

# Домашнее задание №1

Как было сказано на лекции, на протяжении всего курса мы будем разрабатывать чат. И для начала тебе нужно создать проект и настроить его:

- Deployment target: 11.0
- Devices: Universal или iPhone

Добавь AppIcon по своему усмотрению, используя картинки подходящего размера.

После этого нужно закрепить знания о жизненном цикле приложения и понять, как приложение переходит из одного состояния в другое. Тебе нужно будет вывести в консоль сообщения для каждого состояния. Для этого:

- В **AppDelegate** в каждый метод жизненного цикла добавь вывод в консоль названия этого метода. Для этого можно использовать литерал **#function**. К выводу названия метода добавь вывод состояний, при переходе между которыми он вызывается. Вывод в консоли должен выглядеть примерно так:

Application moved from <стейт из которого переходим> to <стейт в который переходим>:  
<здесь название метода>

- Сверни/разверни приложение, закрой. В консоль должны выводиться названия вызываемых при этом методов **AppDelegate**.

Также мы познакомились с событиями жизненного цикла **UIViewController**. Для того, чтобы лучше понять его жизненный цикл:

- Добавь во **ViewController** следующие методы:
  - **viewWillAppear**
  - **viewDidAppear**
  - **viewWillLayoutSubviews**
  - **viewDidLayoutSubviews**

- **viewWillDisappear**
- **viewDidDisappear**
- Не забудь добавить вызов методов из суперкласса в начале метода (пример: `super.viewDidAppear`)
- В каждый метод добавь вывод в консоль названия вызываемого метода, аналогично **AppDelegate**.

Выполненное домашнее задание нужно будет выложить в репозиторий (Github, Bitbucket). Для этого у тебя должны быть начальные знания Git и понимание, как с ним работать. Если ты еще не знаком(а) с Git:

- Прочитай основную информацию на <https://git-scm.com/book/ru/v1/Введение-Основы-Git>
- Прочитай про основные команды, которые тебе потребуются:
  - **git add**
  - **git commit**
  - **git push**
  - **git pull**
  - **git merge**
- Прочитай про **.gitignore** (<https://git-scm.com/docs/gitignore>), добавь файл **.gitignore** в свой проект. Заполни **.gitignore** таким образом, чтобы в коммит не попадали:
  - папки **build**, **DerivedData**, **xcuserdata**
  - файлы с расширением **xccheckout**, **xcscmblueprint**

### \* **Задание со звездочкой:**

Как ты мог заметить, в основном в домашнем задании придется логировать события жизненного цикла. Попробуй реализовать решение, позволившее бы включить и отключить логи перед компиляцией (в интерфейсе приложения не надо добавлять никаких кнопок для этого).

### **Как получить максимальный балл?**

Для этого нужно соблюсти следующее:

- Проект должен собираться без ошибок и warning'ов

- Проект не должен падать
- Файл .gitignore заполнен правильно
- События жизненного цикла логируются корректно
- Выполнено задание со звездочкой