Subclassing UIView

В этой главе создаем собственный подкласс UIView который заполняет экран концентрическими кругами (рис. Стр. 115)

**The drawRect метод.**

Каждый UIView реализует drawRect метод – он содержит код для view (например drawRect UIButton – метод рисует прямоугольник с закругленными углами и заголовком в центре)

Каждый раз когда UIView должен быть нарисован система подготавливает контекст графики для данного view. Экземпляру UIView отправляется сообщение drawRect с готовым контекстом.

Типом графического контекста является CGCOntextRef (Core Graphics Context Reference) – он отвечает за прорисовку изображения.

Для того чтобы нарисовать view можно использовать возможности Objective-C определенные в рамках UIKit либо используя возможности С (в рамках графического ядра). В этой главе попробуем и то и другое.

В моей программе (обычный проект создал класс HypnosisView – NSObject (затем переопределил супер класс в UIView)) и теперь буду переопределять метод drawRect (стрю 116)

Переопределение стандартного метода drawRect для HypnosisView.m

Пример

Здесь я создаю структуру CGRect – это прямоугольник (он еще называется грязным прямоугольником) его обычно игнорируют задавая свою инструкцию по рисованию

**Создание экземпляра UIView**

Есть два варианта создания UIView

1).Визуально с помощью Interface Builder

2). Программно с alloc и initWithFrame

В этой главе будет создан программно:

Стр118 (лучше взять из приложения написал там комменты)

**Drawing text and Shadows**

Добавим на наше view еще текст и тень:

Открываем HupnosisView.m и добавляем код:

(стр 120 лучше из проги)

**Using UISCrollView**

Для того, чтобы пользователь мог перемещаться между по view делают подтаблицу из UIScrollView

(рисунок 6.4 стра. 121)

В делегате приложения добавляю UIScrollView (стрю 121 лучше из кода)

Запускаю приложение ScrollView отображается (справа и внизу он есть), но я не могу зуммировать (перемещаться по изображению) нахожусь лишь в одном window из 4.

**ZOOMING**

**Чтобы** осуществлять масштабирование нужно реализовать методы из делегата UIScrollView. Указываем в делегате приложения, что он поддерживает протокол UIScrollViewDelegate

Далее в .m делегата приложения указываю что для моего ScrollView делегатом является делегат приложения

[scrollview setDelegate:self]

А также необходимо установить пределы масшатабирования

[scrollview sitMinimumZoomScale:0.5];

[scrollview sitMaximumZoomScale:5]

Далее в делегате приложения реализую метод из протокола для UIScrollView:

-(UIView \*)viewForZoomingInScrollView (UIScrollView \*)scrollView

{

return view;

}

Запускаю приложение с помощью ALT делаю масштабирование.

**Hiding the Status Bar. (скрытие Status Bar)**

Когда запускается мое приложение я не хочу, чтобы было видно время и заряд батареи (status bar)

Для этого перед [self.windiw makeAndVisible]

Пишу

[[UIApplication sharedApplication] setStatusBarHidden:YES WithAnimation: UIStatusBarAnimationFade];

Если запустить приложение, то строка состояния пропадет после запуска приложения.

Вариант 2:

Скрыть строку состояния до появления приложения на дисплее

Открываю файл .plist и добавляю ключ: Status Bar is initially hidden YES

Либо target-info-ключи-status bar is Initially hidden (по сути то же редактирование файла .plist только через другое место)