frame.size.width - originalFrame.size.width)/2;  增加的宽度，水平方向平分（contentOffset >0 改变 y，contentOffset < 0 ,改变 height、width、x;）

2.序列帧动画（loading）

1、通过 UIImage \*image = [[UIImage alloc] initWithContentsOfFile:filePath];方式设置animationImages会快速释放图片，不要通过UIImage \*image = [UIImage imageNamed（这种方式设置图片使用完成后，不会直接被释放掉，具体释放时间由系统决定，）。

2、开启动画时：

self.loadingImage.animationImages = 图片数组;

[self.loadingImage startAnimating];

3、停止动画时：

[self.loadingImage stopAnimating];

self.loadingImage.animationImages = nil;(一定要把数组置空，否则还在内存中)

3.国际化(I18N)

1. 选中project->Info->Localizations，然后点击"+"，添加需要国际化/本地化的语言（默认需要勾选Use Base Internationalization）----- “File type”有2中选择（1.Interface Builder Storyboard:对系统整个界面也设置国际化，这是最彻底对国际化；2.Localizable Strings:只对字符串国际化）

2. App的名称国际化: 选中Info.plist，按下键盘上的command + N，选择**Strings File**（iOS->Resource->Strings File）, 文件名字命名为**InfoPlist**，且**必须是这个名字.** 选中InfoPlist.strings，在Xcode的File inspection（Xcode右侧文件检查器）中点击Localize，目的是选择我们需要本地化的语言, 点击Localize后，会弹出一个对话框，展开对话框列表，发现下拉列表所展示的语言正是我们在上面配置的需要国际化的语言，选择我们需要本地化的语言，然后点击对话框的Localize按钮,最后我们分别用不同的语言给InfoPlist.strings下的文件设置对应的名字：CFBundleDisplayName = "Localizable App Name";（CFBundleDisplayName可以使用双引号，也可以不使用双引号！）

3. 代码中字符串的本地化: 和应用名称本地化一样，首先需要command + N，选择iOS -> Resource -> Strings File，文件名必须命名为**Localizable，**我们只需要使用Foundation框架自带的NSLocalizedString(key, comment)这个宏根据Key获取对应的字符串，然后赋值给代码中的字符串。

// NSLocalizedString(key, comment) 本质

// NSlocalizeString 第一个参数是内容,根据第一个参数去对应语言的文件中取对应的字符串，第二个参数将会转化为字符串文件里的注释，可以传nil，也可以传空字符串@""。

#define NSLocalizedString(key, comment) [[NSBundle mainBundle] localizedStringForKey:(key) value:@"" table:nil]

**（因为启动程序时，系统将根据语言加载相应的文件得到其对应的字符串文件，这个字符串可以通过系统将NSLocalizedString中的宏生成名为“Localizable.strings”的文件。那么如何让系统加载我们自己命名的本地化文件而非系统默认的Localizable.strings呢？这就是 NSLocalizedStringFromTable(<#key#>, <#tbl#>, <#comment#>)的用处。**

**也就是说，如果你的strings文件名字不是Localizable而是自定义的话，如VVS.strings，那么你就得使用NSLocalizedStringFromTable这个宏来读取本地化字符串。**

// key：系统根据key取字符串

// tbl：自定义strings文件的名字

// comment：可以不传

NSLocalizedStringFromTable(<#key#>, <#tbl#>, <#comment#>)**）**

4. 图片本地化: 有两种方式，第一种方式和本地化代码中的字符串一样，第二种直接对图片本地化

5. 查看/切换本地语言: NSArray \*languages = [[NSUserDefaults standardUserDefaults] valueForKey:@"AppleLanguages"];

NSString \*currentLanguage = languages.firstObject;

切换语言无需在模拟器中设置，只需要在Xcode中进行如下设置： Edit->Scheme->Run->Arguments Passed On Launch ->-AppleLanguages (语言代码)。例如，我们模拟器此时虽然是韩语，如果通过上述步骤设置为zh-Hans，那么语言环境将会变为汉语