# UI基础

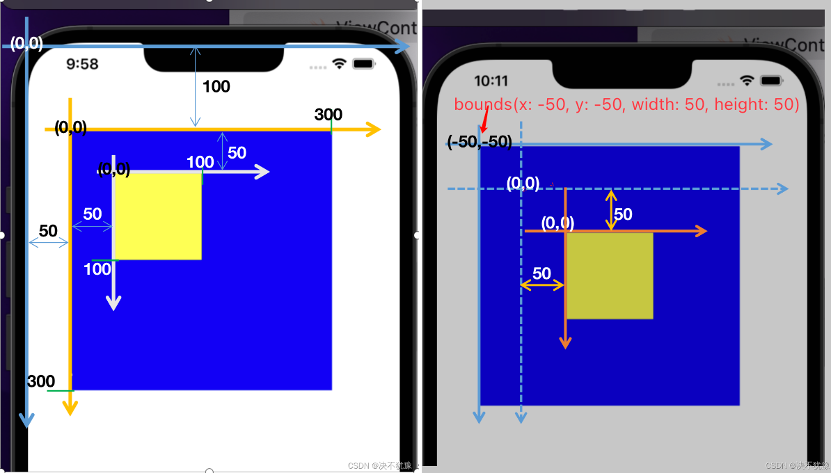
**bounds和frame的区别**

frame:  frame(x: 100, y: 100, width: 50, height: 50)

当前视图相对于父视图的位置

bounds: bounds(x: -100, y: -100, width: 50, height: 50)

bounds中的x，y值决定当前视图左上角的坐标是多少，也会影响子视图的位置。



（csdn图片）

# 数据的编解码、传递、存储

**SwiftJSON作用：**

SwiftJSON是一个第三方包，简单来说，就是方便在swift中对json数据进行操作。在swift中进行网络请求后会收到json格式的数据，但是该格式在swift中不方便取值，SwiftJSON对json数据解析后，就可以通过链式方法取值

**页面跳转方式：**

1. present/dissmiss
2. push/pop

**页面传值方法：**

可选型、隐式可选型：

系统更新视图，不会调用DataSource方法

reloadData()

**本地存储方法：**

UserDefaults:

存数据：

调用UserDefaults.standard.set(value,key),value=data,在本地sandbox中以plist的形式存储

data是通过JSONEncoder()编码的数据,被编码的数据必须是基本数据类型或者遵循Codable(Encodable&Decodable)协议的数据类型

取数据：

调用UserDefaults.standard.data(key),取到的数据通过JSONDecoder().decode([Todo].self, from: data)解码，需要传入解码类型（元类型：[String].self）和解码数据

CoreData:

CoreData是Swift中的一个数据仓库框架，能够方便开发者对数据库进行操作。

# 类与结构体

# **//结构体和类的区别：**

结构体在栈、类在堆

# 栈、堆

**//栈为什么比堆快**

**//深拷贝、浅拷贝**

深拷贝：

         浅拷贝：

**//swift 写时复制**

**//看文章**

//事件传递及响应链

**//intrinsicContentsize**

**//单例模式**

设计模式

一个类的初始化方法init()的权限是私有的，在类内有该类的实例化对象，可供外部调用。可以使一个类只有一个实例化对象。